WATouCH: Enabling Direct Input on Non-touchscreen Using Smartwatch's Photoplethysmogram and IMU Sensor Fusion

Hui-Shyong Yeo*
Google Inc.
Mountain View, CA, USA
hsy@st-andrews.ac.uk

Wenxin Feng
Google Inc.
Mountain View, CA, USA
wenxinfeng@google.com

Michael Xuelin Huang
Google Inc.
Mountain View, CA, USA
mxhuang@google.com

ABSTRACT
Interacting with non-touchscreens such as TV or public displays can be difficult and inefficient. We propose WATouCH, a novel method that localizes a smartwatch on a display and allows direct input by turning the smartwatch into a tangible controller. This low-cost solution leverages sensor fusion of the built-in inertial measurement unit (IMU) and photoplethysmogram (PPG) sensor on a smartwatch that is used for heart rate monitoring. Specifically, WATouCH tracks the smartwatch movement using IMU data and corrects its location error caused by drift using the PPG responses to a dynamic visual pattern on the display. We conducted a user study on two tasks – a point and click and line tracing task – to evaluate the system usability and user performance. Evaluation results suggested that our sensor fusion mechanism effectively confined IMU-based localization error, achieved encouraging targeting and tracing precision, was well received by the participants, and thus opens up new opportunities for interaction.

Author Keywords
Smartwatch; Public display; Direct input; Tangible input.

CCS Concepts
•Human-centered computing → Human computer interaction (HCI); Pointing devices; Pointing;

INTRODUCTION
Electronic displays are increasingly available at home (e.g., TV), workplace (e.g., presentation display), and public places (e.g., advertising or information boards) [1, 28]. This will be even more so with the emerging vision of ubiquitous displays on every glass in future cities [11]. However, not all displays are equipped with sensing systems, such as touchscreens and gesture sensors, as these sensors can be costly. Therefore, to interact with passive displays, existing solutions require additional controllers, such as keyboards, mouses, or mobile devices [3]. However, it may fall short of providing direct input [8, 37], leading to non-intuitive and inefficient interaction experiences on large or public displays.

*This work was done while the first author was an intern at Google.

CHI '20, April 25–30, 2020, Honolulu, HI, USA.

ACM ISBN 978-1-4503-6708-0/20/04.
http://dx.doi.org/10.1145/3313831.3376198

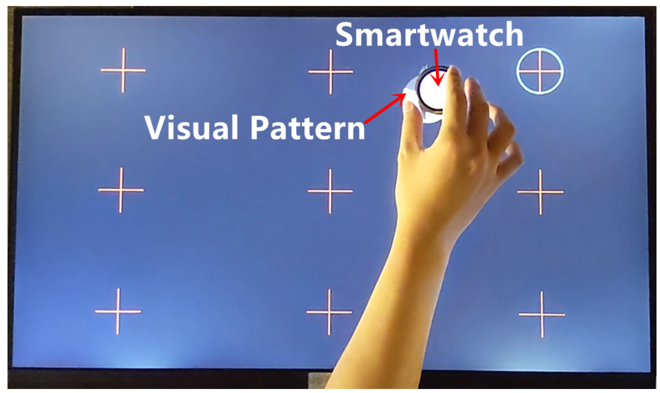

Figure 1. User grabbing and moving a smartwatch on top of a non-touchscreen. The location of the smartwatch is continuously tracked with high frame rate and minimal delay using IMU. If the estimated location is different from the ground truth, the visual pattern beneath the watch will expand or shrink. The changes will be tracked by PPG sensor, and the pattern shift accordingly to minimize error caused by drift.

Pertinent research efforts have been made to enable direct input with displays, such as hand-based [6], head-based [18], and gaze-based pointing [40]. However, due to the lack of haptic feedback, non-contact based pointing can be challenging and causes Midas Touch problem [31]. In contrast, interaction using a tangible controller offers touch sensation and affordance [12], although it requires specialized hardware.

Informed by the fact that smartwatches or fitness trackers are becoming pervasive and can be easily taken off from one's wrist by design [16, 24, 32], we propose an innovative input method, WATouCH, that exploits off-the-shelf smartwatch to achieve direct input on non-touchscreens, leveraging only built-in sensors. Our system allows users to grab and drag a smartwatch on a display as if using a puck or stylus, for point and click interaction (Figure 1). We can achieve fast and reliable tracking using the built-in sensors only and further leverage the touchscreen of the smartwatch for additional input. As such, our technique can be an intuitive, low-cost, and practical solution to non-touchscreen interaction.

Specifically, we track a smartwatch using its inertial measurement unit (IMU) signals, and we correct the location error by leveraging photoplethysmogram (PPG) responses to a visual pattern displayed on the non-touchscreen and underneath the smartwatch. This visual pattern follows the smartwatch, dynamically expands if the location tracked by IMU is in low confidence, shifts towards the center of the smartwatch, and finally shrinks to minimize the error. Taken together, PPG and IMU signals can continuously localize the smartwatch on non-touchscreens. Our user study demonstrated that our system was well-received and showed encouraging performance. Our technique is a software-based only approach, thus can be adopted by fitness trackers and smartwatches in-the-wild.

Our contribution is two-fold. We 1) repurpose a commercial smartwatch to allow direct input on non-touchscreens; 2) design an effective localization method based on PPG and IMU signals and achieve fast and reliable tracking under natural hand movements.

RELATED WORK

This study is related to smartwatch interaction using IMU and PPG sensor and object tracking on display.

Smartwatch Interaction using IMU or PPG Sensor

As smartwatch becomes pervasive, there is an increasing number of HCI studies on the smartwatch platform [33, 35]. One pertinent line of work related to ours is to leverage smartwatch as a three dimensional controller [50], using analog joystick [50] or by repurposing built-in IMU sensor [52]. Indeed, recent work also shown that the watchface can be detachable from the band [16, 24, 32] and repurposed for other use-cases. This is inline with our goal in this work.

Most common uses of the IMU on smartwatch are for features such as step counting, activity recognition and for detecting the raise-to-wake gesture that turns on the display. Research also explored pointing interaction [13], 3DoF control [15] and gesture recognition [19, 43]. In contrast, we repurpose IMU to track the movement of smartwatch relative to a display.

Besides the use of motion sensor, researcher has also investigated the opportunities of repurposing the PPG sensor on smartwatch that is originally designed for blood and heart rate monitoring. For example, researchers exploited the PPG signal for gesture recognition [4, 53]. In contrast, we repurpose it to track a visual pattern on unmodified display.

Inspired by prior explorations, our work leverages built-in sensors on a smartwatch and extends the smartwatch functionality. Different from existing studies, we consider beyond the interplay between user and smartwatch, and study the possibility of smartwatch cross-device interaction with non-touch displays.

Tracking Object(s) on Display

Object tracking methods for large-size displays can generally be categorized into non-vision and vision-based techniques. Common non-vision based techniques include capacitive sensing [27, 34, 36], magnetic field sensing [25, 30] and RFID [10, 38]. However, these techniques often require custom sensors, display and modified objects such as adding or embedding tags (conductive material, magnets or RFID).

Vision-based methods can be further clustered into method using external sensor(s) [9, 29, 47, 48] or built-in sensor on mobile devices [23, 49]. Since external camera(s) increases the cost and form factor of the system, we are more interested in approaches using only built-in sensors that are readily available in consumer devices.

Cross-device localization has also been studied using on-device camera or light sensors. THAW [23] used the phone camera to capture a predefined gradient pattern on the display and track its relative location while moving on top of the screen. Spatial tracking for hand-held devices has been studied by using single [42] or multiple light sensors [21, 22] to perceive the high-frequency pattern generated by a projector.

There were similar attempts for device localization that used light sensors to track the on-screen visual pattern [14, 39, 41, 51]. Stanton et al. [39] used a RGB light sensor to track a color pattern . Sugimoto et al. [41] devised a tangible cube with five one-channel light sensors to track a gray-scale gradient pattern. Kawamoto et al. [14] reduced the number of light sensors to four and used accelerometer data for estimating the landing location when the device was grabbed into midair and placed again on the surface. In contrast, our method continuously estimates the on-screen location of smartwatch from both the IMU and PPG signals. More importantly, prior studies relied on either multiple light sensors or a multi-channel sensor, while our method is able to achieve fast localization using the built-in IMU and PPG sensor without any modification or additional equipment.

SENSOR BACKGROUND

This section provides a brief explanation of each sensing modality (PPG and IMU) that we employ in our tracking technique and the issues arise when only one type of sensor is used to estimate location on a display.

Photoplethysmogram (PPG)

PPG sensor is included in many wearable devices such as smartwatches (e.g., Apple Watch, WearOS) and fitness trackers (e.g., Fitbit). It is generally used for heart rate and blood oxygenation monitoring, whereas some researchers have explored further use cases such as sleep tracking, diabetes detection, or gesture detection. A PPG sensor consists of one or multiple light sensitive photodiodes and light emitters. While different types of emitter are viable, common wearable devices in consumer market preferred green light emitter [20] because it is more robust to user movement and environmental lighting.

The principle of PPG sensing is based on light reflection and refraction. When light travels through biological tissues, it is more strongly absorbed by blood. Therefore, changes of blood flow can be captured by the PPG sensor as the changes of light intensity. In this work, we are the first to repurpose smartwatch's built-in PPG sensor for localization on display.

Inertial Measurement Unit (IMU)

An IMU typically contains accelerometer, gyroscope and magnetometer, which captures device acceleration, angular velocity and magnetic north, respectively. The data from these sensors are fused together to achieve robust three degree-of-freedom orientation tracking through sensor fusion approach.

Specifically, tracking position using IMU alone is notoriously difficult [5, 45] to achieve. The position must be derived

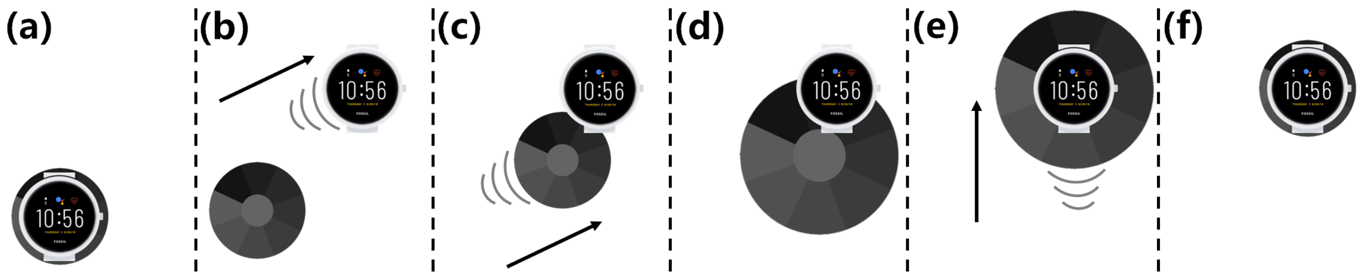

Figure 2. Simplified step-by-step tracking of our IMU + PPG sensor fusion technique, please refer to video figure for slow motion recording. From left to right: a) Initial state, visual pattern is right below the smartwatch. b) The smartwatch has been moved over a distance. c) The visual pattern follows the smartwatch using IMU based distance calculation but there is drift and the estimated location is different from actual smartwatch location. d) The visual pattern expands gradually until the PPG sensor of the smartwatch sees either region of the visual pattern. e) The visual pattern moves towards the smartwatch so that the middle region is right beneath the PPG sensor. f) The visual pattern shrinks until it is fully hidden beneath the smartwatch.

from acceleration through double integration. Acceleration is first integrated once to yield velocity, and then integrated again to yield position. This dead reckoning process through double integration means that even the tiniest measurement errors coming from noise or bias will result in a quadratically increasing error in the final position measurement, also known as *drift*. In practice, to alleviate the impact of drift, location tracking may be corrected by a secondary modality such as GPS [44] or under task-specific constraints such as zeroing velocity [5] errors during each stride, when the foot is detected as stationary [26].

DESIGN AND IMPLEMENTATION

In summary, we propose to track the on-screen location of a smartwatch using PPG and IMU sensor fusion. Essentially, we perform double integration on the device acceleration to measure the movement. To correct for drift, we show a visual pattern on the screen at the estimated location. We then recognize the relative location of the smartwatch to the visual pattern based on its PPG responses. This relative location further informs the adjustment of the visual pattern, whose on-screen location is used as the estimate of smartwatch location.

Location Initialization Using PPG Responses

One straightforward approach to localize the smartwatch is to flash the display with gray-code sequence [21, 22], allowing the PPG sensor to decode its location directly. We implemented a similar method that iteratively searches the smartwatch location, starting from the whole screen area and a narrower region after each iteration (Figure 3). However, given that common display has limited refresh rate (60Hz), this method can only estimate a coarse position every few frames, which is not fast enough for interactive purposes (it takes ~400ms to achieve a spatial precision of ~100 pixels after three iterations). It is also overly intrusive, as the whole display will be consumed, blocking all other UI elements or background images temporarily. Nonetheless, this is useful for the initial localization when the smartwatch touches the display for the first time, and then the system can switch to continuous tracking afterward.

Designing the Dynamics of Circle Visual Pattern

We designed a circular visual pattern consists nine regions (eight quadrants + one middle circle, see Figure 2). Each area

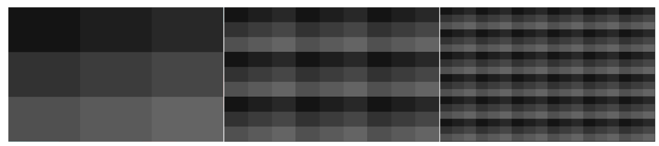

Figure 3. Flashing the whole screen with grid to search and localize the initial location of the smartwatch iteratively. From left to right, this figure depicts three iterations.

has a different shade of color or brightness that can be robustly recognized by the PPG sensor. The sensor we used (PAH8011 as in most of recent WearOS smartwatches) is not sensitive to color but only brightness. We opt for only low brightness so that the pattern appears dark and visually less intrusive. In idle time, only the center circle (most likely hidden beneath the smartwatch) is shown so as to reduce visual intrusiveness.

Localizing Inside Visual Pattern Using PPG Responses

As illustrated in Figure 2, there are three possible states and adjustment strategy for the visual pattern:

1. PPG sensor leaves the visual pattern – expand the size of the visual pattern.
2. PPG sensor stays inside any of the eight quadrants – shift the visual pattern towards that direction.
3. PPG sensor stays inside the middle circle – shrink the size of the visual pattern.

As shown in Figure 2, in case of (1), the pattern will expand its size by 2% every next frame until it is being seen by the PPG sensor again. (2) Once any of the eight outer quadrants is detected, the visual pattern will shift towards that direction. (3) When the PPG sensor sees the middle circle continuously, the pattern will shrink its size by 2.5% every next frame.

We extract the max PPG signal difference within a rolling window (10 PPG frames, 50ms) as feature to classify the watch location among the 11 classes, including 9 regions in the visual pattern, screen background, and unknown. The pattern also blinks every two frames so that it will not be confused with static background image. After taking into account the refresh rate and and pixel response time, we can achieve stable detection using only 3 display frames at 60Hz (common display refresh rate), which is 10 sensor frames at 200Hz (PPG sampling rate), or 50ms.

We chose a minimum size of visual pattern in our implementation for user study. The smaller the middle circle, the higher the attainable accuracy (not smaller than the PPG sensor size, which is ~2x3 mm). However, a small visual pattern takes a longer time to converge due to network delay and feedback loop. To balance the trade-off between speed and accuracy, we set the diameter for the middle circle to 150 pixels (~32 mm) for the user study session, which is smaller than typical smartwatch diameter and thus can be hidden beneath it.

Limitation of Using PPG Responses only

However, using a single PPG sensor to track movement suffers from inherent limitations. For one, it is insufficient to track the orientation of the device, which is why previous work used at least four to five photodiodes to achieve this.

Secondly, no matter using single or multiple photodiodes, this tracking method of using PPG alone will suffer from limited tracking speed, depending on the pattern size, as noticed by previous work as well [22, 39]. For example, if the movement within one frame exceeds the radius of the visual pattern, the photodiode will lose sight of the visual pattern, and hence lose tracking. To alleviate this problem, both the sensor sampling rate and the display refresh rate have to be rather high, and motion-to-photon latency has to be low. Increasing the pattern size can mitigate the issue to a certain extend, but a large pattern shown on screen can be overly intrusive to the user.

Localizing with PPG and IMU Fusion

This is why our approach does not rely solely on PPG sensor tracking. To address this limitation, we further utilize IMU for both position and orientation tracking. Yet, IMU has its own limitations (drift and noise), as we explained in the aforementioned sensor background section. These issues become worse when the hardware specification is limited, as we aim to repurpose built-in sensors of an off-the-shelf smartwatch.

Our technique uses assumptions of users performing impulse movement (akin to gait) to constrain the problems of drift, where integral drift is corrected each time the device is stationary using zero velocity update [5], and position error corrected by PPG tracking method. Hence, our proposed fusion technique aims to seamlessly combine these two types of sensor while alleviate the issues from each independent sensor.

We first remove the startup bias by a quick calibration, as the sensor measurement has an offset bias during startup, i.e., the difference between the real value and the output. Then the linear acceleration (100Hz, with gravity removed) is passed through a low-pass filter. We apply a threshold and zeroing the velocity errors [5] when the acceleration is lower than the threshold. This process will cause small movement not being detected by IMU-based localization method. Instead, this small movement will be detected by the PPG method. Finally the acceleration is integrated once to yield velocity, and integrated again to yield position. When the smartwatch decelerates and acceleration drops below the threshold, we stop integrating the acceleration and zeroing the velocity.

This method works well for a short period (about one second) while the drift is within acceptable range. For a long distance movement, the drift becomes high and the estimated position might differ greatly from the ground truth. In this case, we rely on the PPG method as explained above to correct and recover the true position. All in all, these two mechanisms combine seamlessly to achieve a fluid tracking experience, as shown in Figure 2 and video figure. In addition, the smartwatch's gyroscope can be used for rotation tracking, and the smartwatch's touchscreen can be used as a magic lens [2] for see-through and allow high precision touch within the tracked area.

However, for continuous tracking, the smartwatch needs to be dragged on the surface most of the time. WATouCH supports tracking when the device leaves the display for a limited amount of time (about 1 second). After that, it will start drifting and its location needs to be corrected by the visual pattern by placing it back on the surface (as shown in video figure).

USER STUDY

In order to evaluate the tracking accuracy of our approach based on sensor fusion of PPG and IMU, we conducted a user study aim to evaluate the feasibility of the approach and compare the result to the gold standard – a computer mouse, which we conducted a baseline study before our method. In this study, participants performed two tasks – a point and click and a line tracing task.

Participants

We recruited 12 participants (two female) from our local office, age ranging from 20 to 30 (Mean: 26.6). All participants were right handed and were seated during the study. The two parts study took about one hour and participants were compensated for a reward equivalent to $40.

Apparatus

We connected a Samsung Galaxy Tab S4 Android tablet to a generic, non-touchscreen 27 inches monitor (P2718EC, Foxconn, 2560x1440 resolution) using USB-C. The monitor was laid flat on an office desk. We used an Android WearOS smartwatch (Armani Exchange Connected) with the strap removed. The green light emitter was taped to prevent the automatic brightness adjustment. A wireless mouse (Logitech MX master) was used for the baseline study by connecting directly to the Android tablet using Bluetooth connection.

The tablet and smartwatch were connected to a WiFi router. The measured data from the smartwatch were stored in network packet and sent to the tablet at 100Hz (the highest rate of linear acceleration sensor on the smartwatch), using UDP protocol to reduce latency. Each packet is 26 bytes, and it contains PPG result, x-y velocity, yaw, click status and timestamp.

Procedure

We first introduced the technique to the participants and allowed them to practice for 3 minutes. During this time, we provide feedback to help them to become more accurate.

Study I: Point and Click

The primary study was a point and click task. There were 9 cross-hair targets in a 3x3 pattern distributed over the display area (Figure 1). The horizontal and vertical distance between each target were set to 950 pixels and 550 pixels, respectively, to account for different combinations of distance between each trial. The radius of each target was 100 pixels (approximately 2.1 cm on the 27 inches monitor, pixel pitch of 0.21 mm).

We set the four corner targets plus the middle target as the starting point and the remaining targets as the ending point. We used a special indicator for starting point to guide the participants to place at the starting location for the next trial. In total, there were 5 x 8 = 40 trials per session. In each trial, participants first placed the smartwatch on the starting point, touched anywhere on the smartwatch's touch display (the watch flashed green, and the tracker pattern started shrinking), moved it to the ending point (highlighted in cyan), and touched again to end (the watch flashed cyan). Each participant went through four sessions in this task, among which the first session was the baseline study using a wireless mouse and the next three sessions using a smartwatch.

Study II: Line Tracing

11 out of the 12 participants from the first study participated in the second study. The second study was a line tracing task. There were eight types of lines and shapes, including four directional lines, rectangle, triangle, circle and sine wave. Therefore, there were 5 rounds x 8 shapes = 40 trials for each session. In each trial, participants first placed the smartwatch on the starting point, touched anywhere on smartwatch's touch display, dragged the device along the shown path, and tapped again at the end point. Similar to the first study, each participant went through two sessions of this task, among which the first session was the baseline study using a wireless mouse and the second session using a smartwatch.

Results

Participants produced 1920 click trials and 880 drawn shapes.

Result I: Point and Click Spatial Accuracy and Speed

The spatial precision for click is shown in Figure 4. 95% confidence ellipses are drawn for the nine crosshair targets. Error vs. travel distance plot and time vs. travel distance plot are shown in Figure 5. The average error (Euclidean distance) from target center is 34.6 pixels with mouse and 54.7 pixels with smartwatch. Average minimum button diameter necessary to encompass 95% of clicks is 60.8 pixels (~12.8mm) with mouse and 95.2 pixels (~20.0mm) with smartwatch.

The mean completion time of a trial is 1115 ms with mouse and 3041 ms with smartwatch. It is worth noting that the movement time in the wireless mouse condition is used as a reference rather than a direct comparison. This is because mouse uses relative pointing with cursor acceleration and conceivably leads to a shorter time than that of WATouCH, which requires moving a tangible object physically on a large surface. As shown in Figure 5 (bottom), the time to click on a target increases with the travel distance for both input devices. The other factor that contributes to the longer completion time with smartwatch is the waiting time for the visual pattern to converge. For longer distances, the drift becomes considerably larger, therefore the visual pattern has to expand, shift, and shrink, and thus resulted in a longer completion time.

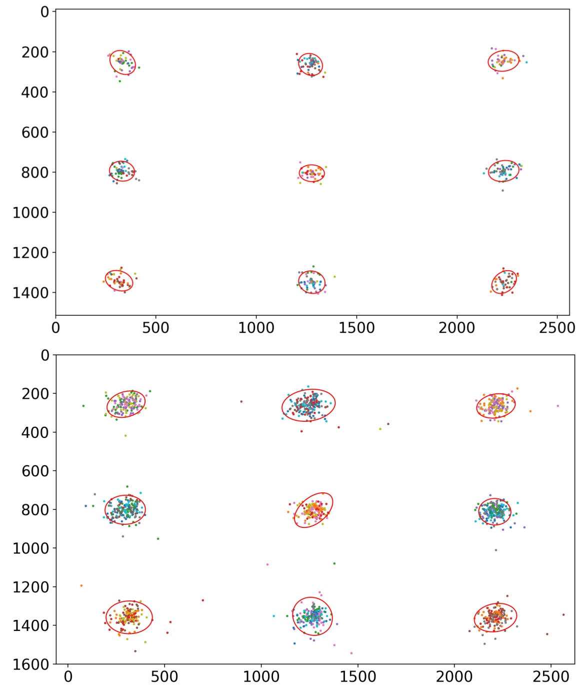

Figure 4. Scatter plot of user click distributions for the two input devices we tested. (Top) Baseline result using wireless mouse. (Bottom) Result using smartwatch as tangible controller. 95% confidence ellipses are shown in red. Axis units in pixel.

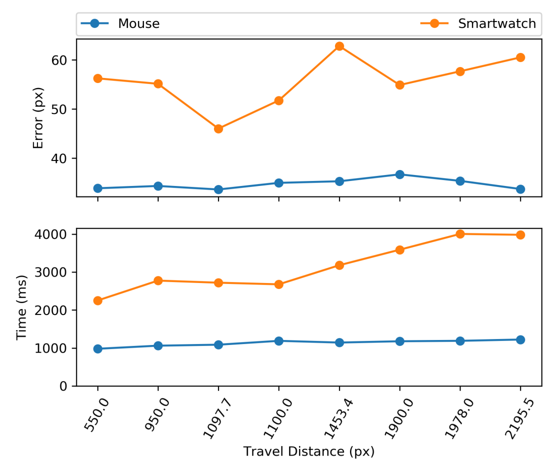

Figure 5. Top: Error (Euclidean distance, px) vs. travel distance (px), Bottom: Completion time (ms) vs. travel distance (px).

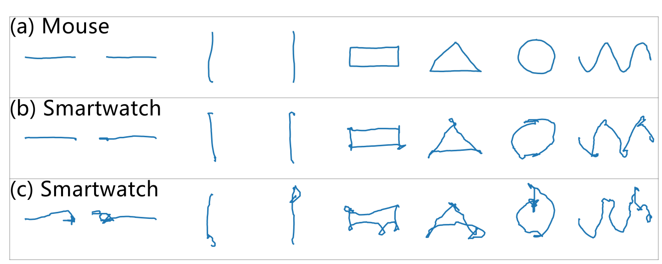

Figure 6. Example lines drawn by participants in the line tracing task. (a) Baseline result using a wireless mouse. (b) Well drawn examples using a smartwatch. (c) Less well drawn examples using a smartwatch.

Result II: Line Tracing Spatial Accuracy and Speed

There are four lines and four shapes in the line tracing experiment. Following the previous practice [7], we calculated the absolute Euclidean distance from the closest point on the ideal path as metric to measure the distance error. On average, participants deviated from the ideal path by 57.3 pixels (SD = 31.8) when using the smartwatch, compared to 25.1 pixels (SD = 16.5) when using a wireless mouse. Example lines drawn by participants can be seen in Figure 6. The lines and shapes drawn by the smartwatch are recognizable.

In our current implementation, drawing a long path can be challenging due to the accelerometer drift. When drawing the shapes, participants made pauses at turning points to allow the visual pattern to shift and converge.

Participants Feedback

Participants rated their experience using the NASA-TLX on a 5-point Likert scale, as depicted in Figure 7 (lower score is better) and a further four subjective questions (Figure 8). It can be seen that the load across all parameters is low, and the technique is generally well-received.

Since our study was conducted with the participants seated, some participants questioned the easiness if they have to use the smartwatch on a vertical display (TV or public display). They concerned it could be difficult to grab and tap the smartwatch's touchscreen at the same time in a vertical display setting. Participants also worried that they might accidentally drop the device. One participant was reluctant to use his own smartwatch to slide on a public display, as he said: "I don't want to scratch my smartwatch and the PPG lens". Due to the form-factor of the smartwatch, it requires at least two to three fingers to grip. Participants compared this to using a touchscreen where only one finger is needed.

Regarding intuitiveness, the majority of the participants found the technique intuitive, given that the tracking speed could be improved in a future version. With respect to social acceptability, the majority of the participants were willing to use this technique in public, and they did not feel awkward, assuming that future smartwatch can be easily detachable. For intrusiveness, e.g., "does the flickering pattern cause eyestrain, headache or dizziness?" all participants did not find the flickering intrusive. Only some participants reported that it was intrusive when the visual pattern became too large and blocking a large area of the screen content.

POTENTIAL USE CASES

WATouCH enables direct input on non-touchscreens, and thus opens up a number of interaction possibilities. We envision that it can be used to perform pointing, dragging, and selecting in, but not limited to, the following use cases:

Interacting with public displays - Although non-touch public displays have been pervasively used for advertising [28] in a variety of places, including airport, shopping mall, and exhibition, the way that they convey visual messages passively to users confine the degree of user engagement. There is a pressing need to make such displays interactive and engaging. Interactive ads and game demo can be two obvious applications. WATouCH allows for intuitive walk-up-and-use interaction in an ad hoc fashion. Users can interact with non-touchscreens by direct input using smartwatch as if using a stylus or a puck. This natural interaction design can encourage user engagement, enhance user experience, and thus boost the advertisement effect. For example, users can play a quick game and earn rewards on digital signage at a bus stop while waiting for a bus. Similarly, users can browse the list of shops in a mall, get a coupon, and enjoy the shopping experience.

Gesture typing on smart TVs - Text input on TV displays using a remote or game controller is often challenging, inefficient, and even frustrating. WATouCH allows users to drag over a virtual keyboard on the TV display, and the estimated trajectory of hand movement can be parsed by the system and used for gesture typing (shape writing) [17]. This can provide a marked improvement on typing throughput on the TV dis-

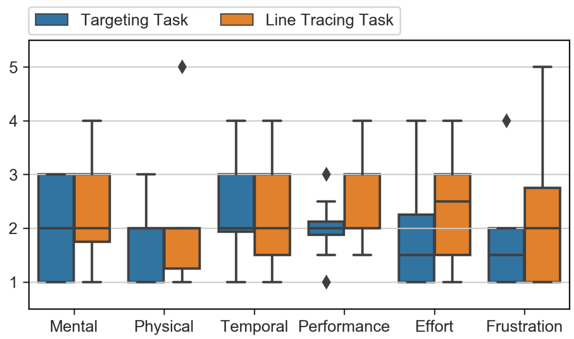

Figure 7. NASA Task Load Index (TLX) subjective ratings for both tasks. A lower score is better.

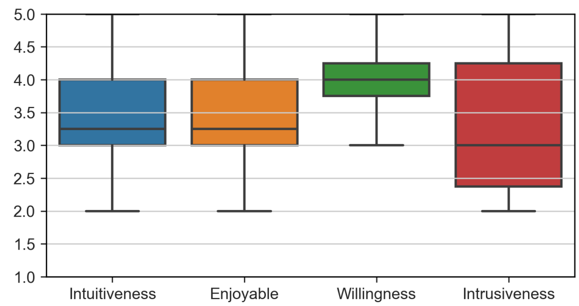

Figure 8. Extra subjective questions including intuitiveness, enjoyable, willingness and intrusiveness. A higher score is better.

Figure 9. Applications. (Left) Selecting and highlighting text in a presentation slide. (Right) Inputting text using on-screen gesture keyboard.

plays (Figure 9 (right)). By doing so, it is easier for users to type complete keywords rather than initial alone, or multiple keywords to achieve a precise search.

Operating on tangible tabletops - A tangible tabletop can be useful in a work space or entertaining space, such as being deployed on collaboration desks, coffee tables, or exhibition tables in museum, but it often has high entry cost to setup. WATouCH repurposes any display and off-the-shelf smartwatches as a low-cost tangible tabletop. Because the smartwatch also contains gyroscope, it can be used as a rotatable dial for painting and music application. Furthermore, the smartwatch screen can be leveraged as a magic lens [2] for see-through or display additional information.

Presenting in meetings - During a meeting or a discussion, one may want to highlight, draw notes, or change slides directly on the presentation display (Figure 9 (left)). In such cases, WATouCH allows the users to directly "touch" the display with their smartwatch, so that they do not have to go back and forth to the presentation machine.

DISCUSSION AND LIMITATIONS

This is the first study to our knowledge that achieves direct input on non-touchscreens by repurposing a smartwatch. WATouCH uses a new localization method based on PPG and IMU sensor fusion. It is a promising technique as it relies on existing sensors available on a smartwatch, and it provides the sense of direct input and manipulation for users.

Experiment results showed that WATouCH is well received by the participants. Although comparing with mouse, WATouCH falls short of control precision and speed. This can conceivably be improved by an additional filtering mechanism to refine the trajectory or high-level gesture classifier to recognize a user's intention. Encouragingly, WATouCH can achieve sufficient pointing precision for visual targets in a touch-based graphical user interface.

For dragging, WATouCH works well for short and medium distances, which can be the most common cases for natural hand movements in the comfort zone. For the longer distance, localization error due to IMU drift requires the visual pattern to expand and correct for IMU estimation based on PPG sensing, and it makes the input slower than dedicated input devices. This issue increases the difficulty of drawing a shape, where pausing is required.

Identifying Hardware Challenges

In the course of our study, we identified some challenges.

This first challenge is the sampling rates of sensors and displays. Although the PPG sensor we explored (PAH8011) has a sampling rate of 200Hz, which is fast enough for continuous tracking, the effective tracking rate is inherently limited by the underlying display characteristics, in particular, the refresh rate and pixel response time. Most common displays (TVs and monitors) only have a 60Hz refresh rate, and a non-negligible pixel response time, i.e., latency for a pixel to change from one color to another. Taking into account the flickering (for distinguishing between static background and our visual pattern) and pixel response time, we can achieve reliable detection at three display frames (50ms, 10 PPG frames at 200Hz sampling rate). Other displays, such as iPad or gaming monitors that have a higher refresh rate and lower pixel response time, can increase tracking speed, and hence improves robustness and reduces intrusiveness (flickering will be less obvious).

The second challenge is motion-to-photon latency [54], mostly caused by WiFi network latency (with occasional packets drop when using UDP) and display input lag. We see Bluetooth low energy as a suitable option to minimize such latency.

The third one comes from the ambiguity between the regions in the visual pattern. Since the PPG sensor has a size of ~2x3mm, it can be trapped on the edge between two regions of the visual pattern, where the sensor sees two different regions, and hence the estimated result fluctuates between states. To avoid this issue, we shift the visual pattern in a small step in a randomized direction, much like Pixel shifting [46] technique used in smartphone's always-on display to avoid burn-in. The feature was turned off in the user study because of the quick shifting is not imperceptible to the user.

Lastly, we noticed that the PPG sensor we used (PAH8011, used in most of the recently launched WearOS smartwatches) has a built-in feature that automatically changes the emitter intensity based on the reflected light. This affects our measurement, therefore we blocked the emitter using adhesive tape for fast prototyping. In practice, the green light or automatic intensity adjustment feature can be disabled in the low-level Android system.

Future Work

In future work, we aim to improve the tracking speed by applying the Kalman filter and compare with touchscreen system. Battery consumption is an important factor that requires attention and effort to make WATouCH practical at scale. Tracking multiple devices simultaneously will be useful, especially for tangible tabletops. Data transfer is another avenue for future research. Finally, we would like to explore more use cases and applications such as tangible tabletop and cross-device interaction.

CONCLUSION

We presented a novel method for repurposing and combining the built-in PPG and IMU sensors in off-the-shelf smartwatch for accurate location tracking on unmodified, non-touchscreen displays. The results from our user study showed that participants can use the system with minimal training. WATouCH produces a viable precision of target selection in touch-based graphical user interface, with a comparable precision to that of a wireless mouse. As the first work of repurposing smartwatch for direct input on non-touchscreens, it leads to useful insights of technical challenges and achieves promising results to encourage future studies. WATouCH also enables new interaction possibilities when a wireless mouse or touch input is not readily available, such as interacting with public displays, ad hoc presentation on a TV, or tangible interaction on a tabletop. With a further improvement of tracking speed, we envision that this technique can be well-received at scale and be included in commercial smartwatches and fitness trackers without hardware modification.

REFERENCES

[1] Rafael Ballagas, Michael Rohs, and Jennifer G. Sheridan. 2005. Sweep and Point and Shoot: Phonecam-Based Interactions for Large Public Displays. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems (CHI EA '05)*. Association for Computing Machinery, New York, NY, USA, 1200–1203. DOI: http://dx.doi.org/10.1145/1056808.1056876

[2] Eric A. Bier, Maureen C. Stone, Ken Pier, William Buxton, and Tony D. DeRose. 1993. Toolglass and Magic Lenses: The See-through Interface. In *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '93)*. Association for Computing Machinery, New York, NY, USA, 73–80. DOI: http://dx.doi.org/10.1145/166117.166126

[3] Sebastian Boring, Marko Jurmu, and Andreas Butz. 2009. Scroll, Tilt or Move It: Using Mobile Phones to Continuously Control Pointers on Large Public Displays. In *Proceedings of the 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group: Design: Open 24/7 (OZCHI '09)*. Association for Computing Machinery, New York, NY, USA, 161–168. DOI: http://dx.doi.org/10.1145/1738826.1738853

[4] Thisum Buddhika, Haimo Zhang, Samantha W. T. Chan, Vipula Dissanayake, Suranga Nanayakkara, and Roger Zimmermann. 2019. fSense: Unlocking the Dimension of Force for Gestural Interactions Using Smartwatch PPG Sensor. In *Proceedings of the 10th Augmented Human International Conference 2019 (AH2019)*. ACM, New York, NY, USA, Article 11, 5 pages. DOI: http://dx.doi.org/10.1145/3311823.3311839

[5] E. Foxlin. 2005. Pedestrian tracking with shoe-mounted inertial sensors. *IEEE Computer Graphics and Applications* 25, 6 (Nov 2005), 38–46. DOI: http://dx.doi.org/10.1109/MCG.2005.140

[6] Jefferson Y. Han. 2005. Low-cost Multi-touch Sensing Through Frustrated Total Internal Reflection. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology (UIST '05)*. ACM, New York, NY, USA, 115–118. DOI: http://dx.doi.org/10.1145/1095034.1095054

[7] Chris Harrison, Hrvoje Benko, and Andrew D. Wilson. 2011. OmniTouch: Wearable Multitouch Interaction Everywhere. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology (UIST '11)*. ACM, New York, NY, USA, 441–450. DOI: http://dx.doi.org/10.1145/2047196.2047255

[8] Ken Hinckley and Daniel Wigdor. 2002. Input technologies and techniques. *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications* (2002), 151–168.

[9] Steve Hodges, Shahram Izadi, Alex Butler, Alban Rrustemi, and Bill Buxton. 2007. ThinSight: Versatile Multi-touch Sensing for Thin Form-factor Displays. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology (UIST '07)*. ACM, New York, NY, USA, 259–268. DOI: http://dx.doi.org/10.1145/1294211.1294258

[10] Meng-Ju Hsieh, Rong-Hao Liang, Da-Yuan Huang, Jheng-You Ke, and Bing-Yu Chen. 2018. RFIBricks: Interactive Building Blocks Based on RFID. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 189, 10 pages. DOI: http://dx.doi.org/10.1145/3173574.3173763

[11] Microsoft in Business. 2015. Microsoft: Productivity Future Vision. (2015). Retrieved September 14, 2019 from https://www.youtube.com/watch?v=w-tFdreZB94.

[12] Hiroshi Ishii and Brygg Ullmer. 1997. Tangible Bits: Towards Seamless Interfaces Between People, Bits and Atoms. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI '97)*. ACM, New York, NY, USA, 234–241. DOI: http://dx.doi.org/10.1145/258549.258715

[13] Keiko Katsuragawa, Krzysztof Pietroszek, James R. Wallace, and Edward Lank. 2016. Watchpoint: Freehand Pointing with a Smartwatch in a Ubiquitous Display Environment. In *Proceedings of the International Working Conference on Advanced Visual Interfaces (AVI '16)*. ACM, New York, NY, USA, 128–135. DOI: http://dx.doi.org/10.1145/2909132.2909263

[14] U Kawamoto, Takeshi Kurata, N Sakata, Takashi Okuma, and Hideaki Kuzuoka. 2008. Position/orientation-aware physical tags using photo sensors and accelerometers for a tangible tabletop interface. In *Proc. IEEE International Conference on Distributed Human-Machine Systems (DHMS)*. 397–404.

[15] Daniel Kharlamov, Brandon Woodard, Liudmila Tahai, and Krzysztof Pietroszek. 2016. TickTockRay: Smartwatch-based 3D Pointing for Smartphone-based Virtual Reality. In *Proceedings of the 22Nd ACM Conference on Virtual Reality Software and Technology (VRST '16)*. ACM, New York, NY, USA, 365–366. DOI: http://dx.doi.org/10.1145/2993369.2996311

[16] Rushil Khurana, Mayank Goel, and Kent Lyons. 2019. Detachable Smartwatch: More Than A Wearable. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3, 2, Article 50 (June 2019), 14 pages. DOI: http://dx.doi.org/10.1145/3328921

[17] Per-Ola Kristensson and Shumin Zhai. 2004. SHARK2: A Large Vocabulary Shorthand Writing System for Pen-based Computers. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology (UIST '04)*. ACM, New York, NY, USA, 43–52. DOI:http://dx.doi.org/10.1145/1029632.1029640

[18] Mikko Kytö, Barrett Ens, Thammathip Piumsomboon, Gun A. Lee, and Mark Billinghurst. 2018. Pinpointing: Precise Head- and Eye-Based Target Selection for Augmented Reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 81, 14 pages. DOI:http://dx.doi.org/10.1145/3173574.3173655

[19] Gierad Laput, Robert Xiao, and Chris Harrison. 2016. ViBand: High-Fidelity Bio-Acoustic Sensing Using Commodity Smartwatch Accelerometers. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, New York, NY, USA, 321–333. DOI: http://dx.doi.org/10.1145/2984511.2984582

[20] J. Lee, K. Matsumura, K. Yamakoshi, P. Rolfe, S. Tanaka, and T. Yamakoshi. 2013. Comparison between red, green and blue light reflection photoplethysmography for heart rate monitoring during motion. In *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. 1724–1727. DOI: http://dx.doi.org/10.1109/EMBC.2013.6609852

[21] Johnny C. Lee, Paul H. Dietz, Dan Maynes-Aminzade, Ramesh Raskar, and Scott E. Hudson. 2004. Automatic Projector Calibration with Embedded Light Sensors. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology (UIST '04)*. ACM, New York, NY, USA, 123–126. DOI: http://dx.doi.org/10.1145/1029632.1029653

[22] Johnny C. Lee, Scott E. Hudson, Jay W. Summet, and Paul H. Dietz. 2005. Moveable Interactive Projected Displays Using Projector Based Tracking. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology (UIST '05)*. ACM, New York, NY, USA, 63–72. DOI: http://dx.doi.org/10.1145/1095034.1095045

[23] Sang-won Leigh, Philipp Schoessler, Felix Heibeck, Pattie Maes, and Hiroshi Ishii. 2015. THAW: Tangible Interaction with See-Through Augmentation for Smartphones on Computer Screens. In *Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '15)*. ACM, New York, NY, USA, 89–96. DOI: http://dx.doi.org/10.1145/2677199.2680584

[24] LEMFO. 2019. LEMFO LEM11 4G Smart Watch Android 7.1 3GB 32GB Video Call With 1200mah Power Bank Wireless Bluetooth Speaker Strap Replaceable. (2019). Retrieved December 26, 2019 from https://www.aliexpress.com/item/4000141268956.html.

[25] Rong-Hao Liang, Kai-Yin Cheng, Liwei Chan, Chuan-Xhyuan Peng, Mike Y. Chen, Rung-Huei Liang, De-Nian Yang, and Bing-Yu Chen. 2013. GaussBits: Magnetic Tangible Bits for Portable and Occlusion-free Near-surface Interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 1391–1400. DOI:http://dx.doi.org/10.1145/2470654.2466185

[26] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan. 2011. Estimation of IMU and MARG orientation using a gradient descent algorithm. In *2011 IEEE International Conference on Rehabilitation Robotics*. 1–7. DOI: http://dx.doi.org/10.1109/ICORR.2011.5975346

[27] Rafael Morales González, Caroline Appert, Gilles Bailly, and Emmanuel Pietriga. 2016. TouchTokens: Guiding Touch Patterns with Passive Tokens. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 4189–4202. DOI: http://dx.doi.org/10.1145/2858036.2858041

[28] J. Müller, A. Schmidt, and F. Alt. 2012. Advertising on Public Display Networks. *Computer* 45, 05 (may 2012), 50–56. DOI:http://dx.doi.org/10.1109/MC.2012.150

[29] Alex Olwal. 2006. LightSense: Enabling Spatially Aware Handheld Interaction Devices. In *Proceedings of the 5th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '06)*. IEEE Computer Society, Washington, DC, USA, 119–122. DOI:http://dx.doi.org/10.1109/ISMAR.2006.297802

[30] James Patten, Hiroshi Ishii, Jim Hines, and Gian Pangaro. 2001. Sensetable: A Wireless Object Tracking Platform for Tangible User Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '01)*. ACM, New York, NY, USA, 253–260. DOI: http://dx.doi.org/10.1145/365024.365112

[31] Abdul Moiz Penkar, Christof Lutteroth, and Gerald Weber. 2012. Designing for the Eye: Design Parameters for Dwell in Gaze Interaction. In *Proceedings of the 24th Australian Computer-Human Interaction Conference (OzCHI '12)*. ACM, New York, NY, USA, 479–488. DOI:http://dx.doi.org/10.1145/2414536.2414609

[32] POISE. 2018. POISE Detachable Smartwatch. (2018). Retrieved September 14, 2019 from https://www.indiegogo.com/projects/poise-detachable-smartwatch#/.

[33] Reza Rawassizadeh, Blaine A. Price, and Marian Petre. 2014. Wearables: Has the Age of Smartwatches Finally Arrived? *Commun. ACM* 58, 1 (Dec. 2014), 45–47. DOI: http://dx.doi.org/10.1145/2629633

[34] Jun Rekimoto. 2002. SmartSkin: An Infrastructure for Freehand Manipulation on Interactive Surfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '02)*. ACM, New York, NY, USA, 113–120. DOI: http://dx.doi.org/10.1145/503376.503397

[35] Steven Schirra and Frank R. Bentley. 2015. "It's Kind of Like an Extra Screen for My Phone": Understanding Everyday Uses of Consumer Smart Watches. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '15)*. ACM, New York, NY, USA, 2151–2156. DOI: http://dx.doi.org/10.1145/2702613.2732931

[36] Martin Schmitz, Jürgen Steimle, Jochen Huber, Niloofar Dezfuli, and Max Mühlhäuser. 2017. Flexibles: Deformation-Aware 3D-Printed Tangibles for Capacitive Touchscreens. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 1001–1014. DOI:http://dx.doi.org/10.1145/3025453.3025663

[37] Shneiderman. 1983. Direct Manipulation: A Step Beyond Programming Languages. *Computer* 16, 8 (Aug 1983), 57–69. DOI: http://dx.doi.org/10.1109/MC.1983.1654471

[38] Andrew Spielberg, Alanson Sample, Scott E. Hudson, Jennifer Mankoff, and James McCann. 2016. RapID: A Framework for Fabricating Low-Latency Interactive Objects with RFID Tags. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 5897–5908. DOI: http://dx.doi.org/10.1145/2858036.2858243

[39] R. Stanton, E. Pontelli, Z. Toups, and M. Manshad. 2018. Exploring a Novel Inexpensive Tangible Interface for Non-visual Math and Science. In *Computers Helping People with Special Needs*, Klaus Miesenberger and Georgios Kouroupetroglou (Eds.). Springer International Publishing, Cham, 619–627.

[40] Yusuke Sugano, Xucong Zhang, and Andreas Bulling. 2016. AggreGaze: Collective Estimation of Audience Attention on Public Displays. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, New York, NY, USA, 821–831. DOI: http://dx.doi.org/10.1145/2984511.2984536

[41] M. Sugimoto, K. Kodama, A. Nakamura, M. Kojima, and M. Inami. 2007. A Display-Based Tracking System: Display-Based Computing for Measurement Systems. In *17th International Conference on Artificial Reality and Telexistence (ICAT 2007)*. 31–38. DOI: http://dx.doi.org/10.1109/ICAT.2007.50

[42] Jay Summet and Rahul Sukthankar. 2005. Tracking Locations of Moving Hand-Held Displays Using Projected Light. In *Pervasive Computing*, Hans W. Gellersen, Roy Want, and Albrecht Schmidt (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 37–46.

[43] Hongyi Wen, Julian Ramos Rojas, and Anind K. Dey. 2016. Serendipity: Finger Gesture Recognition Using an Off-the-Shelf Smartwatch. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 3847–3851. DOI: http://dx.doi.org/10.1145/2858036.2858466

[44] Wikipedia. 2019a. GPS/INS. (2019). Retrieved December 26, 2019 from https://en.wikipedia.org/wiki/GPS/INS.

[45] Wikipedia. 2019b. Inertial navigation system. (2019). Retrieved December 26, 2019 from https://en.wikipedia.org/wiki/Inertial_navigation_system.

[46] Wikipedia. 2019c. Pixel shifting. (2019). Retrieved September 14, 2019 from https://en.wikipedia.org/wiki/Pixel_shifting.

[47] Andrew D. Wilson. 2005. PlayAnywhere: A Compact Interactive Tabletop Projection-vision System. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology (UIST '05)*. ACM, New York, NY, USA, 83–92. DOI: http://dx.doi.org/10.1145/1095034.1095047

[48] Andrew D. Wilson and Raman Sarin. 2007. BlueTable: Connecting Wireless Mobile Devices on Interactive Surfaces Using Vision-based Handshaking. In *Proceedings of Graphics Interface 2007 (GI '07)*. ACM, New York, NY, USA, 119–125. DOI: http://dx.doi.org/10.1145/1268517.1268539

[49] Robert Xiao, Scott Hudson, and Chris Harrison. 2016. CapCam: Enabling Rapid, Ad-Hoc, Position-Tracked Interactions Between Devices. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces (ISS '16)*. ACM, New York, NY, USA, 169–178. DOI: http://dx.doi.org/10.1145/2992154.2992182

[50] 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, New York, NY, USA, 193–196. DOI: http://dx.doi.org/10.1145/2556288.2557017

[51] Akio YAMAMOTO and Taku NAKAMURA. 2018. Performance of displayed-marker-based position tracking. *Journal of Advanced Mechanical Design, Systems, and Manufacturing* 12, 7 (2018), JAMDSM0125–JAMDSM0125. DOI: http://dx.doi.org/10.1299/jamdsm.2018jamdsm0125

[52] Hui-Shyong Yeo, Juyoung Lee, Andrea Bianchi, and Aaron Quigley. 2016. WatchMI: Pressure Touch, Twist and Pan Gesture Input on Unmodified Smartwatches. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '16)*. ACM, New York, NY, USA, 394–399. DOI: http://dx.doi.org/10.1145/2935334.2935375

[53] Yu Zhang, Tao Gu, Chu Luo, Vassilis Kostakos, and Aruna Seneviratne. 2018. FinDroidHR: Smartwatch Gesture Input with Optical Heartrate Monitor. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 1, Article 56 (March 2018), 42 pages. DOI: http://dx.doi.org/10.1145/3191788

[54] J. Zhao, R. S. Allison, M. Vinnikov, and S. Jennings. 2017. Estimating the motion-to-photon latency in head mounted displays. In *2017 IEEE Virtual Reality (VR)*. 313–314. DOI: http://dx.doi.org/10.1109/VR.2017.7892302