A Shadow Cursor for Calibrating Screen Coordinates of Tabletop Displays and Its Evaluation

Makio ISHIHARA^†a, Member and Yukio ISHIHARA^†b, Nonmember

SUMMARY This paper discusses the use of a common computer mouse as a pointing interface for tabletop displays. In the use of a common computer mouse for tabletop displays, there might be an angular distance between the screen coordinates and the mouse control coordinates. To align those coordinates, this paper introduces a screen coordinates calibration technique using a shadow cursor. A shadow cursor is the basic idea of manipulating a mouse cursor without any visual feedbacks. The shadow cursor plays an important role in obtaining the angular distance between the two coordinates. It enables the user to perform a simple mouse manipulation so that screen coordinates calibration will be completed in less than a second.

key words: shadow cursor, screen coordinates calibration, tabletop display, interaction design

1. Introduction

A tabletop display is a table type computer system. It has a horizontal touch screen and provides graphical user interface or GUI with users. The tabletop display gives users a natural way to work together sharing new ideas and opinions, easily reflecting on their decisions. Thus tabletop displays are suitable for collaborative work.

One of the problems around tabletop displays comes from a pointing interface. A pointing interface is generally referred to a system that gives users a means of pointing at the interest spot on a computer screen. A common way of a pointing interface for tabletop displays is provided by a touch screen. Users can touch and drag the content on the screen. If the size of the touch screen becomes large enough for users not to reach the content by their own hands, it will not work for a pointing interface. It is called out-of-reach target problem.

There are some possible solutions to this problem. The first is the use of sophisticated magnetic sensors often seen in CAVE systems. Magnetic sensors are capable of tracking any subtle changes in the magnetic field (it is mostly an artificial one) so that they locates themselves in the field. It is very sensitive but expensive. The second is the use of handheld cameras. A handheld camera is designed to track the coordinates of a computer screen and calculate homography between the computer screen and the camera image then the center of the camera image works as a graphical pointer. It is inexpensive but it only works as far as the handheld camera captures the entire computer screen. The third is the use of gray-code [1]. Gray-code is a way to encode values 0 up to \(2^n - 1\) into \(n\) bits binary sequences with the property that only one bit changes between any two consecutive sequences. A gray-code binary pattern is a visual representation of the binary sequences at a specific bit. This way uses a handheld camera but it does not require for the camera to capture the entire computer screen. A sequence of binary patterns appears on a computer screen in order then the temporal pattern captured by a handheld camera at a specified spot on the screen uniquely represents its own spot. Those binary patterns are hopefully invisible to users, so that this way requires other techniques such as infrared ones [2].

Our pointing interface uses a common computer mouse. It is inexpensive and does not rely on quadrilaterals of a computer screen, and it also requires no infrared techniques. Generally, a computer mouse is designed to be used just in front of a computer screen, so that the mouse cursor moves as users expect. For a tabletop display, users might stand at the top of the screen. The problem is that the coordinates of the screen (hereafter the screen coordinates) are not always aligned with the coordinates of the mouse control space (hereafter the mouse control coordinates). Wigdor, D. et al. [3] showed that the correlation between display positions and mouse control coordinates would have a significant impact on performance.

In our first attempt to align the screen coordinates with the mouse control coordinates [4], we introduced the basic idea of a reflex in eye-hand coordination and built a proof of concept system of a floor navigation system. This system shows a store map and some graphical buttons on the floor by projection and the user manipulates a tilt sensor (it could be a smartphone or a tablet that is connected to the system by Bluetooth) to move the graphical cursor. The user can press any graphical button by moving the graphical cursor on it and flicking that tilt sensor vertically. In this floor navigation, the system has to locate the user’s position so that he/she is able to manipulate the graphical cursor properly. To do this, the system exploits a reflex in eye-hand coordination. A reflex in eye-hand coordination is a natural response to inconsistency between kinetic information of a tilt sensor and visual feedback of the graphical cursor. This system demonstrated that the reflex depends on the angular distance between the coordinates of the projection image and the co-
ordinates of the tilt sensor’s control space, and the reflex is also viable to align them. The result showed that the reflex takes 4 seconds to complete the alignment.

To improve performance of the alignment in this paper, we take the basic idea of a shadow cursor. A shadow cursor is an invisible mouse cursor and lets users calibrate the screen coordinates less than in a second if they move the cursor in the given direction correctly. Previously we introduced the shadow cursor in [5] and implemented a prototype system of screen coordinates calibration using the shadow cursor in [6], and analyzed properties of its calibration performance in [7]. The aim of this paper is to propose a shadow cursor and build a screen coordinates calibration method, and evaluate its performance.

Section 2 describes related work and Sect. 3 introduces a shadow cursor and describes how it aligns the screen coordinates with the mouse control coordinates. Section 4 conducts a pilot experiment in feasibility of the shadow cursor and Sect. 5 describes implementation of screen coordinates calibration using the shadow cursor and designs an experiment in calibration performance. Section 6 discusses calibration accuracy and time efficiency, and Sect. 7 evaluates our approach to the conventional one. Section 8 gives the concluding remarks.

2. Related Work

There have been a lot of researches on a pointing interface for tabletop displays from various aspects such as target occlusion problem, small target problem and out-of-reach target problem. The target occlusion problem is referred to difficulties with touching a target because it is occluded partly or entirely by the tip of the finger, resulting in low efficiency of pointing. The small target problem is referred to difficulties with touching a smaller target than the size of the fingertip because of ambiguity of the contact point with the touch screen plane. This section summaries these researches and clarify the position of our approach.

B. Amartya et al. [8] built an in-air pointing technique to deal with out-of-reach target problem. Their technique employs perspective-based pointing. The graphical cursor is placed at the intersection of the tabletop display plane and the vector from the nose through the tip of the index finger. The nose and the tip of the index finger are tracked in real time by a Vicon motion capture system at 100Hz with an accuracy of 3mm resolution. This technique demonstrates a good performance in Fitts’s law throughput, which is close to that of a common mouse cursor but it only works with a sophisticated sensor system. Our approach relies on a low-cost common mouse.

A virtual mouse is often focused as an alternative approach. B. Tom et al. [9] used a virtual mouse designed for multi-touch screens, which is called SurfaceMouse, to deal with target occlusion problem and small target problem as well as out-of-reach target problem. They concluded that SurfaceMouse makes users feel less frustrated when clicking on targets because of visual feedbacks of the mouse cursor without any occlusions and accuracy of pointing at small targets even if it was out of the users’ reach in comparison to touch-based interaction. However they pointed that users miss physical feedbacks afforded by a real mouse. Similarly, A. Martha et al. [10] proposed a virtual grabber called iGrabber to deal with out-of-reach target problem. iGrabber enables users to use a virtual grabber to extend the hand and select a target. It is reported that iGrabber is natural and easy to understand concepts. M. Justin et al. [11] explored various configurations of a virtual mouse supporting full mouse functionality like three buttons and scroll wheels in order to compare their performance to that of a real mouse. The result shows that a real mouse still demonstrates the best score in task completion time and error rates. Our approach relies on a real mouse to make the most of performance.

As regards appropriate tasks for touch input and real mouse input, F. Clifton et al. [12] conducted an experiment in performance of unimanual tasks and bimanual ones between direct touch input on a tabletop display and real mouse input on a laptop. The result shows that users benefit from direct touch input for bimanual tasks and they do from real mouse input for unimanual ones. E.H. Thomas and P.H. Juan [13], in addition, showed that untangling a ring tasks by multiple users are efficiently solved by a multi-real mouse single display setup rather than a multi-touch setup. Because tabletop displays are suitable for collaborative work, the result implies that a real mouse is preferred for tabletop displays. Our approach focuses on preference for a real mouse in a collaborative setup.

In the field of touch interaction for mobiles, where target occlusion and small target problems are actively discussed, V. Daniel et al. [14] proposed a touch technique called Shift to deal with target occlusion problem. Shift creates a copy of the occluded screen area and displays the copy nearby in a non-occluded area. Users can move the graphical cursor showed in the copy and click on a target by lifting the finger. Furthermore, Y. Koji et al. [15] proposed a target quick selection technique called Escape to deal with selection speed problem that stems from poor input resolution of finger and thumb selection. Escape assigns every different gesture icon for each target, enabling users to select a target quickly and correctly by performing the assigned gesture. Escape is applicable to a pointing interface for large display environments but it works well for such an environment that not so many targets exist because each different target requires a different gesture icon.

3. A Shadow Cursor

A common computer mouse gives visual feedbacks of the mouse cursor without any occlusions and accuracy of pointing at small targets even if they were out of the users’ reach. So it is a solution to all the three problems of target occlusion problem, small target problem and out-of-reach target problem. A shadow cursor is a way of using a common computer mouse as a pointing interface for tabletop displays.

A shadow cursor is an imaginary mouse cursor. Fig-
Figure 1 shows how it works to align the screen coordinates with the mouse control coordinates. In the figure, there is the angular distance of 30 degrees between the screen coordinates and the mouse control coordinates. First, (1) a vertical arrow to the screen appears on it and the mouse cursor becomes invisible, and (2) the user moves an imaginary mouse cursor (it is hereafter called a shadow cursor) in the direction indicated by that vertical arrow. Then (3) that invisible mouse cursor will actually move at the angle of $x$ degrees. In this case, $x$ equals 30 degrees. In order to obtain the correct angular distance, it is necessary for the user to manipulate the mouse in the same direction as the arrow without seeing any other visual feedbacks.

The next section conducts a pilot experiment on feasibility of a shadow cursor.

4. Pilot Experiment on Feasibility

This section conducts a pilot experiment to investigate how accurately users can move a shadow cursor and shows that the shadow cursor has the potential to align the screen coordinates.

4.1 Design of Experiment

We conduct an experiment on how accurately users can move a mouse without seeing any visual feedbacks. Figure 2 shows design of the experiment. The rectangle represents a laptop screen of Dell XPS M1210, which is 261mm wide and 164mm high (1280x800 pixels in resolution). The laptop has a processor of 1.73GHz. It is also equipped with a graphics card of nVidia GeForce Go 7400. The subject sees the laptop screen right in front.

A radial arrow appears in the center at each angle of $-150$ to $180$ at 30 degrees intervals after the subject clicks on that position and he/she is asked to move the mouse in that direction, and then the system records movement of a shadow cursor. There were 16 subjects between the ages of 21 and 23. All were right-handed, had an experience working with a mouse. Each subject had 10 trials at each angle, resulting in 120 trials.

$$16 \text{ subjects} \times 12 \text{ angles of a radial arrow} \times 10 \text{ trials} = 1920 \text{ total trials}$$

To avoid learning effects, the sequence of trials is in a random order but it is the same between subjects.

4.2 Result

Figure 3 shows effect of angular positions of a radial arrow on accuracy. The horizontal axis shows the angular positions and the vertical one shows angular error, which is sub-

![Fig. 2 Design of a pilot experiment on accuracy of movement of a shadow cursor.](image-url)

![Fig. 3 Effect of angular positions of a radial arrow on accuracy.](image-url)
traction of the observed angular position from the one of a target radial arrow, at each angular position across all the subjects. From the figure, there seems to be a certain relation between angular positions of a radial arrow and angular errors. There are two increases around ±30 to 60 degrees, one subtle decrease in the center, and two drastic decreases at both sides. Figure 4 shows the standard deviation of accuracy. From the figure, accuracy at 0 and ±90 degrees is comparatively high. Analysis of variance shows that there is a significant impact of angular positions of a radial arrow on accuracy \[ F(11,1908) = 4.54074 \text{ at } p = 0.001 \]. The accuracy is however kept within ±4 degrees.

5. Implementation and Experiment

This section mentions implementation of our prototype system of screen coordinates calibration using a shadow cursor and specifies design of an experiment in usability.

5.1 Implementation

Figure 5 explains how to align the screen coordinates with the mouse control coordinates by a shadow cursor. At first, a vertical arrow appears on the current screen coordinates, where the user right-clicks. The user can right-click wherever he/she wants in order to evaluate our system in a practical condition. Then the mouse cursor becomes invisible (because the mouse cursor will move in the direction that the user does not expect, so that the mouse cursor is hopefully set invisible) and the user moves the shadow cursor in the direction indicated by that vertical arrow. After that the angular distance between that vertical arrow and the direction the mouse cursor has actually taken is calculated when the mouse cursor goes out of the circle of 10 mm radius. Finally, the current screen coordinates are rotated by the observed angular distance.

5.2 Design of Experiment

Figure 6 shows design of an experiment in usability of a shadow cursor. In this experiment, a laptop is adopted instead of a tabletop display without losing generality. The rectangle represents a laptop screen of Dell XPS L702X, which is 385mm wide and 215mm high (1600x900 pixels in resolution). This laptop has a quad-core processor of 2.40GHz and it is also equipped with a graphics card of nVidia GT 555M. The angle of the laptop screen from the surface of a table is 48 degrees so that the subject’s eye rays are perpendicular to the laptop screen if he/she sits right in front of the laptop. The subject has to see the laptop screen with the various angular positions. It means that it is difficult for the subject to find the mouse cursor from the side of the screen and gives a negative effect on the experiment results. It results in the situation that the physical relation between the laptop screen and the subject is fixed and the screen coordinates change on the laptop screen, to eliminate the negative effect. This laptop screen coordinates are set in the angular position of −150 to 180 at 30 degree intervals (it is hereafter called an initial angular position) for subjects to start performing screen coordinates calibration. There were 10 subjects between the ages of 22 and 36. All were right-
handed and had an experience working with a mouse. Each subject had 10 trials at each initial angular position, resulting in 1200 trials:

10 subjects \times 12 \text{ angular positions} \times 10 \text{ trials} = 1200 \text{ total trials} \hspace{1cm} (2)

To avoid learning effects, the sequence of trials is in random order but it is the same between subjects.

6. Calibration Performance and Its Properties

This section describes calibration performance of our prototype system in terms of calibration accuracy and time efficiency.

6.1 Calibration Accuracy

Figure 7 shows a result on calibration accuracy. The horizontal axis shows initial angular positions of screen coordinates and the vertical one shows angular error, which is the subtraction of the observed angular distance from the actual angular distance, at each initial angular position across all the subjects. From the figure, there seems to be a certain positive correlation between the initial angular position and angular error. A regression line for those data points is:

\[ y = 0.0159 \times x - 1.43 \] \hspace{1cm} (3)

where \( x \) denotes the initial angular position in degree and \( y \) does angular error in degree. The regression line shows that the gradient is positive and an increase of 1 degree to the initial angular position has the potential of adding another angular error of 0.0159 degrees. It also shows that the angular error becomes zero when the initial angular position is 89.94 degrees. The positive gradient and the initial angular position with no angular error could be because all the subjects are right-handed. The gradient and the initial angular position with no angular error could be negative and -90 degrees, respectively, for left-handed subjects. A further experiment is required to confirm it. Regression analysis shows that the regression is significant \( t(1198)=3.85, p<.001 \).

The gradient of this regression line is considerably large, so there could be a certain impact of the counter clockwise initial angular positions (-150 to -30 degrees) and the clockwise ones (30 to 150 degrees) on calibration accuracy. Figure 8 shows angular error at those two groups of initial angular positions on average. The standard deviation for counter clockwise initial angular positions is 14.15 degrees and it is 13.35 degrees for clockwise ones. They are almost the same with each other but they are much greater than the angular error. A significant impact of the counter clockwise and the clockwise initial angular positions on calibration accuracy was not confirmed. The average angular error across all the initial angular positions is -1.19 degrees.

Figure 9 shows standard deviation of angular errors at each initial angular position. From the figure, there are four significant drops at -90, 0, 90 and 180 degrees. Because people tend to recognize horizontal lines and vertical ones easily as compared with other angular lines, they could manipulate the mouse along those lines accurately. Referring to Fig. 7 and Fig. 9, it is expected that at the initial angular position of -90, 0, 90 and 180 all the angular errors are

![Fig. 6](image_url)  
*Fig. 6*  
Experiment settings in usability of the shadow cursor.

![Fig. 7](image_url)  
*Fig. 7*  
Result of calibration accuracy.

![Fig. 8](image_url)  
*Fig. 8*  
Calibration accuracy at the counter clockwise and the clockwise initial angular positions.
plotted near zero. To confirm it, Fig. 10 shows absolute angular error, which is the absolute value of subtraction of the observed angular distance from the actual angular distance. From the figure, it is confirmed that people manipulate the mouse more accurately and precisely at the initial angular position of $-90, 0, 90$ and $180$ degrees than any other initial angular positions $[F(1,1198)=69.36$ at $p<.001]$. Note that the absolute angular errors are still roughly 4 to 11 degrees on average and the standard deviation is over roughly 10 degrees. So it may cause serious trouble during using the mouse on some applications. A further experiment is required to obtain the acceptable range of angular errors.

Figure 11 shows angular error and initial angular positions together in a form of pie chart. The radial solid line represents each initial angular position and the arrow represents the corresponding angular error. From the figure, the arrowhead seems to have a pattern in direction. For the initial angular position of $0$ to $120$ degrees, the arrowhead points in clockwise direction (the angular error is negative) and the arrowhead points in counter clockwise direction (it is positive) for the initial angular position of $150$ to $180$ degrees. All the arrowheads point from a certain angular position (the white dot) to another (the black dot). This could be because all the subjects are right-handed, so that they are more likely to move their arms in the direction of $15$ or $165$ degrees and not likely to do in the direction of $-45$ or $135$ degrees.

6.2 Time Efficiency

Figure 12 shows a result on time efficiency. The horizontal axis shows initial angular positions and the vertical one shows the elapsed time to complete screen coordinates calibration across all the subjects. There seems to be a certain change in elapsed time between the counter clockwise initial angular positions and clockwise ones. Figure 13 shows the elapsed time at those two groups of initial angular positions on average. The standard deviation for the counter clockwise initial angular positions is 403.3 milliseconds and it is 258.7 milliseconds for the clockwise ones. A significant impact of those two groups of initial angular positions on time efficiency was however not confirmed. The average elapsed time across all the initial angular positions is 665.2 milliseconds.

To see relationship between calibration accuracy and time efficiency, Fig. 14 shows their scatter diagram. The horizontal axis shows the elapsed time and the vertical one shows the angular error. From the figure, there seems to be a certain correlation between them. Correlation analysis shows that the correlation coefficient is $-0.14$. It confirms that there is almost no correlation between them.
7. Evaluation

This section describes strengths and weaknesses of the use of a computer mouse as a pointing interface for tabletop displays in comparison with the use of touch with fingers and gives evaluation of our prototype system to the latest work of screen coordinates calibration, and specifies improvement in calibration performance.

7.1 Strengths of Using a Computer Mouse

The use of a common computer mouse as a pointing interface for tabletop displays is tolerant of target occlusion problem, small target problem and out-of-reach target problem introduced in Sect. 2 because the mouse cursor gives visual feedbacks without any occlusions and accuracy of pointing at small targets even if they were out of the users’ reach. These three problems stem from the use of touch with fingers on the screen as a pointing interface seen in [8]–[10], [14], [15]. The use of the computer mouse has strength of giving a solution to these problems but weakness of requiring screen coordinates calibration. Our approach relies on the use of a common computer mouse.

To perform screen coordinates calibration, it is desired to be done as accurately and quickly as possible. In this section, our approach is evaluated in comparison with Reflex [4], which is the latest work of the screen coordinates calibration. The reflex exploits a natural response to inconsistency between kinetic information of the mouse and visual feedbacks of the mouse cursor to acquire the angular distance between the screen coordinates and the mouse control coordinates.

7.2 Calibration Accuracy

Figure 15 compares our prototype system Shadow to the conventional one Reflex in calibration accuracy. The horizontal axis shows the two systems and the vertical one shows angular errors on average for each system. Here data values at initial angular positions of zero and 180 degrees for Shadow are excluded for comparison with Reflex. From the figure, their angular errors are almost the same with each other, which are 4.08 and −1.56 for Reflex and Shadow, respectively, but the standard deviation of Shadow is shrunk to about half the one of Reflex. Unpaired t test shows that there is a significant impact of the two systems on calibration accuracy [t(1248)=4.01 at p<.001].

To see the impact in detail, Fig. 16 shows calibration
accuracy for Reflex and Shadow at each initial angular position. For Reflex, the angular error is comparatively large as the initial angular position is away from zero. On the other hand, the angular error is nearly constant through all the initial angular positions for Shadow. It is fairly to say that Shadow works well without any considerable angular errors independently of initial angular positions. Figure 17 shows improvement of calibration accuracy between Reflex and Shadow at both the counter clockwise and clockwise initial angular positions. Unpaired t test shows that there is a significant impact of the two systems on calibration accuracy for the clockwise initial angular positions $[t(623)=8.61$ at $p<.001]$.

Consequently, our prototype system improves degree of calibration accuracy by 61.8% to 1.56 from 4.08 degrees on average.

7.3 Time Efficiency

Figure 18 compares our prototype system Shadow to the conventional one Reflex in time efficiency. The horizontal axis shows the two systems and the vertical one shows elapsed time on average for each system. Here data values from initial angular positions of zero and 180 degrees for Shadow are excluded for comparison with Reflex. From the figure, both the elapsed time and standard deviation for Shadow are less than half the ones for Reflex. Unpaired t test shows that there is a significant impact of the two systems on time efficiency $[t(1248)=28.4$ at $p<.001]$.

To see the impact in detail, Fig. 19 shows time efficiency for Reflex and Shadow at each initial angular position. The elapsed time for Reflex is almost constant of 2000 milliseconds over all the initial angular positions and the one for Shadow is almost constant of 500 milliseconds as well. However the standard deviation of Shadow is about three times smaller than the one of Reflex. It is fairly to say that Shadow works well without any considerable delays over all the initial angular positions. Figure 20 shows improvement of time efficiency between Reflex and Shadow at both the counter clockwise and clockwise initial angular positions.

Unpaired t test shows that there is a significant impact of the two systems on time efficiency for both the counter clockwise $[t(623)=34.5$ at $p<.001]$ and clockwise initial angular positions $[t(623)=27.5$ at $p<.001]$.

Consequently, our prototype system improves time efficiency by 66.6% to 666.1 from 1992.2 milliseconds on average.
8. Conclusions

This paper introduced a screen coordinates calibration technique using a shadow cursor for the use of a common mouse in a tabletop display environment. The shadow cursor is a concept of a mouse cursor without any visual feedbacks. This paper built a prototype system of screen coordinates calibration using a shadow cursor and conducted an experiment in calibration performance. As the result, it was shown that a shadow cursor can perform screen coordinates calibration 2.62 times more precisely and 2.99 times more quickly than the conventional approach.

References

[1] J.C. Lee, P.H. Dietz, D. Maynes-Aminzade, R. Raskar, and S.E. Hudson, “Automatic projector calibration with embedding light sensors,” Proc. UIST2004, pp.123–126, ACM, Oct. 2004.
[2] J.C. Lee, S.E. Hudson, J.W. Summet, and P.H. Dietz, “Moveable interactive projected displays using projector based tracking,” Proc. UIST2005, pp.63–72, ACM, Oct. 2005.
[3] D. Wigdor, C. Shen, C. Forlines, and R. Balakrishnan, “Effects of display position and control space orientation on user preference and performance,” Proc. SIGCHI conference on Human Factors in computing systems, CHI ’06, New York, NY, USA, pp.309–318, ACM Press, 2006.
[4] M. Ishihara and Y. Ishihara, “Calibrating coordinates of a tabletop display with a reflex in eye-hand coordination,” IEICE Trans. Inf. & Syst., vol.E93-D, no.10, pp.2862–2865, 2010.
[5] M. Ishihara and Y. Ishihara, HCI International 2011 – Posters’ Extended Abstracts: International Conference, HCI International 2011, Orlando, FL, USA, July 9-14, 2011, Proceedings, Part II, ch. Properties of Shadow-Cursor for Calibrating Screen Coordinates of Tabletop Displays, vol.174, pp.242–246, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.
[6] M. Ishihara and Y. Ishihara, HCI International 2013 - Posters’ Extended Abstracts: International Conference, HCI International 2013, Las Vegas, NV, USA, July 21-26, 2013, Proceedings, Part I, ch. Calibrating Screen Coordinates of Tabletop Display Using Shadow-Cursor, vol.373, pp.327–331, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
[7] M. Ishihara and Y. Ishihara, “Calibrating screen coordinates of tabletop display using a shadow cursor and its properties,” Proc. 2016 Joint 8th International Conference on Soft Computing and Intelligent Systems (SCIS) and 17th International Symposium on Advanced Intelligent Systems (ISIS), N, pp.796–800, IEEE, 2016.
[8] A. Banerjee, J. Burstyn, A. Giroud, and R. Vertegaal, “Pointable: An in-air pointing technique to manipulate out-of-reach targets on tabletops,” Proc. ACM International Conference on Interactive Tabletops and Surfaces, ITS ’11, New York, NY, USA, pp.11–20, ACM, 2011.
[9] T. Bartindale, C. Harrison, P. Olivier, and S.E. Hudson, “SurfaceMouse: Suplementing multi-touch interaction with a virtual mouse,” Proc. Fifth International Conference on Tangible, Embedded, and Embodied Interaction, TEI ’11, New York, NY, USA, pp.293–296, ACM, 2011.
[10] M. Abednego, J.-H. Lee, W. Moon, and J.-H. Park, “I-Grabber: Expanding physical reach in a large-display tabletop environment through the use of a virtual grabber,” Proc. ACM International Conference on Interactive Tabletops and Surfaces, ITS ’09, New York, NY, USA, pp.61–64, ACM, 2009.
[11] J. Matejka, T. Grossman, J. Lo, and G. Fitzmaurice, “The design and evaluation of multi-finger mouse emulation techniques,” Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI ’09, New York, NY, USA, pp.1073–1082, ACM, 2009.
[12] C. Forlines, D. Wigdor, C. Shen, and R. Balakrishnan, “Direct-touch vs. mouse input for tabletop displays,” Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI ’07, New York, NY, USA, pp.647–656, ACM, 2007.
[13] T.E. Hansen and J.P. Hourcade, “Comparing multi-touch tabletops and multi-mouse single-display groupware setups,” Proc. 3rd Mexican Workshop on Human Computer Interaction, MexIHC ’10, San Luis Potosi Mexico, S.L.P. México, pp.36–43, Universidad Politécnica de San Luis Potosi, 2010.
[14] D. Vogel and P. Baudisch, “Shift: A technique for operating pen-based interfaces using touch,” Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI ’07, New York, NY, USA, pp.657–666, ACM, 2007.
[15] K. Yatani, K. Partridge, M. Bern, and M.W. Newman, “Escape: A target selection technique using visually-cued gestures,” Proc. SIGCHI Conference on Human Factors in Computing Systems, CHI ’08, New York, NY, USA, pp.285–294, ACM, 2008.

Makio Ishihara received his B.E., M.E., and Ph.D. from Yamaguchi University, Japan, in 1998, 2000, and 2004, respectively. He is currently an Associate Professor in the Faculty of Information Engineering, Fukuoka Institute of Technology, Fukuoka, Japan. His research interests include gamification, augmented reality and human computer interaction. He is a member of the Institute of Electronics, Information and Communication Engineers of Japan (IEICE) and a member of the Institute of Electrical and Electronics Engineers (IEEE).

Yukio Ishihara received his B.E., M.E., and Ph.D. from Yamaguchi University, Japan, in 1998, 2000, and 2003, respectively. He currently belongs to Shimane University as an assistant professor, and seeks relationship between domestic power consumption and health state. His research interests include ICT health promotion, 3D presentation and interaction techniques in computer-generated environments. He is also interested in a mechanism of human eye movements during reading. He is a member of the Association for Computing Machinery (ACM).