The Flat Finger: Exploring Area Touches on Smartwatches

Ian Oakley[1], Carina Lindahl[2], Khanh Le[3], DoYoung Lee[1], MD Rasel Islam[1]
[1]Dept. of Human Factors Eng. UNIST, Ulsan, Korea
[2]Technical University of Denmark, Kgs. Lyngby, Denmark
[3]Jacobs University, Bremen, Germany
{ian.r.oakley, carinalindahl1990, vankhanh.le1, ehdud611}@gmail.com, islam@unist.ac.kr

ABSTRACT
Smartwatches are emerging device category that feature highly limited input and display surfaces. We explore how touch contact areas, such as lines generated by flat fingers, can be used to increase input expressivity in these diminutive systems in three ways. Firstly, we present four design themes that emerged from an ideation workshop in which five designers proposed concepts for smartwatch touch area interaction. Secondly, we describe a sensor unit and study that captured user performance with 31 area touches and contrasted this against standard targeting performance. Finally, we describe three demonstration applications that instantiate ideas from the workshop and deploy the most reliably and rapidly produced area touches. We report generally positive user reactions to these demonstrators: the area touch interactions were perceived as quick, convenient and easy to learn and remember. Together this work characterizes how designers can use area touches in watch UIs, which area touches are most appropriate and how users respond to this interaction style.

Author Keywords
Smartwatch; Area touch; Shape touch; Input technique.

ACM Classification Keywords
H.5.2 [Information Interfaces and Presentation (e.g., HCI)]: User Interfaces—Input Devices and Strategies (e.g., mouse, touchscreen)

INTRODUCTION
Smartwatches are an emerging device category [3] promising unique benefits and features. For example, they are a readily availability platform for displaying messages and notifications [5] and their close coupling to the body supports advanced sensing of biological signals for applications such as health tracking [14] or biometric authentication [7]. However, although smartwatches possess a distinct form factor based on a small screen (typically 3cm square or less) firmly affixed to the body, the currently dominant techniques used to interact with them are heavily based on paradigms popularized on much larger handheld devices such as phones and tablets. At heart these involve touch screens that display rich graphical contents that users interact with via taps for selecting on-screen targets and simple gestures, such as directional swipes, for navigation. While these techniques are familiar and effective, much of the richness and value they brought to interaction with handheld devices, such as the scale to select many options, the potential for two-handed input [28] and the expressive power of multi-touch gestures [24], simply do not fit on watch screens. They are just too small.

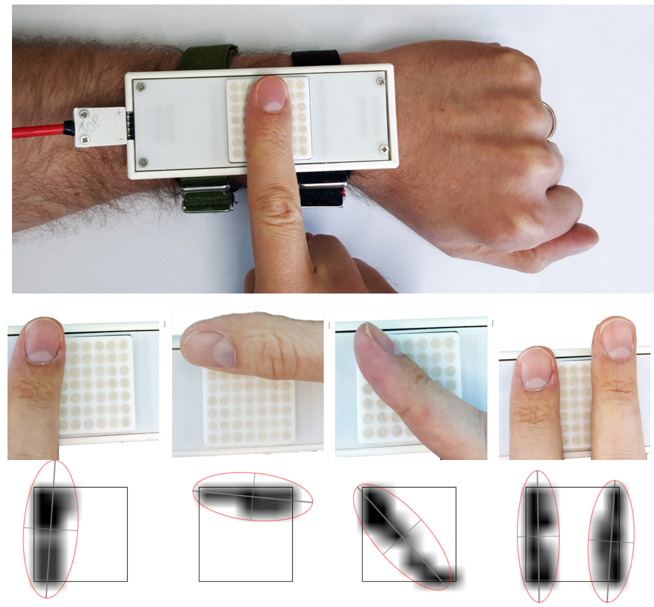


Figure 1. Top shows wearable prototype – the raised central square is the sensor. Bottom image pairs show area touches (index, thumb, side of index and index and middle fingers) and corresponding sensor activations.

Reflecting this observation, device manufacturers and researchers are exploring alterative input mechanisms to increase the richness of physical input on wearables. The Apple Watch (www.apple.com/watch/), for example, features pressure sensitive input and a side-mounted dial used to perform zooming and scrolling. The Samsung Gear S2 (www.samsung.com/global/galaxy/gear-s2/) sports a broadly similar spinning front bevel. Researchers have investigated a much more diverse set of channels such as alternative touch surfaces integrated into a device's strap [21] or edge [19]. A second approach has been to examine the potential of onboard sensors capable of detecting input on the skin immediately surrounding a device [17] or in the

CHI'16, May 07-12, 2016, San Jose, CA, USA

DOI: http://dx.doi.org/10.1145/2858036.2858179

air above it [13]. Still others have explored more eclectic channels such as twisting or tilting the frame of a device [29], or eschewing touches entirely for gaze based input [8].

While much of this research is promising, and serves to highlight the need for input techniques designed expressly for wearables, we argue there remains strong potential in exploring how touch input can be customized, adapted and optimized for watch screens. This argument is motivated by the large body of research that has explored how to improve and enhance touch input on handheld devices by characterizing, quantifying and leveraging qualities such as hand-poses [10], thumb contact regions [1] and reachability profiles [6]. The range and diversity of this literature highlights the fact that there is much more to touch input than simply the center coordinate of a contact point on a screen – a more nuanced understanding of how touches are performed can greatly improve interaction with a device. We believe is as true for smartwatches as it is for phones and, indeed, note that work to explore the unique properties of touch input on watches has already begun. In the area of text entry, for instance, researchers are optimizing procedures to enlarge target sizes via iterative zooming [18] or selective scrolling [15] to support rapid, reliable input. Similarly, research has begun to explore how to enrich single digit input by detecting finger orientation [16] and to investigate what kinds of multi-touch input are practical and feasible on watch-sized screens [20].

Continuing these efforts, the work in this paper adapts the idea of contact area input [4] to the smartwatch form factor. We make the following contributions: we describe the types of contact area touches users can produce on a small wrist mounted screen; we present the outcomes of a workshop ideation session in which five academic designers generated interaction concepts for watch area touches; we describe the results of an empirical user study using a bespoke touch sensor (see Figure 1) that explores the efficiency and reliability of performing watch area touches and; we create three demo applications based on these outcomes and report user reactions. Together this is comprehensive exploration of smart watch area touches that future designers can apply to their devices, applications and interfaces.

RELATED WORK

The use of contact area as an input modality is well established. Early work on this topic was motivated by the contrast between the paucity of the fingertip taps with which we interact with computer touch surfaces and the rich and diverse ways we hold, manipulate and control real world tools. Inspired by this idea, Cao *et al*. [4] constructed a tabletop system that used optical tracking to capture hand contact regions. These shapes were integrated into a physics model that associated contact region size with weight. In this system, users could flexibly push, gather or shove on-table graphical objects with their hands. They could also activate, slide or spin objects via a friction-like effect - by first touching them with a large contact region and then moving. Wigdor *et al*. [27] extended these ideas with their discussion of how area gestures could enhance interaction on tabletops. They present three gestures: rocks (a fist), rails (the edge of the hand with fingers extended) and curved rails (the edge of the hand with fingers bent). They discuss how these primitives can enhance the fundamental graphical operations of translating, rotating and scaling on-screen objects. Related ideas have appeared in commercial devices – Samsung Galaxy phones, for example, support a palm swipe gesture involving the edge of the hand moving across the screen to capture an image [30].

Other authors have explored contact area input at a smaller scale. Wang *et al*. [26], for instance, observe that finger touches on a tabletop are typically elliptical along the finger. By tracking these ellipses, they can deduce the angle of the hand in order to associate finger touches with specific users and create novel widgets such as an orientation sensitive pie menu. Boring *et al.* [1] developed this idea for mobiles by exploring how the size of the elliptical contact region of a thumb touch can be used to switch between modes of operation, such as between zooming and panning a map. Equally, there is a considerable body of work that looks at how finger orientation prior to touch [16] or finger roll after touch [23] can be used as an input technique. Finally, Rogers *et al.* [22] demonstrate that tracking finger angle can improve pointing performance on a phone and outline how this could be applied to create interfaces that, for example, automatically adjust for finger occlusions or rely on finger pitch to scroll through menu options.

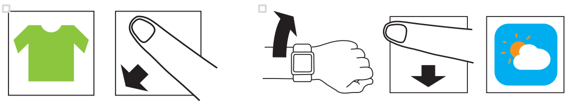
The work in this paper builds on this literature by applying the idea of touch regions to the form factor of a smart watch. We argue this is worthwhile as many of the assumptions about touch input derived from larger devices such as tables or phones, do not apply to the diminutive watch form factor. For example, the small screen size makes prior systems based on full hand input [27] or rich graphical feedback [4] infeasible. Equally, thumb input [1] on a watch is unusual; instead the index finger is dominant. This fact, in conjunction with the fixed mounting point on the wrist, precludes and affords unique types of motion – for example, rotations around the screen surface [26] involve a laborious full arm movement, while pitch adjustments can be achieved simple via finger flexes. Identifying this new context as a design opportunity, the remainder of this paper seeks to characterize the scope, value and potential of area touch input for smartwatches.

SMARTWATCH AREA TOUCH INPUT

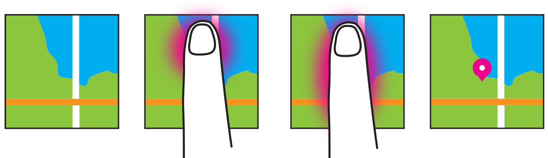
We began to explore this idea by generating a large set of 350+ area touches that could be realized on smartwatches. These included touches made by one to three fingers in a full range of orientations and considered both static touches and dynamic touches that change over time as fingers are moved. We then filtered this list based on four criteria. First, touches that were achievable in comfortable physical poses. Second, static touches that involved the extremities

Abstract Shortcut: Airplane mode is activated and de-activated with a line area touch moving up or down the screen.

Iconic Touches: A vertical finger triggers a mute operation (left). Two vertical fingers trigger heart rate monitor (right).



Spatial Metaphors: An item of clothing is dragged to a shopping basket by sweeping towards the body with a finger (left). The weather data is updated by raising the watch to vertical and sweeping a finger downwards (right).



Action Metaphors: A pin is dropped on a map by moving from a single point of contact to a line area touch.

Figure 2. Example interaction ideas from the design workshop for the themes of Abstract Shortcuts, Iconic Touches, Spatial Metaphors and Action Metaphors.

of the screen. This was to balance the idea of touching a screen area with the fat-finger [25] problem of this obscuring the underlying graphical contents – the goal was to select forms where there would be regions around the touched areas that could still be observed. Third, dynamic touches that fell in three categories: translations (movements of areas around the screen), transitions (changes from one touch area to another) and rotations (touch areas that spin). Finally, we included a small number of candidates that involve two simultaneous touches.

In total we retained 31 area touches based on Horizontal (H), Vertical (V) and Diagonal (D) lines. These represent a wide range of possible touch types. The final set included: eight lines (3H, 3V & 2D); four corner lines (D); three pairs of lines (H, V & D); eight line movements (perpendicular directions from H, V and both D lines), six transitions (changes from a point made by a finger-tip to a line made by a flat finger and vice versa in H, V and D forms) and; two rotations (from D to H and D to V). We also developed graphical icons for the area touches (see Figures 4 and 6).

AREA INPUT DESIGN WORKSHOP

Five academic designers (two professors and three students, two female) participated in a three-hour workshop to generate concepts related to area-input on smartwatches. The workshop began with an icebreaker and explanation of general guidelines and structure. Participants' first activity was to create a collection of useful or valuable tasks or functions that wearable or mobile devices can perform (e.g. take a photo, pay for services, display weather information). Mobile devices were included to increase the range of ideas that would be generated and participants were encouraged to list both existing and prospective new features. All content was generated in a brainstorming style session and noted down on small pieces of card. A second brainstorming session had participants create a set of interface types that are available on mobile and wearable devices (e.g. a map or canvas, a grid of icons, a list).

Participants were then introduced to the set of area touches and asked to create interaction concepts by merging functions, interface types and area touches. The concepts were sketched on 3cm square cardboard tokens and participants were given origami watches to wear. The tokens could be clipped onto the watches, affording simple, rapid physical prototyping of concepts. Ideas that gained traction in the group were documented in a short storyboard and pinned to a whiteboard. In total 40 storyboards were created. We synthesized the results of these ideas into the following four design themes. Figure 2 shows examples, in the form of finalized versions of the sketches generated in the workshop. The themes are intended as an early-stage design resource that highlights diverse, appropriate or interesting ways that area touches can appear on watches.

Abstract Shortcuts: Efficiency and ease of access to functions was a commonly expressed design theme. For example, one concept simply involved covering the corners of the screen to trigger different favorite apps. Others toggled functions, such as airplane mode or Bluetooth connectivity, through specific area touches or movements.

Iconic Touches: Participants also used the physical shape of particular touches as a resource for their ideas. For example, a finger placed vertically along the center of the watch resembles a finger across the lips and was associated with activating a mute function. Similarly, two vertical fingers were proposed to represent both checking heart rate and pausing media playback due to the similarity of the two-finger pose to, respectively, the finger posture used when checking the pulse on the wrist and the visual form of a traditional pause icon (two vertical bars). By leveraging prior knowledge, these metaphors have the potential to aid users in learning to use area touches.

Spatial Metaphors: A watch has a relatively fixed physical relationship to the body and world and participants used this to design dynamic behaviors. For example, saving, storing or accepting an item, message or contact was associated with dragging a line towards the body while sending or rejecting content was associated with moving a line away from the body. Participants also spontaneously proposed spatial metaphors combining area touches with other sensor input, such as raising the watch to vertical and moving a horizontal line downwards – pulling down the weather to get an updated forecast.

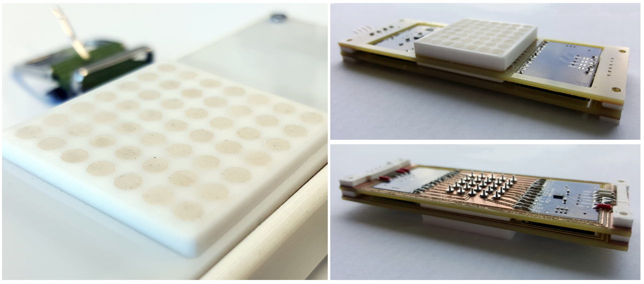


Figure 3. Left image shows close up of 33m square sensor area standing 3mm proud of the surrounding surface. Darker circles are the touch sensor electrodes. Right two images show the device outside its casing from top (upper image) and bottom (lower image).

Action Metaphors: The dynamic area touches, and particularly the transitions (from lines to points and vice versa) were also seen as resembling general-purpose interface actions. For example, moving from a point touch to a line touch was proposed as a mechanism for dropping a pin on a map or pasting content from a clipboard. Equally, moving from a line to a point picked up, removed or copied content. In terms of static touches, sustained contact over content such as an app was proposed as a way to erase it.

PERFORMANCE STUDY

Beyond establishing these design directions, a key objective of this paper was to assess how users practically perform contact area input on small wrist mounted wearables. In order to do so we required a watch form-factor touch sensor capable of reporting raw individual sensor activations at a typical spatial resolution (e.g. approximately 4-5mm [11]) and without preprocessing or filtering designed to exclude large or sustained contact areas, low level techniques commonly implemented in commercial devices (e.g. for palm rejection). We were also interested in touches that extend to the edge of the device as we believe that tactile cues from finger contact with this device rim will be an important factor in achieving good user performance. As such we wanted a sensor with a small bevel that was completely flush with the touch surface. However, current commercial smartwatches typically feature either relatively large bevels (e.g. 6mm in the Apple Watch) or bevels that protrude vertically above the screen (e.g. Samsung Gear S2). The requirement also precluded use of the common prototyping technique of using section of a mobile device as a surrogate for a wearable [18] – a phone screen will not have perceivable edges in the appropriate locations. Reflecting these problems, we opted to develop a bespoke touch sensor to achieve our desired form factor and ensure access to raw, unfiltered touch data for analysis.

Watch Prototype

The prototype touch sensor is shown in Figure 3. It combined four Sparkfun MPR121 breakout boards each with 12 independent capacitive electrodes for a total of 48 sensors. The boards were mounted in two vertically stacked bespoke PCBs that arranged the sensors in a seven by seven grid with a spacing of 4.5mm. This led to 49 sensor locations, so the center grid point was not connected – we were largely focused on area touches at the device edges. On top of the PCB boards we mounted a watch-sized piece of white acrylic – a 33mm by 33mm by 5mm square. Three mm holes were milled in the center 30mm square region of the acrylic, one situated directly over each electrode. This led to a bevel of 1.5mm at the edge of the device. The two PCBs and acrylic were then electrically connected by M1 bolts. The holes in the acrylic were filled with an off-white conductive dough and, when dry, sanded to yield a smooth, flush finish. Finally, a 2mm piece of acrylic was placed over the surface of the top PCB and around the 5mm high watch surface – leaving the sensor piece standing 3mm proud – and the whole unit enclosed in a 108mm by 43mm by 15mm 3D printed case that featured two watch bands in order to enable it to be securely fastened to a user's wrist without protruding over the sides (Figure 1, top).

This device was connected to an Arduino Nano and the MPR121 boards were configured to use default onboard filters and calibration routines and an update rate of 16ms. The Arduino polled the MPR121 boards for baseline and measured capacitance values as rapidly as possible: 25Hz. It then calculated the proportion between these figures to approximate touch magnitude – light contact yields near-baseline values while firmer touches result in greater differences. This hardware showed low noise (less than 1% variation in the measured capacitance for each electrode over a five-minute period) and uniform performance (light contact with a metal probe resulted in 2%-3% change in measured capacitance for each electrode). A light finger touch led to a 2% change in capacitance and a moderately strong touch led to a change of 12%.

The Arduino reported the proportional capacitance values over an RS232 link to a Java application on a host PC. This application processed the sensor data to yield touch areas in the form of ellipses – an approach directly derived from closely related prior work using cameras [26]. We followed a typical process. First, we assigned a value to the center grid location based on the mean of its neighbors and ignored variations of less than 1%. Second, we generated a scaled up gray scale image from the remaining data (for visualization) and used a fill based blob detection algorithm to isolate touch areas. We then simplified the resultant polygons to a maximum of 20 vertices and used Fitzgibbon *et al.*'s [9] direct least squares algorithm to fit ellipses around them and derive the centroids, angles and the lengths of major and minor axes. Figure 1 shows examples of touches, sensor activation patterns and the ellipses calculated from this data.

Study Design

The primary goal of this study was to characterize how participants produce different area touches on a smartwatch. This includes in terms of ease, comfort, accuracy and efficiency. A secondary goal was to determine the

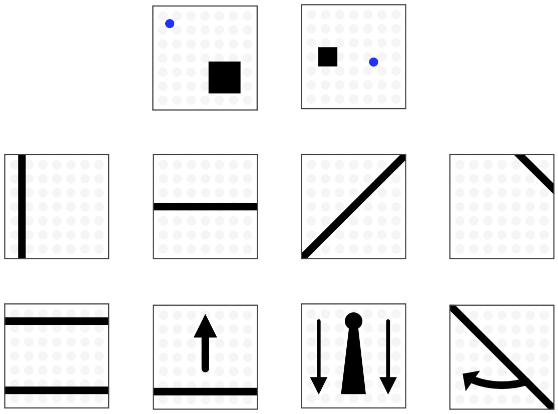


Figure 4. Experimental instructions. Top row shows the targeting tasks for the four-target (left) and nine-target conditions (right). The target is the black square and the cursor is the blue circle. Middle and bottom row show a sample of area touch icons. From left to right, the middle row shows: left vertical line; center horizontal line; bottom-left to top-right diagonal line; top right corner line. The bottom row shows: horizontal pair of lines; horizontal movement upwards; vertical transition from finger-tip to finger-flat; clockwise rotation.

distinctiveness of area inputs with respect to more traditional touch inputs, such as those resulting from tapping regular on-screen targets such as square icons. In order to achieve these objectives, we ran a multi-stage study composed of a set of basic targeting tasks, followed by training stages for participants to experiment with producing different area touches followed by assessment of the performance of these touches. The study used the set of 31 area touches described previously.

Twenty participants (ten male) completed the study. They were screened to be right-handed and were aged between 18 and 25 (mean 21). They were all students, recruited from online university message boards. They reported high levels of familiarity with computers (4.2/5), smartphones (4.4/5 but not wearable devices (1.1/5). They took an average of 50-70 minutes to complete the study and were compensated approximately 10 USD for their time.

Procedure

The study was primarily descriptive. Consequently, it featured a single condition completed by all participants. This condition involved five distinct stages presented in the following order: four-targets; nine-targets; area-exploration; area-production; and area-testing. Throughout the study participants wore the watch prototype on their left wrist and interacted with it using the fingers on their right hand. As the sensor unit did not feature a graphical display, participants were seated in front of a computer screen that displayed all experimental instructions. This enabled them to move both hands freely, as with a real watch. Figure 4 shows example instructions shown on the PC in the study.

In the four-target and nine-target stages, participants completed a standard button selection task. Each stage featured trials as follows: first a message on the PC instructed participants to touch the sensor to start. A black target and a fixation spot were displayed in a bounding box representing the absolute input space of the sensor. After 500ms the fixation spot disappeared and the trial began. Subsequent touches to the sensor caused a round cursor to be displayed around the centroid of the detected ellipse – the participant's task was to move the cursor over the black target and release (see Figure 4, top). They then received feedback as to the correctness of their input in the form of a tick or cross and a new trial began. In the four-targets stage, the targets occupied 11.25mm by 11.25mm of the sensor while in the nine-targets stage they were 6.66mm square. In both stages, targets were positioned in a grid (2x2 or 3x3, respectively) that started 2.5mm from the edge of the sensor and featured 2.5mm gaps between targets. The goal of these stages was to familiarize users with the sensor and capture baseline performance data for typical targeting tasks. Participants successfully completed 12 trials in the four-target stage and 27 trials in the nine-target stage – three for every target in both stages. Within each stage, the targets were shown in a random order. In both stages, the first third of the trials were discarded as practice leaving, respectively, 8 trials and 18 trials per participant for analysis.

In the area-exploration stage, participants were presented with a sheet of paper showing all 31 area touches used in the study. This included a textual description and a graphical icon that was identical to the instructions used later in the study. They were able to make any area touch on the sensor and see how the system classified their input (see next section). They were also encouraged to ask the experimenter questions about the area touches or how to make them. They were asked to make each of the 31 area touches two or more times and allowed a maximum of ten minutes in this stage. No data was captured in this stage.

In the area-production stage, participants experienced each of the 31 area touches in a random order. For each area touch they were required to successfully complete 12 trials that followed the same structure as the earlier targeting stages: tap to begin followed by instructions, fixation, input and finally feedback as to their success or failure. The input period differed in that no cursor or instructions were shown on screen. Instead, participants were encouraged to focus on making the correct area touch on the sensor in the absence of direct visual cues. We believe this is appropriate as most interaction scenarios from the design workshop did not involve area touches that correspond to on-screen elements – such graphics would simply occupy to much of the watch screen to be feasible. While completing this stage, participants were asked to finalize a technique for making each area touch in the early trials and use the final trials for rapidly and reliably issuing the touch. The first four repetitions for each touch were treated as practice and not analyzed. At the end of each block of 12 trials, there was a break and participants illustrated how they performed the area touch to an experimenter. The goal of this stage was to assess how participants opted to make each area touch and the reliability and efficiency with which they could achieve this. The repetitions were used to lower the impact of the high cognitive load implied by the large cue set size on the actual physical performance of the input. In this way, we argue data from this stage more closely approximates expert use. In total, in this stage, 248 trials per participant were retained for analysis.

In the final area-testing phase, participants were exposed to three trial blocks, each featuring one repetition of the full set of 31 area touches delivered in a random order. Individual trials were identical to the area-production stage. The goal was to assess how quickly the system could be learned and used in a more practical scenario in which any area touch might be employed at any time. In this stage, the first block was considered practice and discarded, leaving a total of 62 trials per participant for analysis.

Area Touch Classifier

In order to match user's input to specific area touches, we first selected ellipses to examine. For static areas (including points) and touches longer than 150ms, we used ellipses captured 75ms before releasing the sensor. For touches shorter than 150ms, we used the ellipses recorded in the middle of the touch. We used this threshold as touch areas produced during the initial and closing moments of contact can vary in size and shape [26] and we wanted to capture stable representations of participant's intended touch regions. For dynamic area touches, the start position classification used the ellipses captured 75ms after the initial touch while the end-position classification used the ellipses detected 75ms before release of the sensor.

We developed a simple decision-tree to classify the ellipses into the 31 area touches in the study. First each ellipse was set as either a long or regular touch based on whether the length of its major axes exceeded a threshold of 80% of the sensor size (24 mm). For long lines, touch orientation was classified as vertical, horizontal or at 45° using a symmetric 20° window around these values. A 30° window was used for regular touches, as it was more challenging to accurately produce desired angles with shorter touches. The centroid of each touch was classified by thirds in both x (left/center-x/right) and y (top/center-y/bottom) axes. Finally, all dynamic areas touches were checked before classifying static touches. These dimensions provided sufficient power to uniquely identify all 31 area touches. We note the thresholds were determined via subjective, iterative testing and the system not intended to achieve optimal recognition performance, but rather serve as a simple, tolerant way of supporting capture of participants' area touch input.

Measures

Before the study began, participant hand size was captured via measurement of the span with a ruler and the width of the thumb and each finger at the last joint with calipers. All trials in all stages of the study featured the same objective measures. We logged *preparation-time*, the moment from when the fixation spot disappeared to first contact with the sensor and *touch-time*, the duration between first and last contact with the sensor. We also logged the outcome of our area touch classifier and the raw ellipse data used in this algorithm: the centroid, angle and the length of the major and minor axes. During the area-production stage we also recorded how participants touched the device immediately after completing each area touch. This was achieved via an experimenter asking participants to demonstrate their preferred touches and noting the digit used, the orientation of that digit (flat or edge) and the cardinal or intercardinal direction of the touching hand relative to the watch.

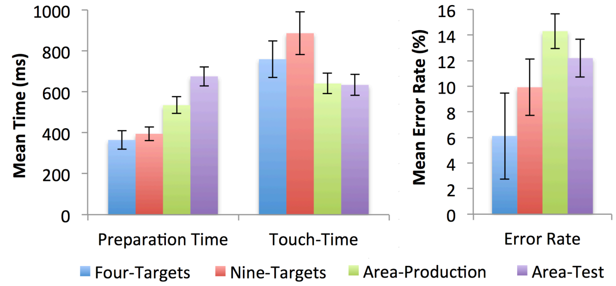


Figure 5. Mean task times and error rates from the four stages in the performance study. Bars show standard dev.

Results

Time and error data from the study are shown in Figure 5. Repeated measures ANOVAs, incorporating Greenhouse-Geisser corrections if sphericity was violated and followed by *post-hoc* t-tests with Bonferroni corrections, were used to explore differences. Preparation-time showed a significant trend ($F (3, 57) = 14.2, p<0.01, \eta_p^2 = 0.427$) with a moderate effect size. *Post-hoc* tests indicated that preparation-times in the four-target and nine-target stages were significantly faster than the area-test stage (both $p<0.001$). Touch-time showed no significant trend ($F (1.37, 26) = 2.65, p = 0.11, \eta_p^2 = 0.12$). Error data in the four-target condition was highly non-normal (the median was zero), so this stage was excluded from the analysis. No significant differences were observed among the remaining three stages ($F (1.46, 27.7) = 2.88, p = 0.087, \eta_p^2 = 0.13$).

Beyond these comparisons, the main goal of this study was to characterize performance of the individual area touches. Figure 6 shows a breakdown of the data from successful trials in the area-preparation stage. It is divided by area touch, with graphical depictions of means and standard deviations for each ellipse position, size and angle and numerical data showing mean times and error rates. We highlight the high stability and low variance of this data in terms of position, size and angle. The finger poses used during the area touches were also an important aspect of this characterization. We found 69.2% of touches involved the index finger, with thumb, middle and pinky accounting for 6.9% 6.4% and 6.1% of touches. Those area touches requiring two simultaneous touches all took place with the index and middle finger (9.8% of total). The remaining 1.4% took place using diverse hand and finger areas. The

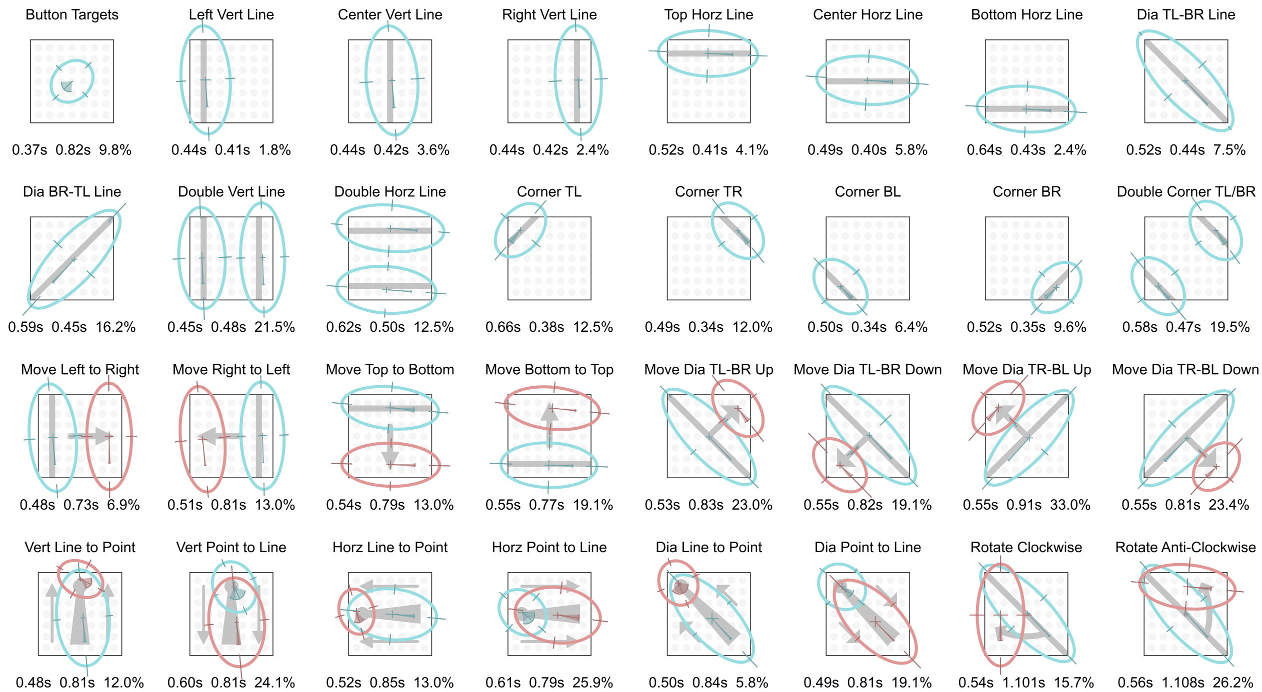


Figure 6. Contact areas captured during successful trials. Top left thumbnail shows mean ellipse size and angle from targeting stages. The remaining thumbnails illustrate the area touches participants were asked to produce (light grey background) in the area-production stage. Captured ellipses are shown to scale and illustrate mean position, size and angle. For dynamic touches, blue ellipses indicate start touch and red ellipses end touch. Standard deviation shown via bars for position and size and wedge for angle. Numbers at the base of each thumbnail show mean preparation-time (left), mean touch-time (center) and mean error-rate (right).

flat of a finger was used in 65.2% of touches and the edge in 34.8%. The right hand was predominantly located in a natural position between the users body and the watch – either to the right of the watch (24.4%), the bottom-right (33.2%), the bottom (29.4%) or the bottom-left (9.8%). We suggest this diversity indicates participants readily adopted a wide range of different hand and arm poses in order to most comfortably execute the area touches.

We examined the hand measurements to assess if physical size impacted performance. Mean hand-span was 202mm (SD 8.9). From thumb to pinky, finger widths were 18.1mm, 14.1mm, 14.3mm, 13.2mm and 11.6mm. They were highly correlated (Pearson's R 0.71 to 0.89 and $p<0.001$) so were combined into a single data-point: a mean finger width of 14.3mm (SD 1.05). Span correlated weakly with finger width ($R = 0.23$, $p = 0.36$) so was retained as a separate measure. We then ran two multiple regressions with span and mean finger width as predictors and the outcome variables of touch-time (R^2=0.158, $F(2,19)$ =1.58, p=0.23) and error rate (R^2=0.114, $F(2,19)$ =0.79, p=0.47) captured from the area-production stage. Both show small effect sizes and neither revealed significant relationships.

Discussion

The study used a bespoke and unproven sensor unit. To validate its performance, we contrasted data from the targeting stages to prior work. Specifically, we examined temporal performance against that reported in Leiva *et al.*'s [18] study of the ZShift watch keyboard. This is because the ZShift task is analogous – it involves an initial touch to select a target that is refined via a cursor shown on a callout. With targets of 2.85mm, Leiva *et al.* report 9.1 Words-per-Minute, or a target selection time of 1319ms. This figure is broadly comparable with the 1124ms and 1281ms observed for 11.25 and 6.66mm targets in the current study. We note that although Leiva's targets were smaller (and likely slower), their continuous typing task also likely lowered the contribution of preparation time to the data. As such, we believe the time data indicates the sensor unit performed adequately. We performed a similar examination of the error data, comparing the current results with the 12.96% (7mm targets) to 2.22% (10mm targets) figures reported in Hara *et al.*'s [12] recent study of selecting buttons on smartwatches. Again we note these data are broadly equivalent to the 9.9% (6.66mm targets) and 6.1% (11.25 targets, including a large rightward skew) recorded in the current study. In sum, we believe these comparisons indicate that our sensor performs sufficiently well to support studying touch input on watches.

We then moved on to examining the area touch data. The time data generally supports the viability of area touches as a watch interaction technique. Although preparation time in the area-test stage was a mean of 295ms slower than in the two targeting stages, no further differences were recorded. We believe this increase is due to the high task variability implied by the 31 different and randomly delivered trials in the area-test stage – it took time to process this information.

A detailed look at the individual touch-times depicted in Figure 6 also shows the mean touch-time data obscures a binary distinction between the static (mean 422ms, SD 49.6) and dynamic (mean 846ms, SD 88.8) area touches. This is unsurprising: the static areas are single touches, while the dynamic areas require movement between two areas. However, we note the static area touches take roughly half the time of target selections. This indicates they represent simple, rapidly executable input primitives – it is faster to perform them than to select on-screen targets.

In terms of the error data, performance with the area touches was less reliable than that in the four-target stage and equivalent to that in the nine-target stage. Given the much larger number of 31 area touches used, this suggests the technique may be a more reliable way of increasing the expressivity of input on smartwatches than simply shrinking target sizes. The breakdown of error data from the area-preparation stage in Figure 6 provides a more nuanced picture of this performance. Error rates varied substantially with area touch – the most reliably performed touches approached zero (1.8%) whilst the least reliable had rates high enough to preclude their use in an interactive system (33%). To explore these differences in more detail, we examined the data in three ways: median errors per area touch; error rate by number of successful trial completions and; area touch misclassifications. Median results revealed strong positive skews – 13 of the area touches had a median error rate of zero and the median for the reminder was an average of 66% of the mean value. This indicates that small numbers of participants were disproportionately responsible for the high mean error rates of different area touches. We suggest this is partly attributable to experimental artifacts such as presentation order, practice or fatigue that would be less evident during real world use.

Dividing the error data by the number of successfully completed trials supports the presence of a practice effect. Participants produced a mean of 0.211 errors (SD 0.181) directly before successfully completing a first area touch. Immediately prior to the final area touch (after seven successes) they produced a mean of 0.145 errors (SD 0.094), a reduction of 31.3%. Although a regression on this data did not confirm this as a significant trend (R^2=0.482, F (1,7) =5.581, p=0.056), the medium to large effect size and the low number of possible observations (eight) suggest this is a type II error and that participants' performance was improving steadily. This observation is borne out in mean error rates for the area-test stage. They modestly, but non-significantly, improve on performance in the production stage even though participants were performing a substantially more challenging task in which any of 31 possible area touches needed to be produced on demand.

Examining misclassifications also shed light on performance. In total 857 errors were recorded. We excluded touches longer than 24mm (long lines in the classifier) from triggering target selections and found 477

Area Touch Recorded	Area Touch Requested (occurrence count shown in brackets)
Left Vert Line	Rotate Clockwise (11)
Center Vert Line	Double Horz Line (5), Vert Point to Line (34)
Top Horz Line	Move Bottom to Top (16), Rotate Anti-Clockwise (21)
Center Horz Line	Double Vert (12), Horz Point to Line (23), Horz Line to Point (8), Move Top to Bottom (5)
Bottom Horz Line	Move Top to Bottom (13)
Dia TL-BR Line	Double Horz Line (5), Dia Point to Line (18)
Corner TL	Move Dia TR-BL Up (52)
Corner TR	Move Dia TL-BR Up (22)
Corner BL	Move Dia TL-BR Down (14)
Corner BR	Move Dia TR-BL Down (30)
Button Targets	Corner TR (5), Corner BR (12), Vert Line to Point (6), Horz Point to Line (11), Dia Point to Line (7), Move Dia TR-BL Up (6), Rotate Anti-Clockwise (5)

Table 1. Misclassifications in area-preparation stage (data only shown when exceeding 2.5% of successful trials).

(55.6%) touches were misclassified as either another area touch or as a selection of one of the buttons in the two targeting stages. We were not able to classify the remaining 380 touches due to, for example, short duration or the presence of multiple contact areas. The misclassifications were broadly spread, so are summarized in Table 1. All misclassifications that occurred more than 4 times (2.5% of the number of successful trials recorded) are shown. This represents 71.4% of the misclassifications. This data reveals two key things. Firstly, misclassifications as button targets were infrequent –just 81 in total (1.2% of total trials). This indicates that area touches are unlikely to interfere with traditional targeting operations. Secondly, the data highlights limitations of the sensor and classifier. The most frequent misclassifications of dynamic touches, such as a diagonal line that moves to one of the corners or a transition from a point touch to a line touch, are in the form of the static touch that represents the end state. This suggests that many of the errors stem from participants performing the area touch too rapidly to be detected by our 40Hz hardware and the 75ms timing threshold used in the classification algorithm. Improvements to the sensor and classification system would likely resolve many of these error cases.

The demographics, pose data and general stability of the touch areas shown in Figure 6 are highly supportive of the viability of area touches for smartwatches. Participants used different fingers (or finger areas) to make a wide range of area touches whilst keeping the hands in a comfortable position relative to one another. The resultant touches show high accuracy and low variability in terms of location, size and angle. Furthermore, finger/hand size did not predict performance, suggesting the technique is viable for a broad range of users. That said, the data shown Figure 6 does show trends suggesting which area touches are the most promising candidates for deployment in real interfaces. Specifically, we note that specific types of touch showed higher variance in particular data than others, suggesting they may be more difficult to reliably produce. For rotation angle, touches showing this effect include short touches (up to four times greater) and the final positions of dynamic touches (40% up). Horizontal touches showed similarly increased variance over vertical touches in terms of major axis length (20%). Differences also appear in specific types of touch, perhaps most prominently in the diagonal touches; the top-left to bottom-right diagonals were performed with 39% lower error rate than the inverse. Finally, some forms of touch, such as the rotations were notably slower (26%) than other dynamic touches. These variations highlight touches that are more challenging to produce and can be used as a starting point for future studies or to recommend specific area touches for use in interfaces.

DEMONSTRATIONS APPS

We used the knowledge acquired in the design workshop and performance study to create demonstrators that deploy area touches on smartwatches. These prototypes used the sensing platform from the performance study augmented with a top-projected visual display (Figure 7a). Projector alignment was achieved by creating a fixed mounting frame for the sensor unit. This allowed a user to place his or her arm under the watch and was adjustable to accommodate different arm sizes. When placed at the edge of a desk, it allowed a comfortable pose with the user's non-dominant wrist on the desk and the watch held stationary above it. However, after projector alignment, the sensor unit could not be moved. In contrast to the free motions possible in the previous study, this static pose impacted the ease or comfort of making some of the area touches. We designed three prototypes, each implemented in Java. We explored themes from the design workshop and focused predominantly on the area touches that were executed rapidly and reliably in the performance study. We were also sensitive to the flat, fat fingers [25] our technique entails and designed visuals either around covered areas [22] or for eyes-free interaction [2]. Finally, we sought to integrate area touches with traditional input styles such as taps and common widgets such as lists and grids of icons.

Watch Face Shortcuts: This application (Figure 7b) explored area touches as shortcuts – a key theme from the design workshop. The prototype showed a standard watch face with time and weather data and responded to area touches in the form of fingers placed along its four edges. A finger covering the right edge displayed a *glance* – a screen of content designed for viewing no further interactions. During the touch, the glance screen slid in rightwards to display on the left three-quarters of the screen, away from covering finger. When the finger was removed, the glance slid back off-screen. The goal was to allow users to rapidly view an extra screen of personalized contents – our example showed a to-do list, but other possibilities include sports results, news or social media feeds.

Three other forms of shortcuts were explored. A finger along the bottom of the device caused a menu containing a grid of four app icons to be displayed. These could be

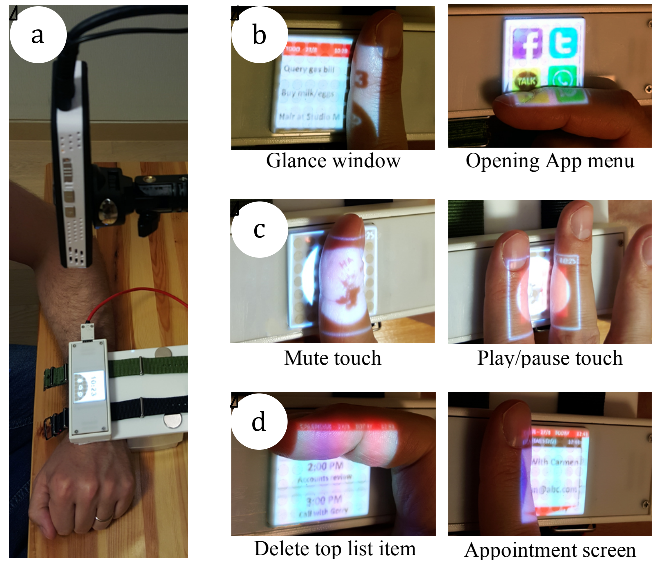


Figure 7. Projector setup can be seen in (a). Image sections (b), (c) and (d) respectively show area-touches from the Watch Face Shortcuts, Media Player and Calendar Apps.

triggered with a standard tap, or the menu removed by a second area touch along the bottom of the device. An inbox application that operated in much the same way was triggered by an area touch along the left edge of the device. Messages could be selected with taps and the app closed by re-issuing the left edge area touch. These designs used area touches as shortcuts to activate typical smart watch interfaces. Finally, an area touch along the top of the device toggled Bluetooth mode, showing how area touches could provide quick access to specific features or commands.

Media Player: The media player application (Figure 7c) showed a simple animation. Area touch interactions were inspired by the metaphors in the design workshop. A finger placed vertically down the center of the watch toggled the mute function and two vertical fingers toggled between play and pause. These interactions were inspired by the symbolism of a finger across the lips for silence and the iconic similarity of two fingers and the two vertical bars of a pause icon. A final interaction involved an area touch in the form of a horizontal line along the bottom of the screen toggling subtitles. This idea is simply based on covering the screen area that displays the subtitle content.

Calendar: The final application was a calendar (Figure 7d) – a list view showing three appointments from a single day. Taps would open appointment to show more information. Area touches performed a range of functions in this system. Two types of navigation were created. In the list view vertical line area touches on the left and right navigated to the previous and next days. On the appointment pages, vertical lines moving from left to right and right to left across the watch face executed a panning action to see more contents. The goal here was to explore how both static and dynamic area touches could support navigation actions.

Area touches also issued different commands. In the list view, covering an appointment with a horizontal line deleted it – obscuring content to erase it. In the appointment view, downward movement of a horizontal line pushed the appointment off-screen and returned to the list. A message screen showing an incoming event request could be accepted by moving a diagonal line area touch towards the body and rejected by moving the touch away. These interactions were inspired by the spatial metaphors proposed by participants in the design workshop. Finally, we also implemented simple clipboard operations based on the action metaphors from the workshop. A user could copy a currently viewed appointment with an area touch that transitioned from a line across the screen to a point – a movement achieved by picking up a finger from flat against the screen to just a fingertip on the screen. Back on the list view, they could move to another day and paste this data with the opposite movement, a transition from a point made by the fingertip to a line made by the flat of the finger.

Demonstration User Study

We conducted a short user study of these prototypes as an informal validation of area touch input for smartwatches. The goal was to explore how users responded to the design concepts identified in the workshop and expressed through the application prototypes and receive qualitative feedback on the area touch input technique. Eleven naive participants (all right-handed students, five female, mean age 21) spent 20-30 minutes completing the study in a quiet office. Each participant first received a demonstration: an experimenter donned the watch and illustrated each function. Participants then put on the watch, the top projection was recalibrated and they tried out the applications. The experiment was videoed and comments were captured. One limitation of this study is that the projection system constrained participants to a single fixed posture – this may have impacted their ability to make the area touches.

Participants reported valuing the speed, simplicity and convenience of the area touches, all making statements about how "fast", "easy" the techniques were or the "quick access" they afforded. There was also strong support for the use of metaphors through the designs. The media mute and pause commands were "intuitive" (P1, P3, P5, P11) and participants appreciated they "resembled [a] body gesture" (P2) and the pause symbol "that everybody knows" (P5) while the copy/paste commands were described as taking "information into the air" (P1, P5, P11) and "attach[ing]" it to the device (P1, P7). However, participants had concerns about the comfort of some of the area touches. Comments were most typically raised during the initial watch face demo and with the two horizontal touches – seven participants expressed concerns about comfort when performing these touches. These concerns subsided in subsequent demos, as participants grew used to the area touch technique. However, their initial prominence highlights the importance of conducting a subjective evaluation of watch area touches to formally assess factors such as workload or comfort as a next step for this work. More generally, participants were also concerned about the availability of, or metaphors used, in some of the techniques: copy/paste commands were "hard to notice" (P4) and the use of static vertical touches for page navigation did not "feel like turning [a] page" (P9). Several participants also worried about the number of area-touches they were exposed to – they saw potential for "confusion" (P11) or the need for "instruction the first time" (P4, P10). If area touches are to be deployed on real smart watches, it will require the creation of a clear and consistent set of interactions that facilitate learning and memorability.

CONCLUSIONS

This paper explored contact area input on smartwatches. It contributes a set of 31 candidate area touches, four design themes, a detailed empirical characterization of user performance and the design of (and user feedback on) three demonstration applications. The results are supportive – suggesting the area touches can be used for variety of purposes and that many are readily performed, easy to integrate with existing interfaces and well received by users. This data represents a rich resource that future designers can use to create their own systems. We also note there are benefits to the technique when compared to other approaches for increasing the expressivity of touches on smartwatches. Pressure input (e.g. Apple watch), for example, supports few unique input states; area touches are more diverse. Furthermore, unlike many other techniques to enrich touch input [13, 16] for watches, area touches rely on the screen sensor alone rather than requiring physical devices to be mounted on the touching finger.

Future work on this topic should first address the limitations of the sensing system used in this work. The hardware, while functional, was slow (40Hz) and constructed from atypical materials (sanded acrylic and dough) – commercial sensors under smooth glass would likely improve performance. As such, a clear next step for this work is to implement area touches on commercial devices either by developing bespoke drivers to report the required data [e.g. 11], or by leveraging and adapting existing API functionality (e.g. the latest versions of Google's Android OS report touch ellipses). This would provide a better integration with on-screen UIs and enable more detailed and ecologically valid investigations of the technique. Equally, the software classifier was trivial; a machine learning approach would yield better results. It would also be worthwhile to apply area touches to different sizes and shapes of watch – including those with round screens – in order to maximize the applicability of the technique to the next generation of smart wrist-ware.

ACKNOWLEDGEMENTS

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (2014R1A1A1002223).

REFERENCES

1. 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* (MobileHCI '12). ACM, New York, NY, USA, 39-48. http://doi.acm.org/10.1145/2371574.2371582
2. Stephen Brewster, Joanna Lumsden, Marek Bell, Malcolm Hall, and Stuart Tasker. 2003. Multimodal 'eyes-free' interaction techniques for wearable devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '03). ACM, New York, NY, USA, 473-480. http://doi.acm.org/10.1145/642611.642694
3. Business Insider. 2015. "The Wearables Report: Growth Trends, Consumer Attitudes, and Why Smartwatches Will Dominate." Business report. Business Insider (21 May 2015). Retrieved 03 Sept. 2015 from http://www.businessinsider.com/the-wearable-computing-market-report-2014-10
4. Xiang Cao, A.D. Wilson, R. Balakrishnan, K. Hinckley, K and S.E. Hudson. 2008. ShapeTouch: Leveraging contact shape on interactive surfaces. In Proceedings of the 3rd IEEE International Workshop on Horizontal Interactive Human Computer Systems (TABLETOP 2008). IEEE Computer Society, Washington, DC, USA, 129-136 doi:10.1109/TABLETOP.2008.4660195
5. Marta E. Cecchinato, Anna L. Cox, and Jon Bird. 2015. Smartwatches: the Good, the Bad and the Ugly?. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems* (CHI EA '15). ACM, New York, NY, USA, 2133-2138. http://doi.acm.org/10.1145/2702613.2732837
6. Youli Chang, Sehi L'Yi, Kyle Koh, and Jinwook Seo. 2015. Understanding Users' Touch Behavior on Large Mobile Touch-Screens and Assisted Targeting by Tilting Gesture. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (CHI '15). ACM, New York, NY, USA, 1499-1508. http://doi.acm.org/10.1145/2702123.2702425
7. The Economist. 2013. A Heart to My Key. News Report. The Economist Newspaper (09 May 2013). Retrieved 03 Sept. 2015 from http://www.economist.com/blogs/babbage/2013/05/biometrics
8. Augusto Esteves, Eduardo Velloso, Andreas Bulling, Hans Gellersen. 2015. Orbits: Enabling Gaze Interaction in Smart Watches using Moving Targets. In *Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (UbiComp '15). ACM, New York, NY, USA, 617-621. 419-422. http://dl.acm.org/citation.cfm?doid=2800835.2800942.
9. Andrew Fitzgibbon, Maurizio Pilu, and Robert B. Fisher. 1999. Direct least square fitting of ellipses. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 21, 5: 476-480.
10. Mayank Goel, Jacob Wobbrock, and Shwetak Patel. 2012. GripSense: using built-in sensors to detect hand posture and pressure on commodity mobile phones. In *Proceedings of the 25th annual ACM symposium on User interface software and technology* (UIST '12). ACM, New York, NY, USA, 545-554. http://doi.acm.org/10.1145/2380116.2380184
11. Christian Holz, Senaka Buthpitiya, and Marius Knaust. 2015. Bodyprint: Biometric User Identification on Mobile Devices Using the Capacitive Touchscreen to Scan Body Parts. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (CHI '15). ACM, New York, NY, USA, 3011-3014. http://dx.doi.org/10.1145/2702123.2702518
12. Hara, Kiyotaka, Takeshi Umezawa, and Noritaka Osawa. 2015. Effect of Button Size and Location When Pointing with Index Finger on Smartwatch. Human-Computer Interaction: Interaction Technologies. Springer Intl. Publishing, 2015. 165-174. http://dx.doi.org/10.1007/978-3-319-20916-6_16
13. Chris Harrison and Scott E. Hudson. 2009. Abracadabra: wireless, high-precision, and unpowered finger input for very small mobile devices. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology* (UIST '09). ACM, New York, NY, USA, 121-124. http://doi.acm.org/10.1145/1622176.1622199
14. Daniel Harrison, Paul Marshall, Nadia Bianchi-Berthouze, and Jon Bird. 2015. Activity tracking: barriers, workarounds and customisation. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (UbiComp '15). ACM, New York, NY, USA, 617-621. http://doi.acm.org/10.1145/2750858.2805832
15. Jonggi Hong, Seongkook Heo, Poika Isokoski, and Geehyuk Lee. 2015. SplitBoard: A Simple Split Soft Keyboard for Wristwatch-sized Touch Screens. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (CHI '15). ACM, New York, NY, USA, 1233-1236. http://doi.acm.org/10.1145/2702123.2702273
16. 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 (CHI '14). ACM, New York, NY, USA, 189-192. http://doi.acm.org/10.1145/2556288.2557258

17. Gierad Laput, Robert Xiao, Xiang 'Anthony' Chen, Scott E. Hudson, and Chris Harrison. 2014. Skin buttons: cheap, small, low-powered and clickable fixed-icon laser projectors. In *Proceedings of the 27th annual ACM symposium on User interface software and technology* (UIST '14). ACM, New York, NY, USA, 389-394. http://doi.acm.org/10.1145/2642918.2647356
18. Luis A. Leiva, Alireza Sahami, Alejandro Catala, Niels Henze, and Albrecht Schmidt. 2015. Text Entry on Tiny QWERTY Soft Keyboards. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (CHI '15). ACM, New York, NY, USA, 669-678. http://doi.acm.org/10.1145/2702123.2702388
19. Ian Oakley and Doyoung Lee. 2014. Interaction on the edge: offset sensing for small devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '14). ACM, New York, NY, USA, 169-178. http://doi.acm.org/10.1145/2556288.2557138
20. Ian Oakley, DoYoung Lee, MD. Rasel Islam, and Augusto Esteves. 2015. Beats: Tapping Gestures for Smart Watches. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (CHI '15). ACM, New York, NY, USA, 1237-1246. http://doi.acm.org/10.1145/2702123.2702226
21. 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* (CHI '13). ACM, New York, NY, USA, 1451-1460. http://doi.acm.org/10.1145/2470654.2466192
22. Simon Rogers, John Williamson, Craig Stewart, and Roderick Murray-Smith. 2011. AnglePose: robust, precise capacitive touch tracking via 3d orientation estimation. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11). ACM, New York, NY, USA, 2575-2584. http://doi.acm.org/10.1145/1978942.1979318
23. Anne Roudaut, Eric Lecolinet, and Yves Guiard. 2009. MicroRolls: expanding touch-screen input vocabulary by distinguishing rolls vs. slides of the thumb. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '09). ACM, New York, NY, USA, 927-936. http://doi.acm.org/10.1145/1518701.1518843
24. Ted Selker. 2008. Touching the future. *Commun. ACM* 51, 12 (December 2008), 14-16. http://doi.acm.org/10.1145/1409360.1409366
25. Katie A. Siek, Yvonne Rogers, and Kay H. Connelly. 2005. Fat finger worries: how older and younger users physically interact with PDAs. In *Proceedings of the 2005 IFIP TC13 international conference on Human-Computer Interaction* (INTERACT'05), Springer-Verlag, Berlin, Heidelberg, 267-280. http://dx.doi.org/10.1007/11555261_24
26. Feng Wang, Xiang Cao, Xiangshi Ren, and Pourang Irani. 2009. Detecting and leveraging finger orientation for interaction with direct-touch surfaces. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology* (UIST '09). ACM, New York, NY, USA, 23-32. http://doi.acm.org/10.1145/1622176.1622182
27. Daniel Wigdor, Hrvoje Benko, John Pella, Jarrod Lombardo, and Sarah Williams. 2011. Rock & rails: extending multi-touch interactions with shape gestures to enable precise spatial manipulations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*(CHI '11). ACM, NY, USA, 1581-1590. http://doi.acm.org/10.1145/1978942.1979173
28. Ying Yin, Tom Yu Ouyang, Kurt Partridge, and Shumin Zhai. 2013. Making touchscreen keyboards adaptive to keys, hand postures, and individuals: a hierarchical spatial backoff model approach. In*Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '13). ACM, New York, NY, USA, 2775-2784. http://doi.acm.org/10.1145/2470654.2481384
29. Robert Xiao, Gierad Laput, and Chris Harrison. 2014. Expanding the input expressivity of smartwatches with mechanical pan, twist, tilt and click. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '14). ACM, NY, USA, 193-196. http://doi.acm.org/10.1145/2556288.2557017
30. Jia Zhou, Jie Zhang, Bingjun Xie, Ning Liu, Ming Jiang, Huilin Wang, and Qiqing Gan. 2014. First-Time User Experience with Smart Phone New Gesture Control Features. In *Cross-Cultural Design*. Springer International Publishing, 2014, 262-271. http://dx.doi.org:8080/10.1007/978-3-319-07308-8_26