SegTouch: Enhancing Touch Input While Providing Touch Gestures on Screens Using Thumb-To-Index-Finger Gestures

Hsin-Ruey Tsai
Te-Yen Wu
National Taiwan University
hsnuhrt@gmail.com
teyanwu@gmail.com

Min-Chieh Hsiu
Jui-Chun Hsiao
National Taiwan University
r03922073@ntu.edu.tw
r04922115@ntu.edu.tw

Da-Yuan Huang
Dartmouth College
Academia Sinica
dayuansmile@gmail.com

Yi-Ping Hung
Mike Y. Chen
Bing-Yu Chen
National Taiwan University
hung@csie.ntu.edu.tw
mikechen@csie.ntu.edu.tw
robin@ntu.edu.tw

CHI'17 Extended Abstracts, May 06-11, 2017, Denver, CO, USA
ACM 978-1-4503-4656-6/17/05.
http://dx.doi.org/10.1145/3027063.3053109

Abstract
Insufficient input modality on touchscreens causes icons, toolbars and mode switching steps required to perform different functions. Although various methods are proposed to increase touchscreen input modality, touch gestures (*e.g.*, swipe), usually used in touch input, are not provided in previous methods (*e.g.*, Force Touch on iPhone 6s). This still restricts the input modality on touchscreens. Hence, we propose SegTouch to enhance touch input while providing touch gestures. SegTouch uses thumb-to-index-finger gestures, *i.e.*, the thumb slides on the index finger, to define various touch purposes. Based on a pilot study, the middle and base segments on the index finger are suitable input areas for SegTouch. To observe how users leverage the proprioception and natural haptic feedback from index finger landmarks to perform SegTouch, different layouts on the index finger segments were examined in the eyes-free. Including the normal touch without thumb-to-index-finger gesture, SegTouch provides 9 input modality and touch gestures on the screen, so novel applications are enabled.

Author Keywords
Touch input; input modality; thumb-to-finger; touchscreens.

ACM Classification Keywords
H.5.2. [Information Interfaces and Presentation (e.g. HCI)]: Input devices and strategies (e.g., mouse, touchscreen)

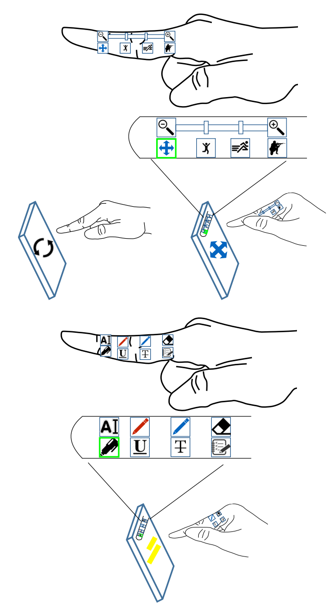

Figure 1: In SegTouch, buttons are assigned to different positions on the index finger to provide mode switching. Top: in 3D navigation, users swipe in conventional touch to rotate and swipe with SegTouch to translate. Down: tool buttons in reader and text editor.

Introduction

Comparing with mouse and keyboard, input modality on touchscreens is insufficient. There are basically only two modes, tap and long press for target selection. Besides, users use some touch gestures such as swipe and drag to perform simple functions. The restriction is even severer in small-screen devices, *e.g.*, smartphones. Although toolbars and icons are used to alleviate the problem, they both require additional mode switching and the content may be partially occluded due to the small screen. Furthermore, the small screen also limits the multi-touch gestures used on tablets. Although some methods are proposed for mode switching, users hardly perform touch gestures using them. Thus, enhancing touch input while providing touch gestures is essential to increase input modality of touchscreens.

Previous studies have proposed methods to enhance touch input. Using different touch poses [5, 10], touch forces [7] or in-air trajectories [3], users are allowed to switch different modes when performing touch input. TapSense [5] triggers functions using different finger parts, including the tip, pad, nail and knuckle, to tap that is recognized by sound classification. Using touch poses by touching with different finger pads, TouchSense [10], implemented by two motion sensors, provides 5 input modes, including the normal touch. ForceTap [7] uses the accelerometer in z-axis to recognize 2 touch forces. Combining in-air gestures and touch, Air+Touch [3] provides various touch input modality in 3 gesture categories, including before, between and after touches. Using multi-touch gestures, TouchTools [6] provides conventional touch gestures to hold and use tools. Using a stylus with different grips [13], gestures [16], stylus poses [1, 14] and pressures [12] is an alternative to enhance touch input. In off-the-shelf products, iPhone 6s provides 3D touch using a force sensing screen. Users can tap with different forces to trigger "peek" or "pop" functions. However, in the previous methods, altering the conventional touch pose generally restrains the touch gestures on the screen and suffers from touch error offsets [8]. Altering the in-air trajectories increases touch time. To enable more novel applications, more input modality while providing touch gestures on the screen are demanded.

SegTouch using the thumb sliding on the index finger defines various touch purposes to enhance touch input, which is similar to press buttons on a joystick or mouse. Using the thumb to perform mode switching, SegTouch allows users to maintain the conventional touch pose and touch gestures. Using the period before the index finger lifts from the screen, users leverage proprioception, haptic feedback, and visual feedback from the screen to quickly slide to thumb to the target position. To realize SegTouch, we first observed the suitable index finger segments as the thumb input area in a pilot study. Less visual attention on SegTouch avoids increasing much touch input time. Therefore, to understand how users perform SegTouch with less visual attention and explore users' limits in SegTouch, a human-factor study is performed in the eyes-free manner using the proprioception and natural haptic feedback of the index finger. Finally, applications, combining SegTouch and touch gestures on the screen, are proposed based on SegTouch (Figure 1).

The contributions of SegTouch are: (1) Defining various touch purposes using the spare thumb increases input modality. (2) Maintaining conventional touch pose provides touch gestures and avoids touch error offsets. (3) Providing haptic and visual feedback in advance reducing touch time.

SegTouch Interaction Design

When performing touch input on screens, users usually stretch the index finger to touch the screen. To enhance touch input, we propose SegTouch to use the dexterous

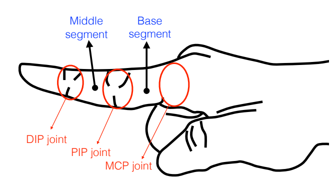

Figure 2: Anatomy of an index finger.

thumb to slide on the index finger. SegTouch is similar to the conventional touch pose and described in the following.

(1): The thumb touches and slides at different positions on the index finger segments and different touch purposes can be defined. The visual feedback is provided on the screen.

(2): The index finger touches the screen to perform target selection or touch gestures. Users can still perform *step 1* to adjust the position on the segments in *step 2*. The touch purpose is then defined based on the last position the thumb holding on the index finger segments.

(3): In target selection, the index finger lifts from the screen to complete the target selection. In touch gestures, the index finger moves to perform the gestures and lifts. The thumb then lifts the index finger segments.

Sliding the thumb in SegTouch provides the natural landmarks [9, 15] and haptic feedback on the index finger. This prevents users to pay much visual attention on SegTouch to increase much touch time. When using SegTouch, users do not need to lift the thumb between two consecutive touch tasks to maintain the natural haptic feedback and speedup to perform the SegTough gestures. To understand which input areas are adequate for SegTouch, how users perform it with less visual attention, and what SegTouch layouts are practical in touch input, we performed the following studies.

Pilot Study - Observing Input Area of SegTouch

There are three finger segments in a index finger. We decided the input area for SegTouch by observing users touching on the screen with the index finger in the pilot study. Although the similar study to determine input area on segments was performed in [9, 15], stretching the index finger when touching on the screen caused the condition quite different from them. 7 participants (4 female) were recruited.

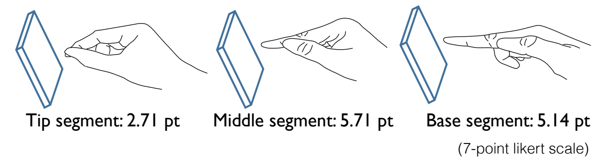

Figure 3: Using smartphones when the thumb touches tip, middle and base segments of the index finger (from left to right).

One is left-handed. They were asked to touch the three index finger segments, including tip, middle and base segments, with the thumb and use the index finger to perform common touch tasks on a smartphone for 3 to 5 minutes, separately. We interviewed them after the experiment. Based on their feeling of the touching poses, they gave a score to each segment using a 7-point Likert scale. 7 points meant the most preferred pose.

The results revealed that the tip segment (mean: 2.71; SD: 1.60) is less preferred. The middle (mean: 5.71; SD: 1.11) and base (mean: 5.14; SD: 1.35) segments obtained higher scores. Based on the interview, two factors, including occlusion and stability are commonly considered. When touching the segments near the tip, the thumb usually occludes the target. Besides, the distal interphalangeal (DIP) joint and proximal interphalangeal (PIP) joint [11] (Figure 2) sometimes move when touching, so the segments near the tip are unstable for touch input. In addition, touching the tip segment made the thumb prone to touch screens accidentally. Although touching on the base segment made the thumb squeezed, the middle and base segments obtained similar scores, so they are used as SegTouch input area.

Human-Factor Study

Although users may look at the fingers and screen while performing SegTouch (and normal touch), paying much

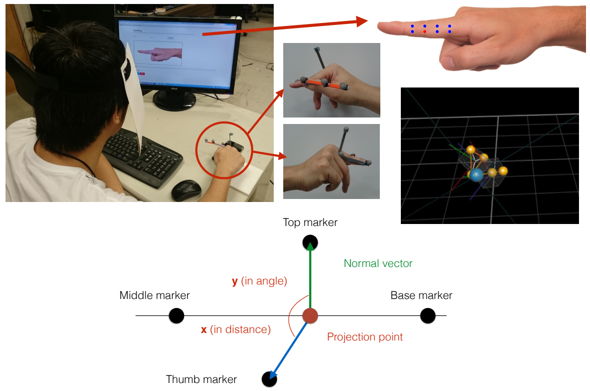

Figure 4: Experiment apparatus (left), including markers (middle) and Vicon tracking system (right). Upper right: instruction in the experiment shown on the monitor. The red point means the target. Down: the thumb position computed in SegTouch.

visual attention on those might slow down the touch input. In this study, we want to observe how users only use the proprioception and natural haptic feedback of index finger in SegTouch to distinguish different positions in different layouts in the eyes-free manner using the middle and base segments as the input area.

Apparatus and Participants

To obtain precise positions of the thumb and index finger, we attached markers on the fingers and used the Vicon system for tracking. Two 3D printed supports with three markers on each were attached to the thumb nail and side of the index finger (Figure 4 (top)). A smartphone was fixed on the desk to provide only screen haptic but no visual feedback. A board fixed on the desk next to the smartphone as the home position. The participants wore a card board on the head to prevent the visual feedback. 8 right-handed participants (4 male) aged 22-30 (mean: 26) were recruited.

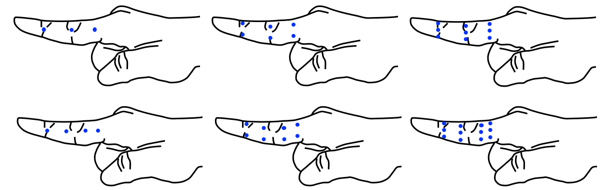

Figure 5: Index finger landmarks and 6 layouts in the human-factor study.

They received some incentive after the experiment.

The Vicon system provided the markers' positions and we further inferred the thumb position in SegTouch. The two markers on the index finger provided the PIP and metacarpophalangeal (MCP) joints' positions [11], and formed a line matching to the pose when stretching the index finger to touch the screen. The thumb marker's position was projected onto the line to provide the horizontal position in SegTouch. We found that the participants distinguished the movement in vertical mainly based on the curve of the index finger in pilot. Thus, we used the angle between the normal vector of back of the index finger and the vector from the projection point on the line to the thumb's marker position to infer the vertical touch position (Figure 4 (down)).

Task and Procedure

We observed that at least 3 positions in horizontal layout could be distinguished in SegTouch easily from a pilot. Thus, we gradually increased the point numbers in horizontal and further tested in the 2D layouts. A total of 6 different layouts, including (3), (4), (3+3), (4+4), (3+3+3) and (4+4+4) were tested orderly, as illustrated in Figure 5. When each layout was shown firstly, the participants had one minute to determine the points' positions on the segments. They were notified that the points' positions in the layout figure

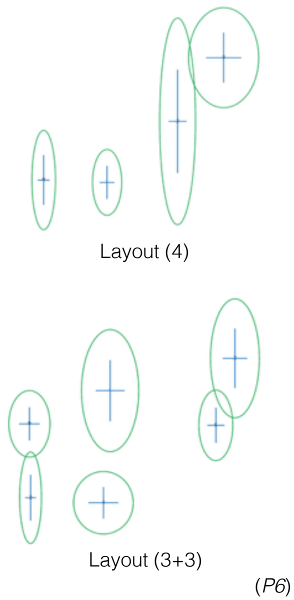

Figure 7: Results of layout (4) and (3+3) from (*p6*). Top: two ellipse-like regions were slightly overlapping in (4). Down: ellipse-like regions were slightly overlapping between two rows in left and right positions in (3+3)

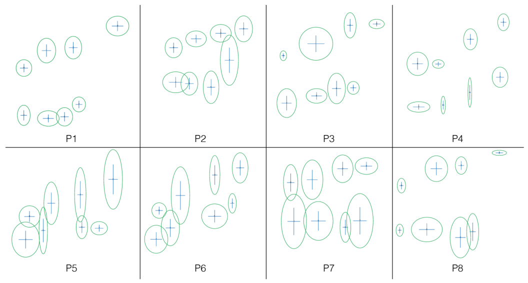

Figure 6: Results of layout (4+4) from all participants in the human-factor study.

were only for illustration. They could determine the position without changing the layout. Before each trial, the hand laid on the home position and the thumb did not touch the segments. After a red target point shown in the layout figure (Figure 4 (upper right)), they slid the thumb to the target position and then touched the smartphone screen using the index finger. The experimenter checked whether the markers were occluded and recorded the markers' positions. There was no feedback provided to the participants. They laid the hand to the home position for the next trial. Each position in each layout was randomly repeated 6 times. A total of 252 (=18+24+36+48+54+72) trials were examined for each participant. We interviewed them after the experiment. The experiment took about 45 minutes.

Results and Discussion

For each target in each layout, the touched positions from all trials were recorded and a 95% ellipse-like confidence region was drawn (Figure 6). All participants clearly distinguished all the targets in layouts (3), (4) and (3+3) except (*P6*). Although two ellipse-like regions were slightly overlapping in (4) and ellipse-like regions were slightly overlapping between two rows in left and right positions in (3+3) from (*P6*) (Figure 7), regions were non-overlapping in upper row and three pairs in the two rows in (4+4) from (*P6*). Hence, we still supposed that (4) and (3+3+3) were distinguishable. In (4+4), more than two ellipse-like regions were overlapping from most of the participants. More and larger overlapping areas appeared in layouts (3+3+3) and (4+4+4).

Most of the participants said that based on the proprioception, they can perceive the approximate positions of the DIP, PIP and MCP joints in eyes-free, and use the joints as reference positions to define the points in each layout. After touching the joint close to the target, they then slid the thumb to the real target. In (3), the PIP joint was commonly treated as the middle point. 5 participants assigned the DIP and MCP joints for the other points, separately. The others assigned the concave parts on middle and base segments for the points, respectively. They did not want to stretch and squeeze the thumb too hard to touch the DIP and MCP joints in (3). In (4), the positions in left and right near the PIP joint were for the two middle points. The DIP and MCP joints were for the other points, respectively. In multi-row layouts, they used the side of the index finger bone as the landmark to recognize different rows. The layouts with two rows were easy to distinguish but those with three rows were generally supposed hard to distinguish in eyes-free.

Although two rows were still distinguishable, half of the participants mentioned that more time was needed in (4+4). Some of the participants sometimes slightly bending the index finger. It caused that the thumb was squeezed when sliding in the lower row, especially near the palm, and the adjacent points were undistinguishable (Figure 6). We also observed that in vertical direction, the targets near the DIP joint were generally lower than those near the MCP joint due to the thumb base close to the MCP joint in anatomy. Furthermore, the input area in vertical direction quite depended on the participants. Based on the results (Figure 6)

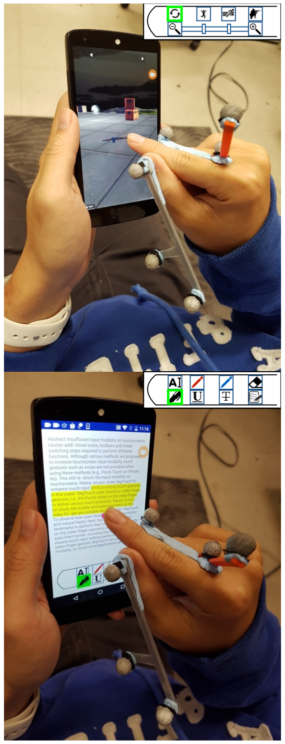

Figure 8: Demo applications. Top: 3D navigation in a first person game. Different movements can be triggered using SegTouch. Rotation is selected so users can swipe on the screen to rotate the view. Down: the reader app provides different tools using SegTouch. Highlight is selected so users can drag the text to highlight.

and comments, we supposed that if one point in the lower row in (4+4) was removed, most of the participants could clearly distinguish the targets (*i.e.*, (4+3)). Certainly, instead of in the eyes-free condition, users could improve performance with less visual attention on SegTouch, which means that (4+4) is feasible based on a pilot. We will further evalute SegTouch performance in the future work.

Applications

Two applications, 3D navigation and reader and text editor, are proposed, as shown Figure 8. SegTouch and touch gestures on the screen (*e.g.*, swipe and drag) are used at the same time in these applications. We also demonstrate the SegTouch applications using Vicon system in the video.

3D navigation: Input in 3D navigation on smartphones is still unsatisfied due to iterative mode switching or additional icons for rotation and translation controls (*e.g.*, Google Street View). SegTouch allows users to perform gesture swipe on the screen with the conventional touch pose to control the translation and use SegTouch to control the rotation. Users can slide on the other positions and tap the screen to perform different movements such as jump, sprint and crouch in first-person shooter games. Using multi-row layouts in SegTouch, the other row is used for zooming. Users slide to the desired scale and touch the zooming target on the screen. Without lifting the index finger, users adjust the zooming scale by sliding on a row in SegTouch. This prevents occlusion using the pinch gesture.

Reader and text editor: Long press and drag gestures are commonly used in reader and text editor apps. However, long press requires about 1 sec. duration to trigger. This is undesired by users. Instead of the long press, users can use SegTouch and drag on the screen to select text for cut or copy, highlight text, underline text and strikethrough text. Combing SegTouch and draw, the users can pens and eraser to write, draw or erase on the screen. By sliding to other positions and tap the screen, the users can add some components such memos and comments. Without lifting the thumb in SegTouch, they can consecutively switch tools.

Future Work

We propose and perform preliminary design and studies of SegTouch in this paper. To further understand and evaluate SegTouch, we will perform a user study that users use SegTouch to perform in target selection and touch gestures, as shown in the demo video. In terms of SegTouch implementation, gesture tracking methods were proposed in previous studies using a fish-eye camera [2], omnidirectional camera [17] or depth camera [3]. We will implement SegTouch by equipping an infrared camera on the smartphone and propose a vision-based recognition method in the future work. Combining SegTouch and touch gestures on the screen, more novel applications will be proposed and implemented such as multitasking [4] in the future.

Conclusion

We propose SegTouch to enhance touch input on touchscreens. Based on our user studies, 6 to 8 points in the layout (3+3) or (4+4) could be distinguished. Combining the normal touch, 9 input modality can be provided. SegTouch provides visual and haptic feedback and maintains conventional touch pose to provide touch gestures and prevent touch error offsets. It provides novel interactions and applications for users and simplifies mode switching.

Acknowledgements

This work was partly supported by Ministry of Science and Technology, MediaTek Inc., and Intel Corporation under Grants MOST 104-2221-E-002-050-MY3, MOST 105-2622-8-002-002 and MOST106-2633-E-002-001.

References

[1] Xiaojun Bi, Tomer Moscovich, Gonzalo Ramos, Ravin Balakrishnan, and Ken Hinckley. 2008. An exploration of pen rolling for pen-based interaction. In *Proceedings of the 21st annual ACM symposium on User interface software and technology*. ACM, 191–200.

[2] Liwei Chan, Yi-Ling Chen, Chi-Hao Hsieh, Rong-Hao Liang, and Bing-Yu Chen. 2015. CyclopsRing: Enabling Whole-Hand and Context-Aware Interactions Through a Fisheye Ring. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. ACM, 549–556.

[3] Xiang'Anthony' Chen, Julia Schwarz, Chris Harrison, Jennifer Mankoff, and Scott E Hudson. 2014. Air+ touch: interweaving touch & in-air gestures. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. ACM, 519–525.

[4] Aakar Gupta, Muhammed Anwar, and Ravin Balakrishnan. 2016. Porous Interfaces for Small Screen Multitasking using Finger Identification. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. ACM, 145–156.

[5] Chris Harrison, Julia Schwarz, and Scott E Hudson. 2011. TapSense: enhancing finger interaction on touch surfaces. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*. ACM, 627–636.

[6] Chris Harrison, Robert Xiao, Julia Schwarz, and Scott E Hudson. 2014. TouchTools: leveraging familiarity and skill with physical tools to augment touch interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2913–2916.

[7] Seongkook Heo and Geehyuk Lee. 2011. Forcetap: extending the input vocabulary of mobile touch screens by adding tap gestures. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*. ACM, 113–122.

[8] Christian Holz and Patrick Baudisch. 2010. The generalized perceived input point model and how to double touch accuracy by extracting fingerprints. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 581–590.

[9] Da-Yuan Huang, Liwei Chan, Shuo Yang, Fan Wang, Rong-Hao Liang, De-Nian Yang, Yi-Ping Hung, and Bing-Yu Chen. 2016. DigitSpace: Designing Thumb-to-Fingers Touch Interfaces for One-Handed and Eyes-Free Interactions. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 1526–1537.

[10] Da-Yuan Huang, Ming-Chang Tsai, Ying-Chao Tung, Min-Lun Tsai, Yen-Ting Yeh, Liwei Chan, Yi-Ping Hung, and Mike Y Chen. 2014. TouchSense: expanding touchscreen input vocabulary using different areas of users' finger pads. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 189–192.

[11] David Kim, Otmar Hilliges, Shahram Izadi, Alex D Butler, Jiawen Chen, Iason Oikonomidis, and Patrick Olivier. 2012. Digits: freehand 3D interactions anywhere using a wrist-worn gloveless sensor. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*. ACM, 167–176.

[12] Gonzalo Ramos, Matthew Boulos, and Ravin Balakrishnan. 2004. Pressure widgets. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 487–494.

[13] Hyunyoung Song, Hrvoje Benko, Francois Guimbretiere, Shahram Izadi, Xiang Cao, and Ken Hinckley. 2011. Grips and gestures on a multi-touch pen. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 1323–1332.

[14] Feng Tian, Lishuang Xu, Hongan Wang, Xiaolong Zhang, Yuanyuan Liu, Vidya Setlur, and Guozhong Dai. 2008. Tilt menu: using the 3D orientation information of pen devices to extend the selection capability of pen-based user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 1371–1380.

[15] Hsin-Ruey Tsai, Cheng-Yuan Wu, Lee-Ting Huang, and Yi-Ping Hung. 2016. ThumbRing: private interactions using one-handed thumb motion input on finger segments. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. ACM, 791–798.

[16] Haijun Xia, Tovi Grossman, and George Fitzmaurice. 2015. NanoStylus: Enhancing Input on Ultra-Small Displays with a Finger-Mounted Stylus. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. ACM, 447–456.

[17] Xing-Dong Yang, Khalad Hasan, Neil Bruce, and Pourang Irani. 2013. Surround-see: enabling peripheral vision on smartphones during active use. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*. ACM, 291–300.