WRISTBAND.IO: Expanding Input and Output Spaces of a Smartwatch

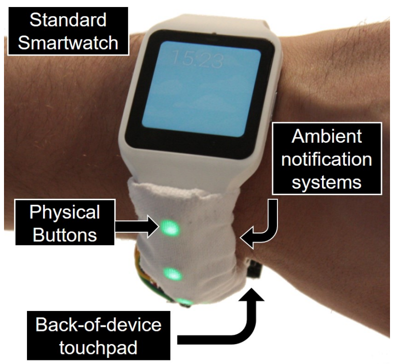

Figure 1: *Wristband.io* expands the interaction space of a smartwatch screen to the entire watchband.

Léa Saviot
École polytechnique
91120 Palaiseau, France
lea.saviot@polytechnique.edu

Steven Houben
Lancaster University
Lancaster, LA1 4WA, UK
s.houben@lancaster.ac.uk

Frederik Brudy
University College London
London, WC1E 6EA, UK
f.brudy@cs.ucl.ac.uk

Abstract
Smartwatches are characterized by their small size designed for wearability, discretion, and mobile interactions. Most of the interactivity, however, is limited to the size of the display, introducing issues such as screen occlusion and limited information density. We introduce *Wristband.io*, a smartwatch with an extended interaction space along the wristband, enabling (i) back-of-band interaction using a touchpad, (ii) a low resolution ambient watchband display for off-screen notification, and (iii) tangible buttons for quick, eyes-free input. Insights gained through a study show that back-of-band input increases accuracy and task completion rates for smaller on-screen targets. We probe the design space of *Wristband.io* with three applications.

CHI'17 Extended Abstracts, May 06-11, 2017, Denver, CO, USA
ACM 978-1-4503-4656-6/17/05.
http://dx.doi.org/10.1145/3027063.3053132

Author Keywords
Smartwatch; Watchband; Interaction Technique; Back-of-device interaction; Wristband.io

ACM Classification Keywords
H.5.2. Information Interfaces. User Interfaces – input devices and strategies

Introduction
People increasingly use smartwatches as yet another device to interact with applications, information, and services. However, the small size of smartwatches and the properties of the touchscreen introduce several fundamental issues. Because of the limited size of the touchscreen on the watch, problems such as *screen occlusion* [24] and *fat-finger problem* make it difficult or even impossible to select small targets [4,23]. Furthermore, the absence of a physical keyboard or

other physical input methods forces the user to direct their attention to the watch for each interaction or input, creating distracting or potentially dangerous situations when using the watch on the go. More generally, the input and output on a smartwatch is confined to a small area on the top of the armband, only using about 20% of the physical real estate. Extending the input and output space of the watch across the entire available watchband can mitigate these problems and open up new design spaces for interaction with smartwatches.

In this paper, we introduce *Wristband.io* (Figure 1), a smartwatch with an interactive band that provides (i) back-of-band input allowing for precise target selection while leveraging the entire screen for visual feedback, (ii) an ambient notification system providing off-screen feedback (Figure 2), and (iii) a set of tangible programmable buttons on the wristband, allowing for eyes-free input (Figure 3). The contribution of this work is (i) a smartwatch prototype that extends interaction across the entire watchband and (ii) a preliminary evaluation of user interaction with the back-of-band touchpad, demonstrating that back-of-band interaction provides a reliable solution for precise target acquisition, at the cost of slower task-completion time and increased user frustration.

Related Work

Several projects have expanded smartwatch input beyond touch and voice. For example, gesture input is achieved using sound [1,11], infrared sensors [12–14], motion tracking [22], or finger and hand recognition [28]. Eye-gaze input has been proposed [6,10]. However, these solutions are dependent on the environment. Input via arm gestures might be socially awkward, and the user may need to switch recognition on and off to prevent an accidental trigger. Speaking to one's watch might not be suited for quiet spaces or not possible in noisy and crowded places. Eye tracking requires visual attention, which might be impossible, for example when driving a car. Other projects have explored using a screen that can be twisted, panned or clicked [27,29], around-the-bezel interaction [3], or haptic feedback [18]. These approaches focus on expanding the interaction space of the existing display area while only little attention is given on using the entire watchband. This watchband space was explored in Watchit [19] and in Funk et al.'s work [9] for touch input, and in Facet [15] for multi-screen output. These approaches demonstrate the feasibility of using the wristband for input, but have not explored tangible input or back-of-device interactions. Several techniques to overcome screen occlusion and fat-finger problem exist, e.g. a cursor offset [21], using the *Shift* technique [25], or back-of-device interaction [2].

A display is "ambient" if it is aesthetic and on the periphery of the user's visual attention [16]. A watch's wristband has been used as an ambient display for group chats [26] and fluid intake reminder [7]. Design implications for presenting information on wrist-worn LED displays have been presented [8]. We extend this work by integrating a low-resolution ambient display providing feedback and a notification mechanism.

Our work resembles Facet [15] and Watchit [19] in its scope: enlarging the interaction space with multi-purpose techniques located on the wristband. *Wristband.io*'s combination of ambient display, tangible buttons and back-of-device interaction creates a novel interaction space.

Figure 2: Back-of-band interaction.

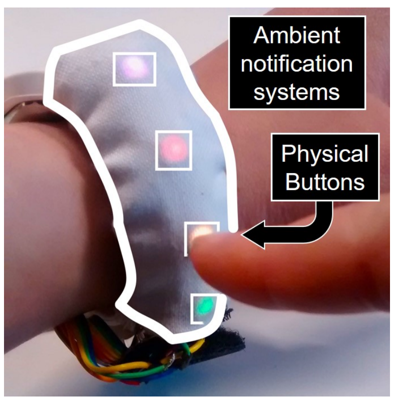

Figure 3: Low resolution ambient color display and physical programmable buttons.

Wristband.io

Most existing smartwatches provide both input and output through its display while ignoring the real estate of the watchband. However, these wristbands are a large empty area with high visibility to the user in many everyday positions of the arm, e.g., when typing on a computer, reading a sheet of paper, carrying a drink, or riding a bike. *Wristband.io* (Figure 1) leverages this unused space by enabling new interactions: (i) *back-of-band touch input*, (ii) a *low resolution ambient notification system,* and (iii) a set of *programmable tangible buttons*.

Back-of-band Interactions

Although direct touch input is an easy-to-use interaction with interfaces on the watch, the finger touch occludes the display, limiting visual feedback, information density, and interface component sizes of smartwatch applications. We propose to include secondary touch input via a back-of-wristband touchpad (Figure 2), enabling users to select targets with higher precision while not occluding the display. The touchpad is designed as an implementation of the back-of-device technique [2], placed diametrically opposite to the screen on the wristband. When touched, a cross appears on the smartwatch's screen, indicating the current cursor position. It allows to relatively move the cursor across an X-Y plane. A single tap triggers a touch input on the smartwatch. This secondary input space complements the existing touch input on the screen, allowing users to choose the input space (touchscreen or wristband pad), depending on target size, application, or purpose. Additionally, both input spaces can be combined, allowing simultaneous use of touchscreen and back-of-band touchpad.

Ambient Notification Display

We include RGB LEDs as a low-resolution ambient display on the inside of the wristband (Figure 3). This off-screen display provides an output space when the screen is turned off and allows users to glance at information, even when the display is not within eyesight (e.g., while riding a bike). Through their location in the user's visual periphery, the LEDs position provide a quick-access output channel, creating an aesthetic low-resolution notification system that does not increase mental load and visual clutter. The notification display can be configured by the user via a smartphone application, allowing to customize triggers that change color and brightness of the LEDs.

Tangible Buttons for Eyes-Free Interaction

Wristband.io includes programmable tangible buttons to allow for eyes-free input (Figure 3). Each button is placed directly underneath one LED of the ambient display, granting visual and haptic guidance when pressing the button, as well as a one-to-one mapping between output (LEDs) and input (button). Each button can be reconfigured to define actions or provide shortcuts to frequently used functions.

Implementation

Wristband.io extends the interactions of a Sony Smartwatch 3 SWR50. Using a curved perfboard, mini push-buttons (mounted under RGB LEDs) are attached to the wristband, covered with fabric. An Ergo Mini Touchpad is placed on the wristband. LEDs, buttons, and touchpad are controlled via a battery-powered RFduino, connected to an Android smartphone, acting as proxy for the communication between watch and *Wristband.io*. Based on a pilot study, cursor speed was set to half the speed of the finger movement.

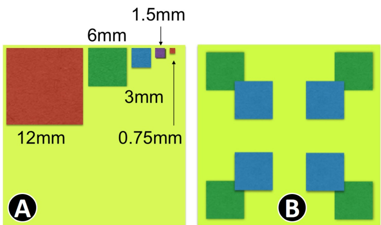

Figure 4: (A) Relative comparison of the target sizes. (B) Positions of the 8 targets used in the user study.

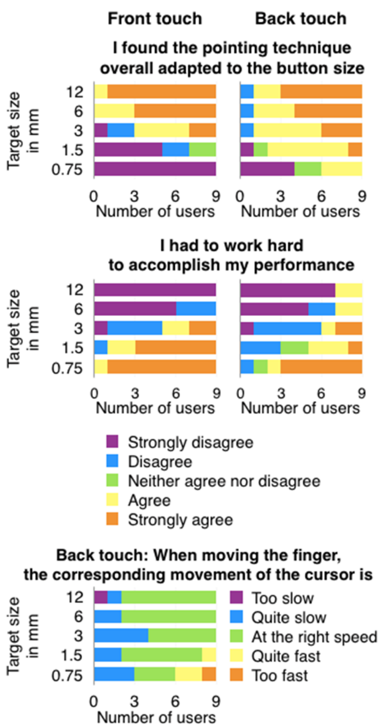

Figure 5: Participants' responses to the questionnaire.

User Study

We conducted a user study to evaluate the efficiency of the new dual input method. We used a 2x5 within-subject experimental design, with two factors: (i) *interaction technique* (with levels "**f**ront-touch**s**creen" (**FS**), consisting in absolute pointing on the screen; "**b**ack-of-band **t**ouchpad" (**BT**), which controls a cursor) and (ii) *target size* (sizes of a target square are 12mm, 6mm, 3mm, 1.5mm and 0.75mm (Figure 4A)). A combination of FS and BT was tested at the end of the study, where users could alternate between techniques. We used 8 positions per target, as shown in Figure 4B.

We recruited 10 volunteers (4 female; 21-58 years old, mean 30). All, except one, were regular touchscreen users and three had prior smartwatch experience. Participants were asked to perform simple selection tasks, using the three interaction techniques. Each experiment was composed of 120 trials (3 techniques x 5 sizes x 8 positions) and lasted ~35 minutes. Latin square counterbalancing of target sizes was used to account for transfer effects. Each user performed the same order of target sizes for every technique. Half of the participants started with FS, the other half with BT, always finishing with the "both technique".

A training phase for each interaction technique was given, and participants were reminded when to use which technique. Upon successful selection, the application logged the selection time and the number of failed attempts ("false clicks") and a new target appeared. If the interaction technique included the touchpad (*BT* and *both*), the cursor was positioned at the center of the screen before each trial. An upper limit of 10 false clicks was set, after which a failed trial was recorded and the trials continued. Participants answered a questionnaire based on the System Usability Scale (SUS) [5] after each technique.

Results

Most participants felt comfortable using *Wristband.io*, only P10 felt that "*the touchpad would be better on the front or on the side of the touchscreen, [as she did not] like not being able to see it*". Participants were overall satisfied with the cursor speed (Figure 5), however for targets ≤0.75mm, some participants felt that the cursor was too slow and they had to "*swipe several times to get to the target*" (P1). On the contrary, P2 felt that the cursor was too fast, especially when using the 'both technique', as "*starting closer to the target, speed was not needed*".

Out of 1200 trials (10 participants x 3 techniques x 5 sizes x 8 positions), 8 were discarded because of technical issues. Participants found selecting bigger targets easy using the touchscreen. As anticipated, the average success rate (Figure 6) for FS drops for targets <3mm and the number of false clicks increases. Note: in this case, average false selections are likely underestimated, as the number of attempts was limited to 10.

Using BT, all participants managed to select all targets and smaller targets could be easier acquired, except for P8 who was not a regular touchscreen user. However, despite the higher success rate for small targets, some users felt that using the touchpad was time consuming and required more concentration.

The completion time for FT uses only the data of successful trials. However, for the 0.75mm condition this comprises of only 19% of these trials. Thus,

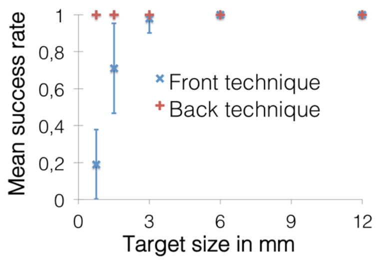

Figure 6: Average success rate (+\-SD) depending on target size.

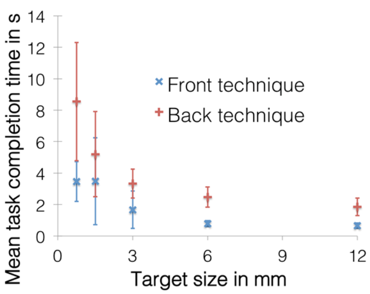

Figure 7: Average task completion time (+\-SD).

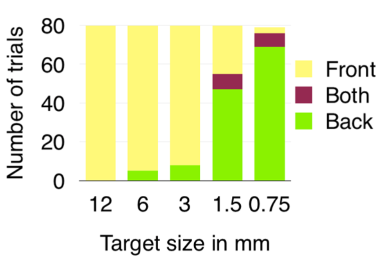

Figure 8: Techniques used to select a target when both techniques were enabled.

average completing times (Figure 7) should be handled with care: if a user failed to select the target for 7 positions but luckily manages to click on the 8th in one second, the average time will be one second. In this situation, we observed that most users would try to aim correctly for the first touch of the first positions and then frenetically click on the screen to quickly get through the 80 clicks. *If* a target was correctly acquired, the completion time was under ~6s, which is the maximum time participants needed to perform 10 fast clicks.

When allowed to use both techniques, users mostly used the touchscreen for targets ≥3mm, and mostly switched to the touchpad for size <3mm. They rarely combined the front and back inputs to acquire a single target (<4%; 3 users; size <3mm; Figure 8). When they did so, they would either move the cursor closer to the target with the touchscreen then use the touchpad, or make several attempts at clicking on the target with the touchscreen, then, upon failure, switch to the touchpad. P1 indicated that he "*liked the mixed technique. [He] did not really use it because the test was too short*", while P3 expressed concerns about the long time needed to move the hand from front to back.

Interaction techniques and applications

To demonstrate functionality and use of *Wristband.io* we describe three example applications.

Application 1: Wristband Express
Smartwatch users get lost in the overview of applications, spending 39.8 seconds searching for an app (compared to 6.7 seconds average smartwatch use) [20]. *Wristband Express* (Figure 9), allows to use buttons as shortcuts to applications and settings, enabling eyes-free input. They are configurable via a smartwatch application; colored LEDs make them easier to remember.

Application 2: Wristband Unlock
A smartwatch is a wearable device that is usually not taken off or shared, making it a good candidate to allow owners to unlock their smartphone. Although e.g. Android phones already allow people to select Bluetooth devices that will unlock the phone when in close proximity, this poses a risk: wearing a smartwatch and leaving the phone near but out of sight, will unlock it and anybody nearby can use it. *Wristband Unlock* mitigates this problem by requiring a deliberate user action to unlock the smartphone (Figure 10). The buttons act as a password keypad; buttons can be pressed simultaneously allowing more complex passwords. Colored LEDs make it easier to remember the code.

Figure 9: *Wristband Express* creates app shortcuts (A), associated with *Wristband.io*'s buttons location and color (B).

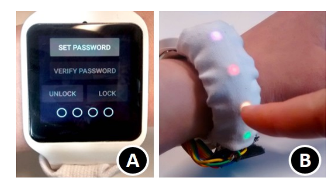

Figure 10: *Wristband Unlock* allows to set passwords (A) and enter them using buttons on the band's side (B).

Application 3: Wristband Notify
Checking phone notifications is the most frequently used and time consuming task [20]. Notifications can be disruptive, increase task completion time and task error rates [17]. Using *Wristband Notify* (Figure 11), users can customize and filter which notifications from

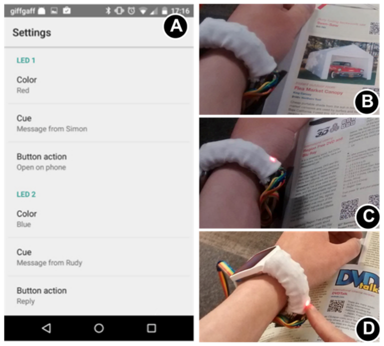

Figure 11: *Wristband Notify* allows mapping lights to notifications and actions (A). A new notification is visualized through light (B, C). Pressing the button (D) here opens the notification on the phone.

their phone they want to display on the wristband using a visual programming interface on their phone. Buttons can be programmed to trigger quick response actions.

Discussion

Wristband.io expands the input and output space of an off-the-shelf smartwatch by introducing (i) back-of-band touchpad, (ii) a low resolution ambient notification system, and (iii) a set of programmable tangible buttons.

Our study demonstrated that back-of-band interaction can be reliably used to interact with interface elements that are smaller than one millimeter, enabling user interfaces with smaller targets and higher number of targets on the screen. However, the accuracy using the touchpad comes at the cost of higher completion times and increased user frustration. A target size of 3mm was the threshold that caused users to switch from FS to BT, suggesting that for elements <3mm, the touchscreen becomes ineffective. The qualitative results show that the touchscreen has a sharp satisfaction threshold: users were highly satisfied when using it for targets >6mm, and highly dissatisfied for targets <1.5mm. For the touchpad, most people were somewhat satisfied for targets >3mm, although all preferred using the touchscreen.

Future designs of applications using back-of-band touchpads should carefully consider the tradeoffs and benefits of small interface components on smartwatches, as decreased user satisfaction, and increased task completion time need to be weighed against increased information density. Further, the time cost when changing between the front touchscreen and the back touchpad has to be considered by limiting the number of required switches.

The physical buttons and low resolution ambient notification system allows to show and interact with information in the periphery of users without the need to turn on the display of the watch. The programmable tangible buttons allow users to define actions, events or functionality to the specific input buttons. Since the buttons are integrated in the LED of the ambient display, the color of the light can be used semantically to visualize input possibilities or provide off-screen feedback on user input. Using three example applications, we demonstrated how *Wristband.io* can be used to build novel applications and support existing watch interface elements.

Future Work

Future work includes building an embedded and smaller version of *Wristband.io*. This will allow us to explore more complex application examples that combine the three interaction techniques we added, and build cross-device scenarios where the interactive band is used to control and share information with a secondary display.

We have evaluated the efficiency of the back-of-band input method. However, while we explore *Wristband.io*’s design space through example applications, we still have to formally study the interaction with the ambient notification display and the programmable physical buttons. Conducting in-the-wild studies will allow to gain insights about the use and applicability of *Wristband.io*’s design for everyday life.

References

1. Brian Amento, Will Hill, and Loren Terveen. 2002. The sound of one hand: a wrist-mounted bio-acoustic fingertip gesture interface. In *CHI'02 Extended Abstracts on Human Factors in Computing Systems*, 724–725. Retrieved June 24, 2016 from http://dl.acm.org/citation.cfm?id=506566
2. Patrick Baudisch and Gerry Chu. 2009. Back-of-device interaction allows creating very small touch devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1923–1932. Retrieved June 24, 2016 from http://dl.acm.org/citation.cfm?id=1518995
3. Gabor Blasko and Steven Feiner. 2004. An interaction system for watch computers using tactile guidance and bidirectional segmented strokes. In *Wearable Computers, 2004. ISWC 2004. Eighth International Symposium on*, 120–123. Retrieved June 24, 2016 from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1364698
4. Sebastian Boring, David Ledo, Xiang'Anthony' Chen, Nicolai Marquardt, Anthony Tang, and Saul Greenberg. 2012. The fat thumb: using the thumb's contact size for single-handed mobile interaction. In *Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services*, 39–48. Retrieved July 6, 2016 from http://dl.acm.org/citation.cfm?id=2371582
5. John Brooke. 1996. SUS-A quick and dirty usability scale. In *Usability Evaluation In Industry*. CRC Press, 4–7.
6. Augusto Esteves, Eduardo Velloso, Andreas Bulling, and Hans Gellersen. 2015. Orbits: Gaze Interaction for Smart Watches using Smooth Pursuit Eye Movements. In *UIST '15: Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, 457–466. https://doi.org/10.1145/2807442.2807499
7. Jutta Fortmann, Vanessa Cobus, Wilko Heuten, and Susanne Boll. 2014. WaterJewel: design and evaluation of a bracelet to promote a better drinking behaviour. 58–67. https://doi.org/10.1145/2677972.2677976
8. Jutta Fortmann, Heiko Müller, Wilko Heuten, and Susanne Boll. 2014. How to present information on wrist-worn point-light displays. 955–958. https://doi.org/10.1145/2639189.2670249
9. Markus Funk, Alireza Sahami, Niels Henze, and Albrecht Schmidt. 2014. Using a touch-sensitive wristband for text entry on smart watches. In *CHI EA '14: CHI '14 Extended Abstracts on Human Factors in Computing Systems*, 2305–2310. https://doi.org/10.1145/2559206.2581143
10. John Paulin Hansen, Florian Biermann, Emilie Møllenbach, Haakon Lund, Javier San Agustin, and Sebastian Sztuk. 2015. A GazeWatch Prototype. 615–621. https://doi.org/10.1145/2786567.2792899
11. Chris Harrison, Desney Tan, and Dan Morris. 2010. Skinput: appropriating the body as an input surface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 453–462. Retrieved June 24, 2016 from http://dl.acm.org/citation.cfm?id=1753394
12. Steven Houben and Nicolai Marquardt. 2015. WatchConnect: A Toolkit for Prototyping Smartwatch-Centric Cross-Device Applications. 1247–1256. https://doi.org/10.1145/2702123.2702215

13. Jungsoo Kim, Jiasheng He, Kent Lyons, and Thad Starner. 2007. The gesture watch: A wireless contact-free gesture based wrist interface. In *2007 11th IEEE International Symposium on Wearable Computers*, 15–22. Retrieved June 24, 2016 from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4373770
14. Jarrod Knibbe, Diego Martinez Plasencia, Christopher Bainbridge, Chee-Kin Chan, Jiawei Wu, Thomas Cable, Hassan Munir, and David Coyle. 2014. Extending interaction for smart watches: enabling bimanual around device control. In *CHI EA '14: CHI '14 Extended Abstracts on Human Factors in Computing Systems*, 1891–1896. https://doi.org/10.1145/2559206.2581315
15. Kent Lyons, David Nguyen, Daniel Ashbrook, and Sean White. 2012. Facet: a multi-segment wrist worn system. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, 123–130. Retrieved June 24, 2016 from http://dl.acm.org/citation.cfm?id=2380134
16. Jennifer Mankoff, Anind K. Dey, Gary Hsieh, Julie Kientz, Scott Lederer, and Morgan Ames. 2003. Heuristic evaluation of ambient displays. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, 169–176. Retrieved July 6, 2016 from http://dl.acm.org/citation.cfm?id=642642
17. Abhinav Mehrotra, Veljko Pejovic, Jo Vermeulen, Robert Hendley, and Mirco Musolesi. 2016. My Phone and Me: Understanding People's Receptivity to Mobile Notifications. In *CHI '16: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 1021–1032. https://doi.org/10.1145/2858036.2858566
18. Jerome Pasquero, Scott J. Stobbe, and Noel Stonehouse. 2011. A haptic wristwatch for eyes-free interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 3257–3266. Retrieved June 24, 2016 from http://dl.acm.org/citation.cfm?id=1979425
19. Simon T. Perrault, Eric Lecolinet, James Eagan, and Yves Guiard. 2013. Watchit: simple gestures and eyes-free interaction for wristwatches and bracelets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1451–1460. Retrieved June 24, 2016 from http://dl.acm.org/citation.cfm?id=2466192
20. Stefania Pizza, Barry Brown, Donald McMillan, and Airi Lampinen. 2016. Smartwatch in *vivo*. 5456–5469. https://doi.org/10.1145/2858036.2858522
21. Richard L. Potter, Linda J. Weldon, and Ben Shneiderman. 1988. Improving the accuracy of touch screens: an experimental evaluation of three strategies. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, 27–32. Retrieved June 27, 2016 from http://dl.acm.org/citation.cfm?id=57171
22. Jun Rekimoto. 2001. Gesturewrist and gesturepad: Unobtrusive wearable interaction devices. In *Wearable Computers, 2001. Proceedings. Fifth International Symposium on*, 21–27. Retrieved June 24, 2016 from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=962092
23. Katie A. Siek, Yvonne Rogers, and Kay H. Connelly. 2005. Fat finger worries: how older and younger users physically interact with PDAs. In *IFIP Conference on Human-Computer Interaction*, 267–280. Retrieved July 6, 2016 from http://link.springer.com/10.1007%2F11555261_24

24. Daniel Vogel and Patrick Baudisch. 2007. Shift: a technique for operating pen-based interfaces using touch. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, 657–666. Retrieved June 24, 2016 from http://dl.acm.org/citation.cfm?id=1240727
25. Daniel Vogel and Patrick Baudisch. 2007. Shift: a technique for operating pen-based interfaces using touch. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, 657–666. Retrieved June 24, 2016 from http://dl.acm.org/citation.cfm?id=1240727
26. Amanda Williams, Shelly Farnham, and Scott Counts. 2006. Exploring wearable ambient displays for social awareness. In *CHI'06 extended abstracts on Human factors in computing systems*, 1529–1534. Retrieved June 24, 2016 from http://dl.acm.org/citation.cfm?id=1125731
27. Robert Xiao, Gierad Laput, and Chris Harrison. 2014. Expanding the input expressivity of smartwatches with mechanical pan, twist, tilt and click. In *CHI'14 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 193–196. https://doi.org/10.1145/2556288.2557017
28. Chao Xu, Parth H. Pathak, and Prasant Mohapatra. 2015. Finger-writing with Smartwatch: A Case for Finger and Hand Gesture Recognition using Smartwatch. 9–14. https://doi.org/10.1145/2699343.2699350
29. Hui-Shyong Yeo, Juyoung Lee, Andrea Bianchi, and Aaron Quigley. 2016. WatchMI: applications of watch movement input on unmodified smartwatches. 594–598. https://doi.org/10.1145/2957265.2961825