HandyScope: A Remote Control Technique using Circular Widget on Tabletops

Takuro Kuribara, Yusaku Mita, Kazusa Onishi,
Buntarou Shizuki, and Jiro Tanaka

University of Tsukuba, Japan
{kuribara,mita,onishi,shizuki,jiro}@iplab.cs.tsukuba.ac.jp

Abstract. A large multi-touch tabletop has remote areas that the users might not touch by their hands. This forces users to move around the tabletop. In this paper, we present a novel remote control technique which we call HandyScope. This technique allows users to manipulate those remote areas. Moreover, users can move an object between the nearby area and the remote areas using a widget. In addition, users can precisely point a remote area quickly because this system includes our proposed control-display ratio changing system. To evaluate the performance of HandyScope, we compared HandyScope with direct touch operation. The results show that HandyScope is significantly faster in selection.

Keywords: bimanual interaction, multi-touch, gesture, dynamic control-display gain, pointing, target acquisition, pull-out.

1 Introduction

A large multi-touch tabletop allows users to surround the tabletop and touch the tabletop from their respective positions. However, it has remote areas that users might not touch by their hands; for example, touching a distant object displayed on the opposite side of the tabletop is difficult due to the large size of the touch screen. This forces users to move around the tabletop.

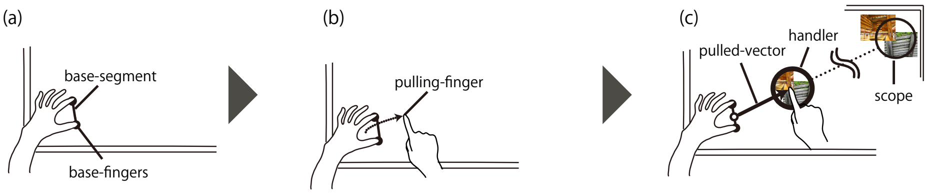

Fig. 1. HandyScope allows users to point and manipulate the remote area. a) When users put two fingers, and b) drag their finger to cross the segment between the two fingers, then c) HandyScope is activated.

To solve this problem, we present a novel remote control technique which we call HandyScope (Figure 1). This technique allows users to manipulate remote areas (e.g., move, rotate, and resize distant objects) and move an object between the nearby area and the remote areas. In addition, users can precisely point a remote area quickly by using the widget because this system includes the control-display (C-D) ratio changing system which we have already proposed [21].

2 Related Work

Remote pointing techniques have been intensively investigated to facilitate especially pointing on large wall displays. Such techniques are device-based pointing [6, 14], gesture-based pointing [19], and gaze-based pointing [8]. In contrast, our technique allows users to point remote areas on tabletops, which adopt a bimanual gesture. Therefore, we focus on pointing techniques for tabletops and studies of bimanual interaction.

Pointing Techniques for Tabletops

Parker et al. used the stylus tip's shadow to point at a remote position [15]. In the work of Banerjee et al. [3], users could point at a remote position on tabletops and dynamically change C-D ratio using one hand while performing a pointing operation with the other hand. The above techniques required additional devices that obtain the position of users' hands to realize direct-pointing. Bartindale et al. [5] developed an onscreen mouse for multi-touch tabletops that allows users to point at a remote position, similar to a conventional physical mouse. However, this technique required to use tabletops that allow for a measurement of the area of hand's contact. In contrast, our technique can be applied to tabletops that detect multi-touch points without additional devices and recognizing the shape of hands. Matejka et al. [13] also developed an onscreen mouse, while activation method is still open.

I-Grabber [1] is an onscreen widget controlled by bimanual multi-touch interactions. Our technique is also controlled by using bimanual interaction. However, our technique allows users to change the C-D ratio and to use only a single multi-touch gesture from activation to pointing. Therefore, users can point precisely and quickly.

Bimanual Interaction

There was some research on bimanual interaction such as 3D operation [16, 20], modeling [2, 10], and precise selection [7] . In contrast, our technique allows users to point remote areas using bimanual interaction.

Tokoro et al. presented a pointing technique that utilized two acceleration sensors, and postures of both hands pointing at a remote position [17]. Furthermore, Malik et al. developed a bimanual pointing technique by using image processing [12]. In contrast with these techniques, our technique is performed by using touch based gestures.

3 HandyScope

HandyScope allows users to manipulate remote areas using a circular widget. The widget is composed of two parts, a handler and a scope. The handler is sent to remote areas to select an area manipulated; the scope is used to manipulate the remote area by users. The scope area is displayed in the handler; and all events onto the handler area are sent to the scope area. Therefore, users can manipulate (e.g., move, rotate, and resize) the remote objects within the scope, using the handler. Moreover, this technique uses pull-out, a bimanual multitouch gesture [22]. This gesture allows multiple users to, without conflicting with other touch gestures, simultaneously manipulate remote areas. Below we describe the interaction of HandyScope and its advantages.

3.1 Activation and Control Technique

Figure 1 shows the procedure of HandyScope. Users put two fingers of their non-dominant hand (base-fingers) on a tabletop as shown in Figure 1a. When users drag a finger of their dominant hand (pulling-finger) to cross the segment between the base-fingers (base-segment) as shown in Figure 1b, a circle (scope) is displayed on the ray between the midpoint of the base-segment; another circle (handler) is displayed around the pulling-finger as shown in Figure 1c. If users arrange the pulled-vector, the scope position is updated according to the change. Users can quit control anytime by detaching both of the base-fingers from the tabletop.

3.2 Deciding the Position of Scope with Dynamic C-D ratio

Suppose that $P_i(x, y)$ is the i-th scope position after i frames have passed since users placed their base-fingers on the tabletop as shown in Figure 2. Then P_i is given by the following formulas:

$$\begin{aligned} P_i &= G_0 + \sum_{j}^{i} k_j \mathit{\Delta} V_j, \\ \mathit{\Delta} V_i &= V_i - V_{i-1}, \\ k_i &= \alpha \times \frac{|S_i|}{|S_0|}. \end{aligned} \tag{1}$$

S_0 is the length of base-segment when base-fingers were placed on the tabletop, S_i is the length of i-th base-segment, and G_0 is the midpoint of base-segment. Furthermore, V_i is the pulled-vector from G_i to the pulling-finger respectively. α is a constant. That is, our technique determines the i-th C-D ratio by k_i. When users pinch out the base-fingers, k_i shows that C-D ratio becomes large. Similarly, when users pinch in the base-fingers, k_i becomes small.

Therefore, P_i is dependent on k_i and $\mathit{\Delta} V_i$ (the difference of V_i) caused by moving dominant or non-dominant hand. P_i is calculated on each frame.

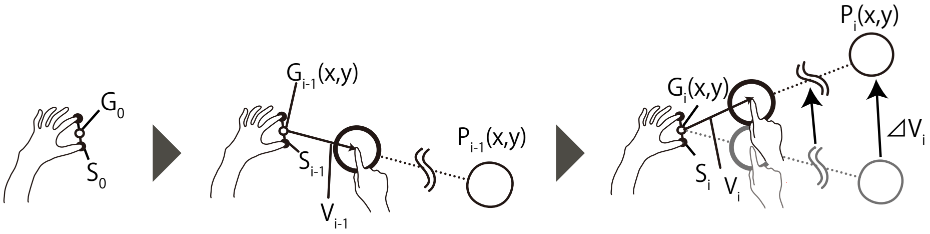

Fig. 2. Moving the circular widget using a simple gesture.

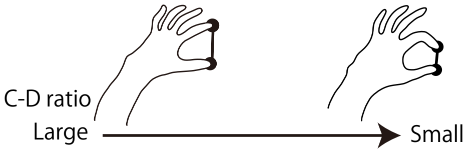

Fig. 3. Dynamic C-D ratio according to the length of base-segment.

C-D ratio changes depending on the length of the base-segment. Figure 3 shows the relation between C-D ratio and the length of base-segment. Users can selectively perform rough control with a large C-D ratio or precise control with a small C-D ratio, because they can simultaneously point while controlling C-D ratio. For example, users can move scope roughly and quickly with large C-D ratio, then they can move the scope precisely and slowly with a small C-D ratio as shown in Figure 4.

3.3 Remote Manipulation using the Widget

Users can manipulate the remote objects using the handler such as resizing (Figure 5a) and rotating (Figure 5b). To achieve this, the scope area is displayed in the handler and all events onto the handler area are sent to the scope area. Therefore, users can manipulate remote objects without walking to remote areas or bringing remote objects to nearby area.

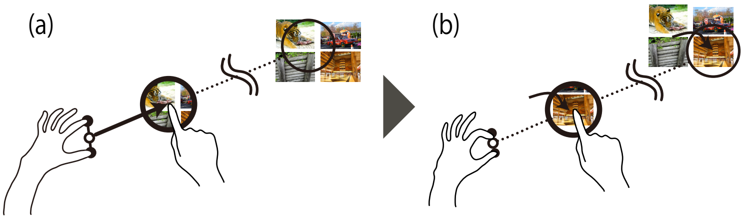

Fig. 4. Usage of dynamic C-D ratio. Users a) point at position far quickly with large C-D ratio, and then b) precisely point at object with small C-D ratio.

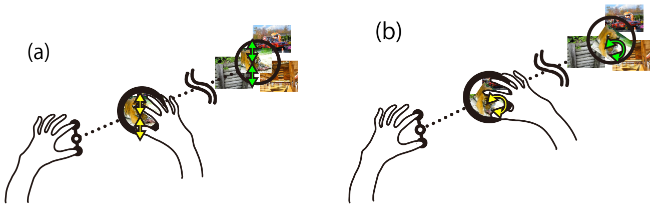

Fig. 5. Manipulating the remote objects from nearby area. a) Resizing the remote objects, b) rotating the remote objects.

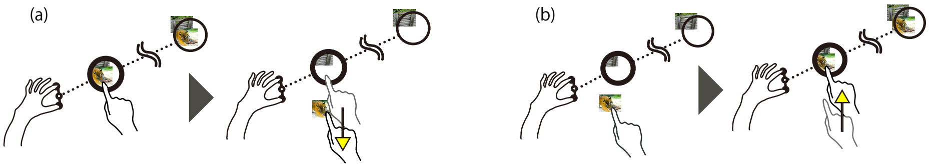

Fig. 6. Transferring the objects between a nearby area and remote area. a) From remote area to nearby area. b) From nearby area to remote area.

3.4 Transferring the Objects Between Nearby and Remote Area

If users select a remote object in the handler and drag it outside the handler, the remote object is transferred to the nearby area as shown in Figure 6a. Correspondingly, if users select a nearby object and drag it into the handler, the nearby object is transferred to the remote area as shown in Figure 6b. In this way, users can transfer the objects quickly between the nearby and the remote area.

3.5 Adjusting the Widget

When users activate HandyScope, the widget will be displayed around the pulling-finger. But once users detach the pulling-finger from the tabletop, all events onto the handler area are sent to the scope area. Therefore, to move the circular widget again, users have to drag the edge of the handler as shown in Figure 7. In this way, users can adjust the widget.

3.6 Resizing the Widget

By pinching in and out on the edge of the handler, users can resize the circular widget. Figure 8 shows resizing the circular widget. In this way, users can manipulate the large objects at the remote areas.

3.7 The Advantages of HandyScope

In this section, we discuss the advantages of HandyScope. This technique allows users to change C-D ratio using two fingers of their non-dominant hand. Users

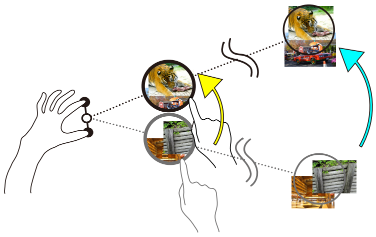

Fig. 7. Moving the circular widget again.

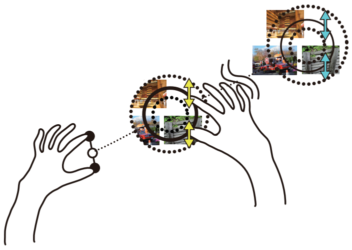

Fig. 8. Resizing the circular widget.

select the pointing area by the relative position between their two fingers and a finger of their dominant hand. Users can interact with the remote area through the widget. In this way, users can manipulate the remote area similar to Frisbee [11] or Dynamic Portals [18]. However, Frisbee requires users to determine the remote area in advance and Dynamic Portals needs collaborator(s) to select the remote area. In contrast, HandyScope allows users to activate it from any position and decide the remote area quickly by dynamically changing C-D ratio.

4 Evaluation

We conducted experiments to measure the performance of HandyScope. In this experiment, we compared HandyScope (HandyScope condition) with existing direct touch (Touch condition) by typical three operations on tabletop. These three operations were Selecting, Rotating and Resizing.

4.1 Participants and Evaluation environment

Ten undergraduate and graduate students with ages ranging from 20 to 24 years participated in this experiment. 1 of them was left-handed. All of them had never used HandyScope. We showed the evaluation environment in Figure 9. We used a 1470 mm × 80 mm 60-inch display (PDP-607CMX[1]) with multi touch function by attaching Multi touch flame (PQ Lab, Multi-Touch G^{3} [2]). We set the height of tabletop to 93 cm. The height of tabletop were between 91 cm to 105 cm in some studies about tabletop [4, 9, 23], so our 93 cm fits this range. And we fixed α of (1) in Section 3.2, as participants do not need to change C-D ratio when they point the edge of tabletop.

4.2 Task

We asked the participants to perform Selecting task, Rotating task and Resizing task by rotation. These tasks were following the study proposed in the pointing

[1] http://pioneer.jp/biz/karte/PDP-607CMX.html

[2] http://multi-touch-screen.com/product_g3.html

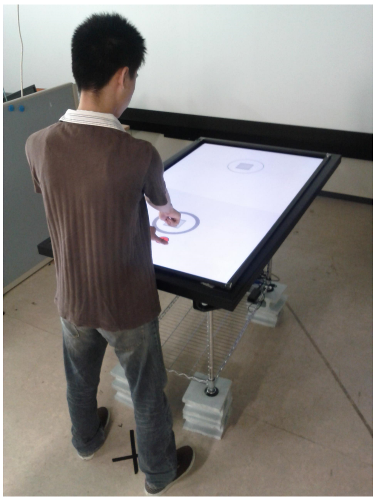

Fig. 9. Experimental environment.

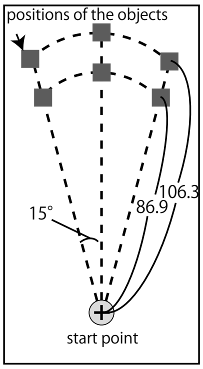

Fig. 10. Positions of start point and target objects.

technique on tabletop [3]. We asked them to complete a practice task before performing the real one. The amount of the practice task is set to be 1/4 of the task. We divided participants into two groups to cancel out order effect between technique conditions. One group performed the Touch condition first, and the other performed HandyScope condition first. Participants could use each hands freely in this experiment. We asked them to answer a questionnaire after having finished all tasks. Experiment time was one and a half hour per one participant, then we pay the rewards.

4.3 Selecting Task

We asked participants to select a target object displayed at any position. First, participants stand at the center of one short side of the tabletop (the spot marked by black tape as shown in Figure 9) before each trial. From this position, participants selected a target object displayed at any position. We show display position of a start point and a target object displayed on tabletop in Figure 10. Start point and target object were displayed before each trial.

In HandyScope condition, the participant started the Selecting task by starting HandyScope on a start point. Then, he moved the scope to a target object, and tapped it. When a target object was tapped, the current trial was completed and a beep was played. In Touch condition, the participant started the Selecting task by tapping a start point. Then, he moved to the position where he could touch the target object, and tapped the target object.

In this experiment, the independent variables were: target distance (900 and 1100 pixels, i.e. approximately 922 mm and 1127 mm respectively), target angle (-15, 0, and 15 degree), target size (40, 60, and 80 pixels, i.e. approximately 41

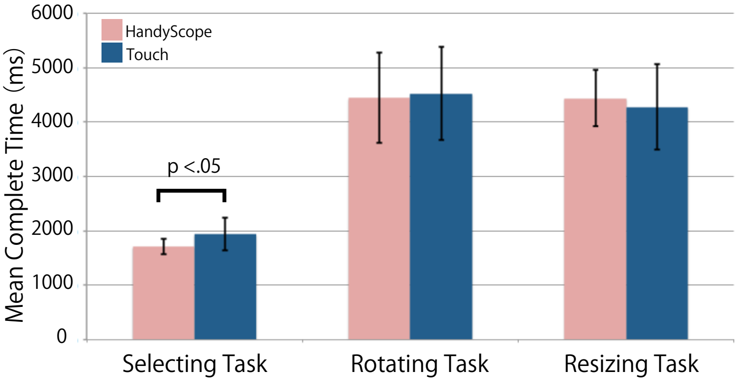

Fig. 11. Mean of the trial-times for each task.

mm, 61 mm, and 82 mm, respectively), and technique (HandyScope and direct touch).

Each participant performed 3 trials per each combination of factors, thus 108 ($2 \times 3 \times 3 \times 2 \times 3$) trials were performed in total. Independent variables for each technique were presented in randomized order.

Results We measured the time to complete a trial (trial-time). The left graphs in Figure 11 show the mean of the trial-times with each technique. The mean time was 1942.6 ms in Touch condition, and was 1715.2 ms in HandyScope condition. And, the result of t-test between mean complete time per participants was t(9)=2.72, p=.011<.050. This result shows HandyScope condition was significantly fast.

4.4 Rotating task

We asked participants to rotate an object to fit the model dock displayed at any position. Display positions of the dock and the objects were the same, while their respective display angles were not. The start point, display positions and the action to start the task were the same as those of Selecting task.

In HandyScope condition, the participant rotated an object to fit the model dock by HandyScope. If the angles of the object and the dock were the same (accident error is within ± 5 degree), the color of the object's border became red. In this condition, when participant finished manipulation, then one trial was completed and a beep was played. In Touch condition, he moved to a position where he could touch the target object, and then rotated the target object.

In this experiment, independent variables were: target distance (900 and 1100 pixels, i.e. approximately 922 mm and 1127 mm respectively), target angle (-15, 0, and 15 degree), dock size (60 and 80 pixels, i.e. approximately 61 mm and 82 mm respectively), rotate angle (-45 and 45 degree), and technique (HandyScope and direct touch). Each participant performed 2 trials per each combination of factors, thus 192 ($2 \times 3 \times 2 \times 2 \times 2 \times 2$) trials were performed in total. Independent variables for each technique were presented in randomized order.

Results The middle graphs in Figure 11 show the mean of the trial-times with each technique. The mean time was 4520.4 ms in Touch condition, and 4443.5 ms in HandyScope condition. And, the result of t-test between mean complete time per participants was t(9)=.267, p=.397>.050. There was no significant difference of time between each technique.

4.5 Resizing task

We asked participants to resize an object and to fit the model dock displayed at any position. Display positions of the dock and the objects were the same, while their respective display size were not. The start point, display position and the action to start the task were same of these of Selecting task.

In HandyScope condition, the participant resized an object to fit the model dock by HandyScope. If the sizes of the object and the dock were same (accident error is within ± 5 pixel), the color of the object's border became red. In this condition, when participant finished the manipulation, then one trial is completed and a beep was played. In Touch condition, he moved to a position where he could touch the target object, and then resized a target object.

In this experiment, independent variables were: target distance (900 and 1100 pixels, i.e. approximately 922 mm and 1127 mm respectively), target angle (-15, 0, and 15 degree), dock size (60 and 80 pixels, i.e. approximately 61 mm and 82 mm respectively), resize operation (expanding and decreasing), and technique (HandyScope and direct touch). Each participants performed 2 trials per each combination of factors, thus 192 ($2 \times 3 \times 2 \times 2 \times 2 \times 2$) trials were performed in total. Independent variables for each technique were presented in randomized order.

Results the right graphs in Figure 11 show the mean of the trial-times with each technique. The mean time was 4277.9 ms in Touch condition, and 4438.2 ms in HandyScope condition. And, the result of t-test between average trial-time per participants was t(9)=-.935, p=.187>.050. There was no significant difference of these time between each technique.

4.6 Consideration

The mean of the trial-times in HandyScope condition was significantly fast in Selecting task. However, there was no significant difference of time between techniques in Rotating task and Resizing task. From these results, HandyScope is considered to be useful for selecting a remote area.

There was no significant difference with techniques in Rotating task and Resizing task. The possible cause of this derives from the fact that restarting HandyScope took time.

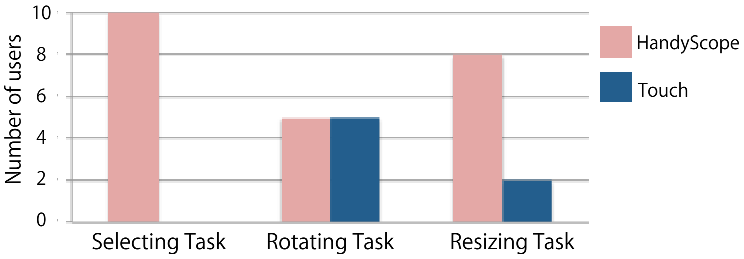

Fig. 12. Questionnaire of preferred technique.

In this experiment, there were situations where participants detached their fingers before finishing the task. In this case, they needed a little time to restart HandyScope to manipulate again. In contrast, in Touch condition, they needed little time to manipulate again because the participant had already moved near the target object. Because of this, we considered that HandyScope took time to Rotating task and Resizing task.

4.7 Questionnaire

Figure 12 shows the results of questionnaire of favorite technique by task.

In Selecting task, all of participants preferred HandyScope. In addition, in Resizing task, eight of participants preferred HandyScope. As reason of these results is that all of these participants said that they could manipulate remote objects without moving, by using HandyScope.

In Rotating task, five of the participants preferred HandyScope. Another five participants preferred direct touch. Two of the participants said that they prefer direct touch because they could use both hands. In addition, two of the participants said that they had some trouble keeping on riding base-fingers on tabletop. One of participants said that it had some trouble restarting HandyScope when he missed the task.

In Resizing task, two of the participants who preferred direct touch said that they had some trouble keeping their base-fingers on the tabletop.

5 Discussion

To investigate whether multiple users simultaneously manipulate remote areas without conflict with other touch gesture using HandyScope, we conducted collaborative task which is the arranging cluttered photos as shown in Figure 13.

In this task, twenty photos were displayed on the tabletop. The size, angle, and location of the photos were random. We arranged the photos cooperating with each other. We stood around the tabletop and did not walk. If we could touch the photos, we manipulated the photos using direct touch. In contrast, if we could not touch the photos, we manipulated the photos using HandyScope. we continued this task five times.

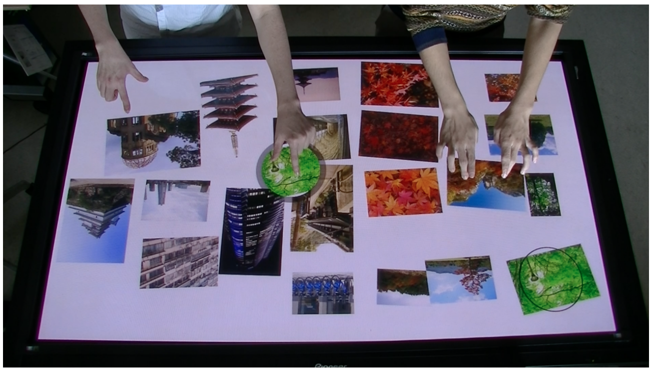

Fig. 13. Collaborative work of multiple users.

As a result of this task, we could not cause accidental activation of HandyScope. Therefore, HandyScope has less incidence of conflicting with other touch gesture. As future work, we would like to perform a detailed evaluation of collaborative work using HandyScope.

6 Conclusion

We designed and implemented a remote control technique, HandyScope. The technique allows users to manipulate remote areas that users might not touch with their hands. In addition, users can move an object between the remote areas and a nearby area using a widget. The user studies using the prototype revealed that HandyScope is a useful technique for selecting a remote object. Moreover, the questionnaire results showed that HandyScope is liked by users. In our future work, we intend to investigate the performance of transferring the objects using HandyScope. Moreover, we would like to use HandyScope on large wall multi-touch displays.

References

1. Abednego, M., Lee, J.H., Moon, W., Park, J.H.: I-Grabber: Expanding physical reach in a large-display tabletop environment through the use of a virtual grabber. In: Proc of ITS '09, pp. 61–64 (2009)
2. Balakrishnan, R., Fitzmaurice, G., Kurtenbach, G., Buxton, W.: Digital tape drawing. In: Proc. of UIST '99, pp. 161–169 (1999).
3. Banerjee, A., Burstyn, J., Girouard, A., Vertegaal, R.: Pointable: An in-air pointing technique to manipulate out-of-reach targets on tabletops. In: Proc. of ITS '11, pp. 11–20 (2011)
4. Banovic, N., Li, F.C.Y., Dearman, D., Yatani, K., Truong, K.N.: Design of unimanual multi-finger pie menu interaction. In: Proc. of ITS '11, pp. 120–129 (2011)
5. Bartindale, T., Harrison, C., Olivier, P., Hudson, S.E.: SurfaceMouse: Supplementing multi-touch interaction with a virtual mouse. In: Proc. of TEI '11, pp. 293–296 (2011)

6. Baudisch, P., Sinclair, M., Wilson, A.: Soap: A pointing device that works in mid-air. In: Proc. of UIST '06, pp. 43–46 (2006)
7. Benko, H., Wilson, A.D., Baudisch, P.: Precise selection techniques for multi-touch screens. In: Proc. of CHI '06, pp. 1263–1272 (2006)
8. Bolt, R.A.: Gaze-orchestrated dynamic windows. In: Proc. of SIGGRAPH '81, pp. 109–119 (1981)
9. Furumi, G., Sakamoto, D., Igarashi, T.: SnapRail: A tabletop user interface widget for addressing occlusion by physical objects. In: Proc. of ITS '12, pp. 193–196 (2012)
10. Grossman, T., Balakrishnan, R., Kurtenbach, G., Fitzmaurice, G., Khan, A., Buxton, B.: Creating principal 3D curves with digital tape drawing. In: Proc. of CHI '02, pp. 121–128 (2002)
11. Khan, A., Fitzmaurice, G., Almeida, D., Burtnyk, N., Kurtenbach, G.: A remote control interface for large displays. In: Proc. of UIST '04, pp. 127–136 (2004)
12. Malik, S., Ranjan, A., Balakrishnan, R.: Interacting with large displays from a distance with vision-tracked multi-finger gestural input. In: Proc. of UIST '05, pp. 43–52 (2005)
13. Matejka, J., Grossman, T., Lo, J., Fitzmaurice, G.: The design and evaluation of multi-finger mouse emulation techniques. In: Proc. of CHI '09, pp. 1073–1082 (2009)
14. Myers, B.A., Bhatnagar, R., Nichols, J., Peck, C.H., Kong, D., Miller, R., Long, A.C.: Interacting at a distance: Measuring the performance of laser pointers and other devices. In: Proc. of CHI '02, pp. 33–40 (2002)
15. Parker, J.K., Mandryk, R.L., Inkpen, K.M.: TractorBeam: Seamless integration of local and remote pointing for tabletop displays. In: Proc. of GI '05, pp. 33–40 (2005)
16. Song, P., Goh, W.B., Hutama, W., Fu, C.W., Liu, X.: A handle bar metaphor for virtual object manipulation with mid-air interaction. In: Proc. of CHI '12, pp. 1297–1306 (2012)
17. Tokoro, Y., Terada, T., Tsukamoto, M.: A pointing method using two accelerometers for wearable computing. In: Proc. of SAC '09, pp. 136–141 (2009)
18. Voelker, S., Weiss, M., Wacharamanotham, C., Borchers, J.: Dynamic Portals: A lightweight metaphor for fast object transfer on interactive surfaces. In: Proc. of ITS '11, pp. 158–161 (2011)
19. Vogel, D., Balakrishnan, R.: Distant freehand pointing and clicking on very large, high resolution displays. In: Proc. of UIST '05, pp. 33–42 (2005)
20. Wang, R., Paris, S., Popović, J.: 6D hands: Markerless hand-tracking for computer aided design. In: Proc. of UIST '11, pp. 549–558 (2011)
21. Yoshikawa, T., Mita, Y., Kuribara, T., Shizuki, B., Tanaka, J.: A remote pointing technique using pull-out. In: Proc. of HCI '13, pp. 416–426 (2013)
22. Yoshikawa, T., Shizuki, B., Tanaka, J.: HandyWidgets: Local widgets pulled-out from hands. In: Proc. of ITS '12, pp. 197–200 (2012)
23. Zhang, H., Yang, X.D., Ens, B., Liang, H.N., Boulanger, P., Irani, P.: See Me, See You: A lightweight method for discriminating user touches on tabletop displays. In: Proc. of CHI '12, pp. 2327–2336 (2012)