Pointing at a Distance with Everyday Smart Devices

Shaishav Siddhpuria[1], **Sylvain Malacria**[2], **Mathieu Nancel**[2] **& Edward Lank**[1,2,3]
[1]Cheriton School of Computer Science, University of Waterloo, Canada
[2]Inria, France
[3]Université de Lille, France
spsiddhp@uwaterloo.ca, sylvain.malacria@inria.fr, mathieu.nancel@inria.fr, lank@uwaterloo.ca

ABSTRACT
Large displays are becoming commonplace at work, at home, or in public areas. However, interaction at a distance – anything greater than arms-length – remains cumbersome, restricts simultaneous use, and requires specific hardware augmentations of the display: touch layers, cameras, or dedicated input devices. Yet a rapidly increasing number of people carry smartphones and smartwatches, devices with rich input capabilities that can easily be used as input devices to control interactive systems. We contribute (1) the results of a survey on possession and use of smart devices, and (2) the results of a controlled experiment comparing seven distal pointing techniques on phone or watch, one- and two-handed, and using different input channels and mappings. Our results favor using a smartphone as a trackpad, but also explore performance tradeoffs that can inform the choice and design of distal pointing techniques for different contexts of use.

ACM Classification Keywords
H.5.2 User Interfaces

Author Keywords
smartphone; smartwatch; pointing; distant display

INTRODUCTION

Large displays have become increasingly common, whether at home, at work, or in public settings. Surprisingly, interaction with each of these display usually requires us to use a dedicated input device for that specific display. For example, to interact with large displays in the home a remote control, a specific pointing tool (e.g. Wiimote), or a camera-based tracking system (e.g. Kinect) is typically used. In public spaces, once the user moves beyond arms-length, camera-based tracking systems are, again, the most frequent solutions to interaction, and these are limited in range to the location view-able by the camera and often require exaggerated movements, giving rise to issues of embarrassment [1].

In public spaces, one interactive screen that the vast majority of individuals have access to is a touchscreen on a personal device. Three quarters of americans now own a smartphone, a trend replicated in other developed countries [29]. Alongside smartphones, smartwatch sales trends indicate that, in 2018, 2% of the world's population will *purchase* a smartwatch[1], and many people already possess smartwatches, fitness bands, or other wearable devices. Given these trends, we believe personal handheld devices (typically, smartphones or smartwatches) offer a platform of opportunity to support interaction with distal displays [7, 32].

One challenge in this space is to discern how to best support interaction with a distal display using a smartwatch or smartphone. A significant body of work in this space has focused on point-and-click interactions [37]. However, even within this space of point-and-click support, there are many different smartphone and smartwatch techniques to support interaction including: surface-based spatially aware solutions [39], solutions that leverage the touchscreen as a touchpad [2], and solutions that leverage the onboard inertial measurement unit (IMU – i.e. accelerometer, gyroscope, magnetometer) to measure device rotation around the user [18]. While these techniques have often been shown to be effective against competing systems that use the same form of interaction (e.g. comparing smartphone as touchpad to computer-based touchpad [2]; comparing IMU solutions to Vicon-tracked freehand pointing [18]), we find only limited work comparing these solutions one-to-another to determine the costs and trade-offs of each technique. With this in mind, it remains unclear how well users control can a cursor on a distant screen with a conventional smartphone or smartwatch.

In this paper, we investigate the use of smartphones and smartwatches as pointing devices for distant screens. Our goal is to study different families of input techniques using conventional handheld devices for pointing on a distant displays. More specifically, we seek to answer questions related to the performance and subjective preferences of users, including: Beyond their almost universal appeal, do rotation gestures performed in mid-air work better than simple touch input? Is the touch surface of a smartwatch large enough for performing efficient pointing? Overall, our goal is not to validate or invalidate any single form of interaction. Many other factors – device availability, whether the user is holding something in one hand while interaction with the other, physical space available to move arms versus use a touchscreen – play a role in device usage. However, at the very least, comparative statistics on speed and error rate for cursor control is one factor that can be considered in deciding on specific interaction techniques.

[1]https://www.statista.com/statistics/538237/global-smartwatch-unit-sales/

We describe an experiment that compares 7 interaction techniques that rely either on a smartphone or a smartwatch to perform 2D pointing. Techniques evaluated include variations on touch-based and IMU-based pointing techniques with both smartphone and smartwatch, using both absolute and relative pointing variants (Table 1). We find that touch-based techniques offer better performance than rotation gestures, both in term of speed and accuracy, and that using a smartwatch screen like a trackpad performs extremely well considering the small input range. We discuss the trade-offs in design exposed by this research to the design of distal pointing techniques that leverage personal devices.

RELATED WORK

Distant Pointing Techniques

Alongside touch, interaction with large public displays has included a variety of distal interaction techniques [36]. Some of these techniques rely simply on embodiment and user position [35]. Others incorporate specialized devices [17]. Still others leverage everyday objects, typically augmented with novel tracking hardware [4, 31].

Perhaps the most obvious form of interaction for distant pointing is freehand pointing where a user simply points at the display [37]. While this paradigm of interaction has been popularized by commodity hardware like the Kinect [28], the deployment of systems that support unaugmented freehand pointing remains challenging. Computer vision systems that track users in the environment, identify explicit pointing gestures from everyday movements, and then responsively interpret this movement still require the deployment of hardware, careful calibration, and advanced computational intelligence to achieve high reliability [21]. In-lab tests of these systems typically leverage prohibitively expensive hardware and augmented markings [37, 15] to simplify the difficulty of the tracking problem. Given that it is difficult to support precise tracking in lab environments with augmentation, in uncontrolled environments the tracking problem becomes even more unreliable, necessitating feedback, for example the use of shadows [34] and larger scale gestures that can contribute to user embarrassment [1].

To avoid fully instrumenting the world with cameras, a second class of solutions to the distal pointing problem leverages everyday objects to support interaction. In some cases, these objects are readily available but not something that the majority of people carry with them ubiquitously – for example laser pointers [9, 38] or Wiimotes [8, 19]. In other cases, these objects are everyday objects found in the environment that are then selectively augmented and co-opted as input [4, 31].

The third class of distal pointing techniques leverages ubiquitous personal devices – specifically smartphones and smartwatches – as platforms of convenience to interact with external displays [5, 18, 14, 27, 25, 30]. Early work in this domain leveraged the touchscreen for input, essentially using the device as a drawing pad [23], a mouse plus sketch surface [39], or a touchpad [2, 16]. More recently, the ubiquity of accelerometers and gyroscopes in these devices has made it possible to mimic freehand pointing techniques [18, 26, 33]. Recent research has shown that freehand pointing with a phone performs on par with the Nintendo Wiimote [26], and that freehand pointing with a smartwatch performs at near parity with Vicon-based freehand distal pointing [18]. As researchers, we believe that distal pointing with ubiquitous personal devices holds the most near-term promise for everyday public display interaction: Essentially, rather than augmenting the entire world or requiring potential users to identify and obtain relevant objects, if the input device is either a smartphone or a smartwatch, the vast majority of people already own and have with them their interaction platform.

The "dual-precision" family of pointing techniques [24] combine different input-to-output mappings (e.g. absolute vs. relative), input devices, and Control-Display (CD) gain functions, to different levels of speed and precision. Some require high-resolution vision-based instrumentation: the PRISM family of techniques [11, 12, 20], Hybrid RayToRelative Pointing [37], and Laser+Position [25] used Vicon tracking systems to simulate ray-casting for coarse pointing and relative translations for precise acquisition, with different switching mechanisms. Others were designed to only require everyday devices: ARC-Pad [22] and ARC-Pad2 [25] allow a coarse absolute and a precise relative modes depending on the characteristics of the user's touch on a smartphone. Finally, some techniques combine external instrumentation, typically for laser-like distant pointing, with everyday device features for precise pointing like touch for Laser+Track [24] and HeadPad [25].

Dual-precision techniques have been shown to perform well for distal pointing tasks on large displays [22, 37], but also to perform at least as good as well-tuned CD gain functions on smartphone-sized touch areas for Fitts' Indexes of Difficulty (ID) up to 6 bits [25]. Dual-precision, or larger input areas, is better for more difficult tasks and so-called "ultra-high resolution" displays, but such tasks are rarer on typical, projection-based displays. In what follows we focus on the latter, thus exploration of dual-precision techniques is left for future work.

Comparing distant pointing techniques

Two aspects influence the adoption of a tool for pointing input: convenience and performance. If an input device is ready-to-hand, then it is highly likely to be used for limited duration, occasional tasks; for example, consider the frequent use of the mouse as a three-dimensional input device [3]. If tasks are frequent and high-performance is required, this increases the likelihood that specialized devices will be adopted.

Considering, first, issues of convenience, one advantage of personal-device (e.g. smartphone or smartwatch) based pointing techniques is that the device belongs to the user and is typically carried with them. Alongside this, however, there are concerns associated with some techniques: for example, Katsuragawa et al.'s Watchpoint system set-up has participants wear the smartwatch on their preferred hand [18], presumably at odds with common practice. Furthermore, smartphone-based techniques that use the smartphone as a pointing device also assume a device held by the preferred hand [26], whereas the use of a smartphone as a touchpad proxy would assume the user holds the device with the non-preferred hand [2].

With smartphones, moving the device between hands has, presumably, limited cost (assuming the user is unemcumbered); however, for smartwatches, the need to switch the typical hand on which a watch is worn may limit convenience for Katsuragawa et al.'s Watchpoint technique.

The more significant trade-off we wish to explore in this paper involves relative performance of these input techniques. While many techniques, highlighted earlier, have been presented in the literature, there have been few controlled direct comparisons. Specifically in the domain of personal-device based pointing interactions, in their work on high-precision pointing and CD-gain, Nancel et al. [25] evaluated a series of devices, including an iPod touch with touchscreen, for high-precision pointing. However, this work contrasted a touchscreen with other more specialized devices rather than comparing across commodity hardware. Jain et al. [16] explore four different pointing techniques for smartphones when interacting with a smart television, including using the phone as a touchpad or using the IMU to measure direction. While they compare both cursor-based pointing and object-pointing [13] their task involves a grid-based smart television layout with large targets in a unique arrangement with variable gaps between targets.

In summary, while many techniques that leverage touchscreen or rotation of personal devices have been proposed and evaluated, we are unaware of any research that directly assesses these competing techniques in a controlled way.

SURVEY

As we noted in related work, two aspects determine the adoption of a pointing technique: convenience and performance. In this section, we present a brief survey to explore convenience factors for the various interaction techniques by eliciting, in particular, which hand or wrist is typically used to hold and control smartphones, smartwatches, and touchpads.

Survey Questions

Questions involved generic background information question (age, gender, country of residence, assessed handedness); whether the participant owns and uses a smartphone, wristwatch of any kind (including smartwatches), and specifically a smartwatch; and which side he uses to hold them during asymmetric bi-manual use: holding hand for smartphones, wearing wrist for watches. We also asked which hand they used on trackpads.

Dissemination

The survey was disseminated essentially through online social media, starting from public forums and the authors' own acquaintances. Participants were informed of the goals of the survey, and that their answers and data would be treated anonymously.

Results

Demographics: 365 people aged 17 to 73 ($\bar{x} = 33,8$, $\sigma = 10.2$) participated in the survey (48.3% female, 50.8% male, 0.8% 'other' or non disclosed), from 33 countries, essentially in Europe (73.8%), Oceania (10%), North America (9.7%), and Asia (6%). 87.4% of respondents considered themselves right-handed, 8.2% left-handed, and 4.5% to be ambidextrous.

Smartphones: Nearly all respondents (96.6%) declared being familiar with touch-based interaction on smartphones. When asked to imagine a situation where one hand holds the phone and the other interacts with the touchscreen, 86.7% replied that they would hold the phone with their non-dominant hand.

Trackpads: Most participants (86.2%) declared being familiar with trackpads, 91.7% of which would control a trackpad with their dominant hand.

Watches: Half the respondents (49.7%) declared usually wearing a wristwatch (of any kind), and most (85.4%) do–or would–wear it on their non-dominant wrist. By comparison, only 9% reported usually wearing a smartwatch. In the 50 participants (13.7%) that do wear one–or would consider doing so– 80% would wear it on their non-dominant wrist. All but three (94%) would wear it on the same side as they would another type of watch.

Summary

While not unexpected, the results of this survey provide an interesting overview on the wearing and interaction habits for smartphones and smartwatches. Interestingly, when it comes to touch-interaction, most participants declared interacting with their dominant hand, which means in the context of two-handed interaction that the device is held (or worn) on the non dominant hand, suggesting that mid-air bracelet based pointing techniques where a smartwatch is worn on the preferred hand [18, 14] might provide more disruption to users' existing practices.

POINTING TECHNIQUES

While convenience is one factor in assessing the likely adoption of a pointing technique, the primary form of assessment of any pointing technique is performance in terms of time and errors. To perform this assessment, and to support replicability of any assessment, this section provides implementation details on each of the pointing techniques evaluated.

The sensors typically available on a smartphone or smartwatch (touchscreens, IMU, etc.) can be used and combined in different ways to remotely control a cursor. Previous work has explored pointing techniques with multiple precision modes adapted to the different phases of a pointing action, sometimes combining more than one input device or type. However these techniques were developed with especially hard pointing tasks in mind, e.g. Fitts' Indexes of Difficulty (IDs) above 6 bits and frequently leveraged mode of interaction to support precise targeting and novel acceleration functions [24].

As a first step, and in order to keep this study under manageable time, we restrict this exploration to single-device, single-precision-mode techniques, and to Fitts' IDs that can reasonably be found on a typical computer screen. Our results can inform the design of more complex techniques in future work.

Design Space

Based on previous work, we considered four dimensions to define possible mid-air pointing techniques:

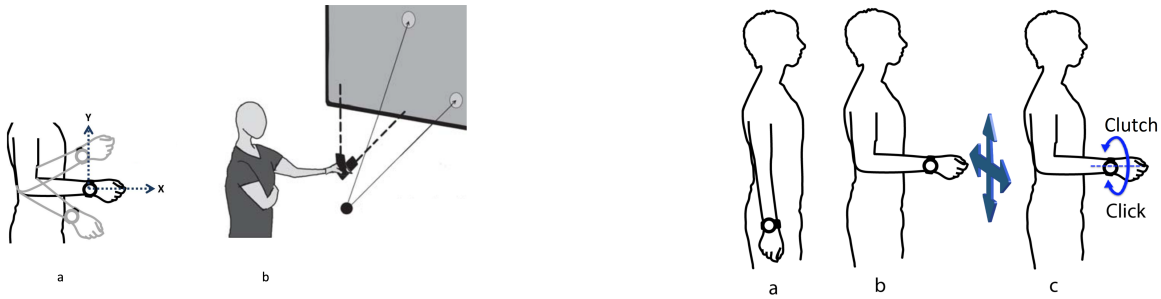


Figure 1. Left: Pointer manipulation in the rotation techniques, W1RR and P1RR, is governed by the orientation changes of the control device. (a) For W1RR, the user must sweep the entire forearm to cause changes orientation of the smartwatch. (b) For P1RR, the user can use the wrist sweeps as a more subtle form of manipulation; Right: W1RR can calibrate the center frame of reference from inactive state (a) to raising the arm in front (b) to switch to manipulation mode. From this mode, a 45° wrist flick outwards and back can be used to trigger a click action (c).

Device: the device itself, *smartphone* (P) or *smartwatch* (W). We decided to investigate these two devices as they are commonly owned and usually carried in various mobile scenarios making them suitable for serendipitous public display interaction scenarios.

Handedness: whether the technique requires *one* or *two* hands, Depending on the condition, operating the device with two hands can drastically impact ease of interaction. Note that all the techniques cannot be necessarily operated with one or two hands.

Control: whether the technique is controlled by *rotation* of the device (R) or *touch* input on a touchscreen (T). These two types of control have been proposed in the literature, can be deployed on devices already on the market, and involve different muscles which suggests different degrees of control and fatigue.

Mapping: whether the cursor's movements are *relative* to the user's movements (R), i.e. like a trackpad, or an *absolute* mapping to the location of the end effector (A), as during direct interaction with a touchscreen.

These dimensions should yield $2^4 = 16$ possible techniques. However, some combinations are impractical or impossible:

Rotation & Absolute (4 techniques): In this work, we focus on sensing capabilities embedded in the device itself. In the absence of external motion sensing, e.g. Vicon or Kinect systems, and without external reference point(s) such as with WiiMotes, we found no practical way to detect the *absolute* orientation of a device using a typical smartphone or smartwatch IMU.

Rotation & 2 Hands (4 techniques): Since we only consider single-device techniques, if the cursor is controlled by moving the device itself, the second hand is unused. We could imagine techniques where the second hand would be used for clicking, but it would require the second hand to track the device which would be constraining, awkward and of very little practicality.

Watch & Touch & Absolute (2 techniques): Typical smartwatches are too small for a 1:1-pixel mapping to most large displays. For instance, a target 144 mm wide on a 4.6 meter wide display, considered an easy condition in [14], would map onto a 1 mm wide area on a 1.3" (33 mm)-wide watch.

Watch & Touch & 1 Hand (2 techniques): The functional ranges of motion of the wrist joint make it impossible for a normal user to touch a smartwatch display with the fingers of the hand wearing the watch. As a result, one handed touch interactions on smartwatch have been discarded.

7 interaction techniques remain after discarding the 9 above mentioned. For shortness, in what follows the remaining techniques will be referred to as abbreviations (see Table 1).

		Rotation		Touch	
		Absolute	Relative	Absolute	Relative
Watch	1 Hand		W1RR		
	2 Hands				W2TR
Phone	1 Hand		P1RR	P1TA	P1TR
	2 Hands			P2TA	P2TR

Table 1. Summary of the evaluated techniques and dimensions. The abbreviations follow the ordering: Device-Hands-Input-Mapping.

Cursor positioning

Rotation techniques.

Both rotation techniques (W1RR and P1RR) map changes in device orientation (Δ_ω) on the Yaw and Pitch axis (in radian) to displacements of the cursor d, using the following function:

$$d = \Delta_\omega \times G(v)$$

$G(v)$ corresponds to the value returned by a piecewise linear CD gain function of the form:

$$G(v) = \min\left(\max\left(s \times v + i, G_{min}\right), G_{max}\right)$$

with v the five-sample average velocity change in orientation v in radians per second. The parameters G_{min}, G_{max}, i, and s are defined empirically. Changes in orientation around the Yaw (respectively Pitch) axis are mapped to x (respectively y) displacements of the cursor. P1RR relies on user wrist movements in order to tilt the device in its rotational axes. This technique, in concept, is similar to Tiltcasting by Pietroszek et al. [27]. W1RR is limited to movements of the whole forearm as the wrist tilt is naturally not sufficient to cause significant changes in the orientation of the watch. As a result, changing

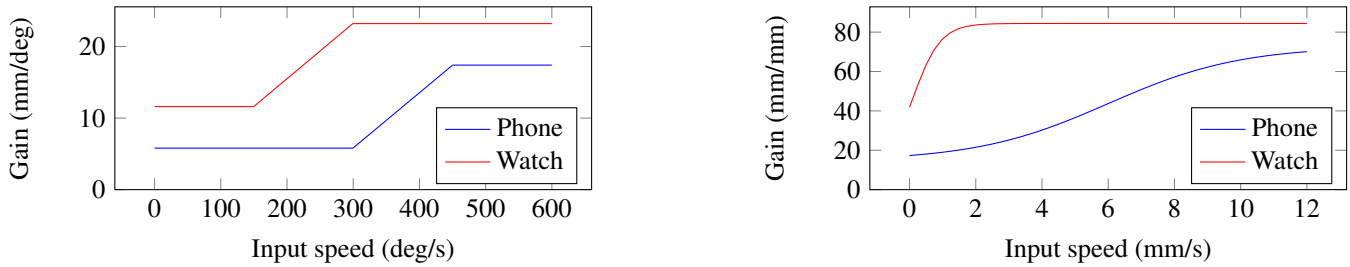


Figure 2. Left: Gain function for rotation techniques with the wrist-movement heavy phone technique being less sensitive than forearm-heavy watch condition; Right: Gain function plots for relative touch conditions for phone and watch.

the orientation of the watch involves different muscles and articulations than with the Phone (figure 1, Left). Therefore, we used different values for the gain functions depending on the operated device (see Figure 2, Left):

Technique	G_{min} ($^{mm}/_{deg}$)	G_{max} ($^{mm}/_{deg}$)	s ($^{\mu m.s}/_{deg^2}$)	i ($^{mm}/_{deg}$)
P1RR	5.8	17.4	77.1	-17.3
W1RR	11.6	23.2	77.5	0

Table 2. Piecewise linear function parameters for phone and watch relative rotation techniques. The parameters value for W1RR replicated those of [18]

In order to provide a clutching mechanism, a cursor's location is not updated for P1RR when the user's thumb is in contact with the display. However, as mentioned above, operating the touchscreen of a watch with the hand that wears it is impossible and alternatives had to be found to support clutching with W1RR. We applied Katsuragawa et al.'s design [18] (see figure 1, Right, which enters clutch mode when the user rolls the wrist more than 45° counter-clockwise – vs clockwise for drag/click – from its initial orientation (clutching stays on until the wrist returns to its initial orientation).

Relative touch-based techniques.

The relative touch techniques (P1TR, P2TR and W1TR) are operated via touch events on the display of the device, using the full display as a *trackpad*. We used a generalized logistic function [24] to define the gain between the move events on the display and cursor displacement:

$$G(v) = \frac{G_{max} - G_{min}}{1 + e^{(-\lambda \times (v - V_{inf}))}} + G_{min}$$

with G_{min} and G_{max} the asymptotic minimum and maximum gains, V_{inf} the input speed at which the function's inflection occurs, and λ a parameter proportional to the slope of the function at $v = V_{inf}$. The parameters used in the study were empirically optimized for each technique, and summarized in table 3.

Absolute touch-based technique.

With P1TA and P2TA, cursor position is automatically mapped to user's finger location on an interactive *trackpad* area displayed on the device of the smartphone. As absolute positioning requires the user to be able to position easily a finger at any location of the trackpad area. We used different areas depending on if the phone was operated with only one hand (thus holding it in portrait mode and interacting with the thumb) or with two hands. For P1TA, the trackpad area was 61mm × 40mm large, and displayed 20mm above the display bottom to accommodate thumb itneraction [6]. For P2TA, the device was intended to be operated in landscape mode and the trackpad area covered the full display.

Technique	G_{min}	G_{max}	λ	V_{inf}
PxTR	14.57 $^{mm}/_{mm}$	72.86 $^{mm}/_{mm}$	0.5 $^{1}/_{mm}$	6.0 $^{mm}/_{s}$
WxTR	8.44 $^{mm}/_{mm}$	84.4 $^{mm}/_{mm}$	2.41 $^{1}/_{mm}$	0.1 $^{mm}/_{s}$

Table 3. Logistic function parameters for phone and watch relative touch techniques.

Clicking triggers

For all the techniques where the user can reach the display, (P1TR, P2TR, P1TA, P2TA, W2TR, and P1RR), clicking is performed by tapping the trackpad area. For stability, taps have to be short enough ($T_{up} - T_{down} \leq 250$ ms) and do not update the location of the cursor. Note that for P1RR, since tapping might change the orientation of the smartphone, tapping had to be short ($T_{up} - T_{down} \leq 250$ ms) with a maximum rotation of the device of 99° in that timespan. For W1RR, we once again applied Katsuragawa et al.'s design [18] where wrist roll 45° counter-clockwise immediately followed by the opposite movement triggers a click (see figure 1, Right). Clutching and tapping are disambiguated with duration and displacement thresholds.

With all techniques, the cursor can jump at times, e.g. when the user presses or releases the touchscreen (due to a quick change in the detected finger area, or to minute phone rotations caused by releasing the finger in P1RR), or rolls her wrist (W1RR). To avoid this, when a click is detected, we fetched the cursor location corresponding to the lowest input speed detected in the previous 250 ms and used it as current cursor location, similar to [14].

STUDY

We compared the accuracy and performance of the 7 interaction techniques (W1RR, W2TR, P1RR, P1TA, P2TA, P1TR, P2TR) in a generic Fitts's Task experiment. Our experimental procedure is inspired by the Myopoint [14] and Watchpoint [18] experimental setups, as a basis to benchmark to form this comparative study.

Apparatus

We used a LG G Watch R (circular display of 33.44mm of diameter and a resolution of 320×320px, for a pixel

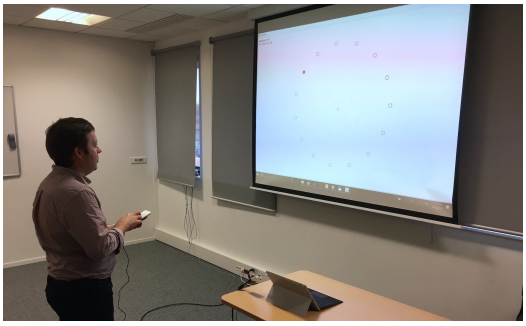


Figure 3. Illustration of the experimental setup: participants stood 2 meters away from a rear-projector screen. The input device was connected via USB to the computer in order to guarantee stable data transfer for smooth control.

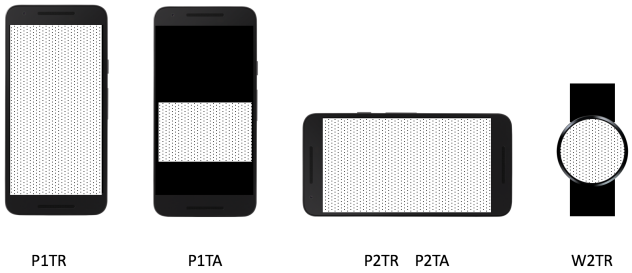


Figure 4. Touch techniques for both relative and absolute mapping use portrait orientation for one-hand and landscape orientation for two-hand use. The touch surface for P1TA is constrained to the width of the display, preserving the aspect ratio of the target device and positioned in the lower-half to accommodate one-handed use. The watch-based W1TR uses the round touch surface of the LG G Watch R.

density of 96.45px/cm) and a Nexus 5X (display of 116.20mm×65.36mm and resolution 1920×1080px, for a pixel density of 166.53px/cm) as input devices. Participants stood 2 meters away from a rear-projector screen of a resolution of 1920px × 1200px projected on an area of 1.68m width and 1.05m height (for a pixel density of 1.143px/mm), as illustrated on Figure 3. For the duration of the task, both the watch and the phone were connected to a laptop via USB for wired data transfer via socket in order to guarantee a stable and frequent data transfer for smooth control. Participant were asked to remove the watch while testing the smartphone techniques. For the smartwatch techniques (W2TR and W1RR), participants were free to place the smartwatch on their preferred wrist. For the P2TA and P2TR, participants were requested to use two hands to operate the smartphone, but were free to use the hand/fingers they wanted to hold/operate the device. Figure 4 illustrates the device orientation and the available touch surface for the touch interactions. The study took place in a closed lab environment at the university.

Design and Protocol

Participants were instructed to perform a sequence of pointing operations as quickly and accurately as possible. For each trial, participants had to select a target of a *width W* and located at a *distance D* from the initial position of the cursor. To select the target, participants had to position the cursor over the target and click on it using the provided interaction technique. The experimental software moved to the next trial only when the target was correctly selected. If target selection was incorrect, an audio cue was played. Once correctly selected, the current target was hidden and the next target, according to the configuration, was displayed on the screen (with only one target displayed on screen at a time). Target selection time was measured as the time between current target selection and the selection of the previous target. Participants had to select a "dummy" target at the beginning of each block, in order to control the initial position of the cursor and use its selection time as referent for measuring the selection time of the next target.

The experiment used a $2 \times 3 \times 9 \times 7$ within-subject design for the factors *Distance* (26.25cm, 43.75cm, 61.25cm), *Width* (1.75cm, 5.25cm), *Block* (1–9) and *Technique* (W1RR, W2TR, P1RR, P1TA, P2TA, P1TR, P2TR). Therefore, the index of difficulties (ID) of the experimental task ranged from 2.58 to 5.17 bits. The order of Technique was counterbalanced across participants using a Latin square.

To more accurately simulate real-world pointing where sequential targets differ in width and distance, the Distance and Width of the targets were generated randomly following a uniform distribution. For each technique, participants performed a sequence of 9 blocks of 12 targeting. We decided to use a block design that allows a block analysis intended to reveal significant learning effects, therefore making no assumption on the duration of the participants' learning phase (as opposed to fixed-size training sessions). In the end, participants performed a total of 12 targeting tasks per block per condition for a total of $12 \times 9 \times 7 = 756$ correct target selections per participant. While width and distance of each targeting task were selected randomly, we examined our distribution and verified uniformity within techniques.

Participants

We collected data from 21 participants aged from 21 to 45 ($\bar{x} = 26.71$, $\sigma = 4.79$), of which eight were female and two were left-handed, drawn from staff and students from the university campus. All participants were familiar with touch-based devices. No participant had previously owned a smartwatch but had owned either an activity tracker (e.g. a Fitbit) or a normal wristwatch. For W1RR, all participants wore the watch on the dominant hand in order to feel more confortable during the task. For W1TR, all participants wore the watch on the non-dominant hand and used the dominant hand to operate the touchscreen. For P1RR, P1TA and P1TR all participants held the device in their dominant hand. Participants received $10 remuneration, and the study lasted 45 to 50 minutes.

Results

We used mixed-design analyses of variance on the whole dataset (no aggregation), considering participant and trial numbers as random variables using the REML procedure of the SAS JMP package. Tukey post-hoc tests were conducted when significant effects were found.

In what follows we refer to completion time as CT, number of clicks outside the target as $\#_{err}$, and whether trials contained one or more errors as R_{err}.

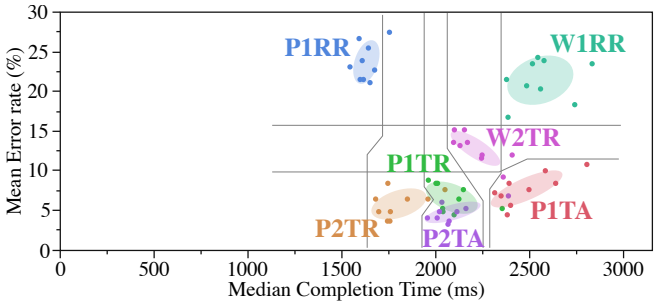


Figure 5. Bivariate density ellipses with 50% coverage of median CT and mean R_{err}, for each Technique and Block. Grey lines represent significant differences between techniques: vertical for CT and horizontal for R_{err}.

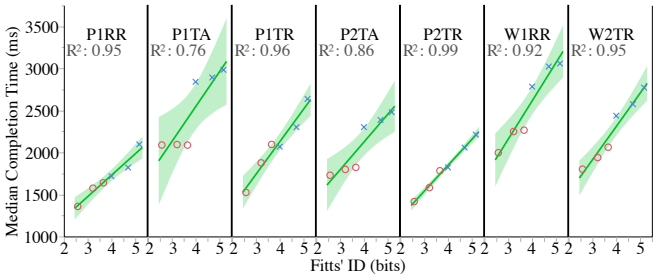


Figure 6. Median CT as a function of Fitts' ID per Technique, with corresponding confidence regions and R^2. Circles and crosses are respectively 60 mm and 20 mm targets.

Order Effects

We found no significant effect of the ordering of Techniques on either CT, $\#_{err}$, or R_{err}.

We did not find a significant effect of block number on $\#_{err}$ or R_{err}, but we found a significant effect of block number on CT ($F_{8,15836} = 13.67$, $p < 0.0001$): trials in Block 1 took significantly longer to complete (mean 2695.8 ms) than in all other blocks (means $\in [2263, 2509.8]$ ms).

We also found a significant difference between Block 2 (2509.8 ms) and Blocks 5, 8, 9, and 6 (mean CT$\leq$ 2324.5 ms), which remains after Block 1 is removed ($F_{7,14073} = 4.91$, $p < 0.0001$). No more significant difference can be observed if Block 2 is removed as well. However, Block 2 was never found significantly different from Blocks 3, 4, and 7, with or without Block 1, so we opted for the conservative approach and kept Block 2 in further analyses.

In what follows we took out Block 1 from our analyses, leaving 672 trials per participant (96 per technique).

Errors

We found similar significant effects of Technique on $\#_{err}$ ($F_{6,14074} = 142.4$, $p < 0.0001$) and R_{err} ($F_{6,14074} = 129.11$, $p < 0.0001$). In both cases, the Rotational techniques P1RR and W1RR caused significantly more errors than the rest (resp. 23.2% and 21.7% trials had erroneous clicks). Next came the remaining Watch technique, W2TR (12.8%), significantly more than the rest. The remaining techniques are the Phone+Touch techniques, P1TA, P1TR, P2TR, P2TA, with non-significantly different error rates between 4.4 and 7.2%. Significant differences are represented on Fig. 5 (vertical axis, horizontal lines). To summarize (the smaller the better):

P1TA, P1TR, P2TR, P2TA $\ll$ W2TR $\ll$ W1RR, P1RR

Since we allowed more than one click per target, we do not discard trials with errors in the analysis of completion time below, for it provides better insight on realistic use.

Completion Time

We found a significant effect of Technique on CT ($F_{6,14074} = 245.33$, $p < 0.0001$). The CT values were significantly different from each other for most Techniques. P1TA (mean 3038.5 ms) and W1RR (3001.7) were not significantly different from each other, but significantly slower than all other techniques. W2TR (2419.2) was slower than all remaining techniques. P2TA (2253.4) and P1TR (2169) were not significantly different from each other, but significantly slower than all remaining techniques. Finally, P2TR (1893.3) and P1RR (1719.9) were significantly different from each other and from all other techniques. Significant differences are represented on Fig. 5 (horizontal axis, vertical lines). To summarize (the smaller the better):

P1RR $\ll$ P2TR $\ll$ P1TR, P2TA $\ll$ W2TR $\ll$ W1RR, P1TA

Fitts' Law

As a sanity check, we confirmed that the techniques follow Fitts' law [10]. To remain comparable with previous work, we aggregated CT for each Technique and Fitts' Index of Difficulty (ID). We used medians to compensate for non-normal distribution of CT.

Fitts' IDs ranged from 2.58 to 5.17 bits. The aggregated times of all techniques correlate positively with Fitts' IDs (Fig. 6) P1RR, P1TR, P2TR, W1RR, and W2TR all have $R^2 \geq 0.92$. For the absolute techniques P1TA and P2TA (R^2 resp. 0.76 and 0.86), visual inspection reveals a much clearer effect of width than of distance, which is to be expected with absolute touch techniques with tap-clicking: after every selection the finger would go back to hovering above the screen, maintaining an approximately constant physical distance between the starting location of the finger and the target, as opposed to relative techniques in which the cursor must be moved from the previous target to the current one.

DISCUSSION

Comparison between techniques and dimensions

In terms of pure performance, using a smartphone as a bimanual relative trackpad (P2TR) is the "best" technique, since no other technique is significantly faster nor more precise. However, some situations might not allow use of both hands. In these situations, using a smartphone as a one-handed relative trackpad (P1TR), though slightly less efficient than P2TR, would be the next-best choice with respect to throughput.

Mappings

Mapping was only varied with touch-based techniques on the smartphone, for which error rates were not significantly different. Consistently with previous work [24], Relative techniques were significantly faster than their Absolute counterpart, for a given number of hands: P1TR $\ll$ P1TA and P2TR $\ll$ P2TA.

Since every Absolute technique has a faster (and similarly precise) counterpart, all other dimensions being kept equal, we advise against absolute pointing techniques.

Input

The Rotation techniques with the phone (P1RR) and watch (W1RR) resulted in a significantly higher error rate (19.8%).

Unlike for touch-based techniques, the rotation-based techniques do not allow a user to trigger a click when the cursor is "locked". To understand this point, consider that when clicking in the touch conditions (W2RT, PXXT) a user would lift his or her finger off the display and then tap to touch, ensuring cursor stability during the click. In contrast, with P1RR/W1RR, the cursor would move as the wrist rotated or the user attempted to tap the screen. While we did try to correct for this cursor tremor by looking for stability over a 250ms window, it is not clear that it was entirely eliminated. We can assume that several of the errors in rotation-based techniques (W1RR or P1RR) were the result of clicking attempts that resulted in first moving the cursor out of the target before the click was triggered. Enhancements in cursor stabilization might improve the error rate of these techniques.

Hands and Devices

The best one-handed technique consists in using a smartphone like a trackpad with the thumb (P1TR). It is second only to P2TR; P1RR is faster but much less precise. Using a smartwatch screen like a trackpad (W2TR) performed surprisingly well, considering the very small input range (1.3") and its circular shape.

Choice of technique in practice

If both the user's hands are available, then the best technique is to use the smartphone as a trackpad held in the non-dominant hand (P2TR). Alternatively, if the smartphone is unavailable and the user owns a smartwatch, then using the watch's screen as a trackpad provides acceptable performance overall.

If the user is only willing to use one hand to point, or only one hand of the user is available, then the best technique is to use the smartphone as a trackpad controlled with the thumb. Alternatively, if too little of the screen is available to point, e.g. if numerous controls need to be displayed on screen, then the user can control pointer movements through rotation using only a small part of the screen for clutch and click control. This however has a cost in performance.

However, consider Figure 5. If one can lower the error rate of the rotation-based techniques, for example with better cursor stabilization, the completion times of all techniques are relatively tightly clustered. The trade-off then becomes one of convenience: For example, if a user's smartphone is in his or her pocket, a smartwatch may serve almost equally as effectively for input.

	Both hands holding something? →W1RR
else	Phone unavailable? →W2TR
else	Limited space on phone screen?→P1RR
else	One hand holding something? →P1TR
else	→ P2TR

Future Work

The controlled experiment reported in this paper compared several relative and absolute pointing techniques, using of The controlled experiment reported in this paper compared, in an abstract task, seven distal pointing techniques on phone or smartwatch, using either a relative or an absolute mapping. Our work focused on internal validity to facilitate performance comparison between numerous candidate techniques and identify the most efficient techniques for each device and control method in a 2D pointing task. However, other elementary tasks such as hierarchical navigation or command selection should be investigated in future work. Future work should also focus on conducting experiments focused on external validity. Of particular interest would be field experiments (e.g. on a public display using simple games or compelling information display) in order to test interaction techniques on less abstract tasks, and evaluate device connection strategies, as well as their time and usability costs.

CONCLUSION

Our work investigates the use of handheld devices, the smartphone and the smartwatch, as remote pointing devices to interact with large displays. We first explore the design space for high-level dimensions and describe the implementation of 7 feasible techniques. Through a controlled pointing study, we suggest the use of the smartphone as a relative trackpad, utilizing landscape orientation and two hands for best performance. We then explore design challenges and offer alternative recommendations for dimensionally-constrained contexts such as when one or both hands are unavailable for use. Most importantly, our work demonstrates the sensor and touch capabilities of modern smart devices allow for rich, flexible, and always-available interactions, which can be dynamically adjusted for real-world contexts without expensive and rigid augmentations to the environment or the large display.

ACKNOWLEDGMENTS

We gratefully acknowledge the financial support of the Natural Science and Engineering Research Council of Canada, NSERC, and the Programme Conférencier Invité, Région Nord Hauts-de-France. We also thank the Technical and General Services from the Inria Lille - Nord Europe research center for their help with the experimental setup.

REFERENCES

1. David Ahlström, Khalad Hasan, and Pourang Irani. 2014. Are You Comfortable Doing That?: Acceptance Studies of Around-device Gestures in and for Public Settings. In *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services (MobileHCI '14)*. ACM, New York, NY, USA, 193–202. DOI: http://dx.doi.org/10.1145/2628363.2628381

2. Matthias Baldauf, Peter Fröhlich, Jasmin Buchta, and Theresa Stürmer. 2013. From touchpad to smart lens: A comparative study on smartphone interaction with public displays. *International Journal of Mobile Human Computer Interaction (IJMHCI)* 5, 2 (2013), 1–20.

3. François Bérard, Jessica Ip, Mitchel Benovoy, Dalia El-Shimy, Jeffrey R. Blum, and Jeremy R. Cooperstock. 2009. *Did "Minority Report" Get It Wrong? Superiority of the Mouse over 3D Input Devices in a 3D Placement Task*. Springer Berlin Heidelberg, Berlin, Heidelberg, 400–414. DOI: http://dx.doi.org/10.1007/978-3-642-03658-3_45

4. Louis-Pierre Bergé, Marcos Serrano, Gary Perelman, and Emmanuel Dubois. 2014. Exploring Smartphone-based Interaction with Overview+Detail Interfaces on 3D Public Displays. In *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services (MobileHCI '14)*. ACM, New York, NY, USA, 125–134. DOI: http://dx.doi.org/10.1145/2628363.2628374

5. Sebastian Boring, Dominikus Baur, Andreas Butz, Sean Gustafson, and Patrick Baudisch. 2010. Touch Projector: Mobile Interaction Through Video. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, New York, NY, USA, 2287–2296. DOI: http://dx.doi.org/10.1145/1753326.1753671

6. 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. DOI:http://dx.doi.org/10.1145/2371574.2371582

7. Emeline Brulé, Gilles Bailly, Marc Serrano, Marc Teyssier, and Samuel Huron. 2017. Explorer le potentiel des interactions tangibles rotatives pour les Smart Watches. In *29ème conférence francophone sur l'Interaction Homme-Machine*. ACM, 8–p.

8. Bryan A. Campbell, Katharine R. O'Brien, Michael D. Byrne, and Benjamin J. Bachman. 2008. Fitts' Law Predictions with an Alternative Pointing Device (Wiimote®). *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 52, 19 (2008), 1321–1325. DOI: http://dx.doi.org/10.1177/154193120805201904

9. James Davis and Xing Chen. 2002. Lumipoint: Multi-user laser-based interaction on large tiled displays. *Displays* 23, 5 (2002), 205–211.

10. Paul M Fitts. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology* 47, 6 (1954), 381.

11. Scott Frees, G. Drew Kessler, and Edwin Kay. 2007. PRISM Interaction for Enhancing Control in Immersive Virtual Environments. *ACM Trans. Comput.-Hum. Interact.* 14, 1, Article 2 (May 2007). DOI: http://dx.doi.org/10.1145/1229855.1229857

12. L. Gallo and A. Minutolo. 2012. Design and Comparative Evaluation of Smoothed Pointing: A Velocity-oriented Remote Pointing Enhancement Technique. *Int. J. Hum.-Comput. Stud.* 70, 4 (April 2012), 287–300. DOI: http://dx.doi.org/10.1016/j.ijhcs.2011.12.001

13. Yves Guiard, Renaud Blanch, and Michel Beaudouin-Lafon. 2004. Object Pointing: A Complement to Bitmap Pointing in GUIs. In *Proceedings of Graphics Interface 2004 (GI '04)*. Canadian Human-Computer Communications Society, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 9–16. http://dl.acm.org/citation.cfm?id=1006058.1006060

14. Faizan Haque, Mathieu Nancel, and Daniel Vogel. 2015. Myopoint: Pointing and Clicking Using Forearm Mounted Electromyography and Inertial Motion Sensors. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 3653–3656. DOI: http://dx.doi.org/10.1145/2702123.2702133

15. Khalad Hasan, David Ahlström, Junhyeok Kim, and Pourang Irani. 2017. AirPanes: Two-Handed Around-Device Interaction for Pane Switching on Smartphones. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 679–691. DOI: http://dx.doi.org/10.1145/3025453.3026029

16. Mohit Jain, Andy Cockburn, and Sriganesh Madhvanath. 2013. Comparison of Phone-Based Distal Pointing Techniques for Point-Select Tasks. In *Human-Computer Interaction – INTERACT 2013: 14th IFIP TC 13 International Conference, Cape Town, South Africa, September 2-6, 2013, Proceedings, Part II*, Paula Kotzé, Gary Marsden, Gitte Lindgaard, Janet Wesson, and Marco Winckler (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 714–721. DOI: http://dx.doi.org/10.1007/978-3-642-40480-1_49

17. Ricardo Jota, Miguel A. Nacenta, Joaquim A. Jorge, Sheelagh Carpendale, and Saul Greenberg. 2010. A Comparison of Ray Pointing Techniques for Very Large Displays. In *Proceedings of Graphics Interface 2010 (GI '10)*. Canadian Information Processing Society, Toronto, Ont., Canada, Canada, 269–276. http://dl.acm.org/citation.cfm?id=1839214.1839261

18. 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

19. Georgios Kouroupetroglou, Alexandros Pino, Athanasios Balmpakakis, Dimitrios Chalastanis, Vasileios Golematis, Nikolaos Ioannou, and Ioannis Koutsoumpas. 2012. *Using Wiimote for 2D and 3D Pointing Tasks: Gesture Performance Evaluation*. Springer Berlin Heidelberg, Berlin, Heidelberg, 13–23. DOI: http://dx.doi.org/10.1007/978-3-642-34182-3_2

20. Werner König, Jens Gerken, Stefan Dierdorf, and Harald Reiterer. 2009. Adaptive pointing : implicit gain adaptation for absolute pointing devices. In *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems - CHI EA '09*. ACM Press, New York, New York, USA. DOI: http://dx.doi.org/10.1145/1520340.1520635

21. Tanwi Mallick, Partha Pratim Das, and Arun Kumar Majumdar. 2014. Characterizations of noise in Kinect depth images: A review. *IEEE Sensors journal* 14, 6 (2014), 1731–1740.

22. David C. McCallum and Pourang Irani. 2009. ARC-Pad: Absolute+Relative Cursor Positioning for Large Displays with a Mobile Touchscreen. In *Proceedings of the 22Nd Annual ACM Symposium on User Interface Software and Technology (UIST '09)*. ACM, New York, NY, USA, 153–156. DOI: http://dx.doi.org/10.1145/1622176.1622205

23. Brad A. Myers, Herb Stiel, and Robert Gargiulo. 1998. Collaboration Using Multiple PDAs Connected to a PC. In *Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work (CSCW '98)*. ACM, New York, NY, USA, 285–294. DOI: http://dx.doi.org/10.1145/289444.289503

24. Mathieu Nancel, Olivier Chapuis, Emmanuel Pietriga, Xing-Dong Yang, Pourang P Irani, and Michel Beaudouin-Lafon. 2013. High-precision Pointing on Large Wall Displays Using Small Handheld Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 831–840.

25. Mathieu Nancel, Emmanuel Pietriga, Olivier Chapuis, and Michel Beaudouin-Lafon. 2015. Mid-Air Pointing on Ultra-Walls. *ACM Trans. Comput.-Hum. Interact.* 22, 5, Article 21 (Aug. 2015), 62 pages. DOI: http://dx.doi.org/10.1145/2766448

26. Krzysztof Pietroszek, Anastasia Kuzminykh, James R. Wallace, and Edward Lank. 2014. Smartcasting: A Discount 3D Interaction Technique for Public Displays. In *Proceedings of the 26th Australian Computer-Human Interaction Conference on Designing Futures: The Future of Design (OzCHI '14)*. ACM, New York, NY, USA, 119–128. DOI: http://dx.doi.org/10.1145/2686612.2686629

27. Krzysztof Pietroszek, James R Wallace, and Edward Lank. 2015. Tiltcasting: 3D Interaction on Large Displays Using a Mobile Device. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. ACM, New York, NY, USA, 57–62.

28. Alexandros Pino, Evangelos Tzemis, Nikolaos Ioannou, and Georgios Kouroupetroglou. 2013. *Using Kinect for 2D and 3D Pointing Tasks: Performance Evaluation*. Springer Berlin Heidelberg, Berlin, Heidelberg, 358–367. DOI:http://dx.doi.org/10.1007/978-3-642-39330-3_38

29. Jacob Poushter. 2016. Smartphone ownership and internet usage continues to climb in emerging economies. *Pew Research Center* 22 (2016).

30. Umar Rashid, Jarmo Kauko, Jonna Häkkilä, and Aaron Quigley. 2011. Proximal and Distal Selection of Widgets: Designing Distributed UI for Mobile Interaction with Large Display. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI '11)*. ACM, New York, NY, USA, 495–498. DOI: http://dx.doi.org/10.1145/2037373.2037446

31. Houssem Saidi, Marcos Serrano, and Emmanuel Dubois. 2017. Objets du quotidien pour interagir avec des environnements multi-écrans publics. In *29ème conférence francophone sur l'Interaction Homme-Machine*. 10–p.

32. Stefan Schneegass. 2015. There is More to Interaction with Public Displays Than Kinect: Using Wearables to Interact with Public Displays. In *Proceedings of the 4th International Symposium on Pervasive Displays (PerDis '15)*. ACM, New York, NY, USA, 243–244. DOI: http://dx.doi.org/10.1145/2757710.2776803

33. Julian Seifert, Andreas Bayer, and Enrico Rukzio. 2013. *PointerPhone: Using Mobile Phones for Direct Pointing Interactions with Remote Displays*. Springer Berlin Heidelberg, Berlin, Heidelberg, 18–35. DOI: http://dx.doi.org/10.1007/978-3-642-40477-1_2

34. Garth Shoemaker, Anthony Tang, and Kellogg S. Booth. 2007. Shadow Reaching: A New Perspective on Interaction for Large Displays. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology (UIST '07)*. ACM, New York, NY, USA, 53–56. DOI:http://dx.doi.org/10.1145/1294211.1294221

35. Garth Shoemaker, Takayuki Tsukitani, Yoshifumi Kitamura, and Kellogg S. Booth. 2010. Body-centric Interaction Techniques for Very Large Wall Displays. In *Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries (NordiCHI '10)*. ACM, New York, NY, USA, 463–472. DOI:http://dx.doi.org/10.1145/1868914.1868967

36. Daniel Vogel and Ravin Balakrishnan. 2004. Interactive Public Ambient Displays: Transitioning from Implicit to Explicit, Public to Personal, Interaction with Multiple Users. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology (UIST '04)*. ACM, New York, NY, USA, 137–146. DOI: http://dx.doi.org/10.1145/1029632.1029656

37. Daniel Vogel and Ravin Balakrishnan. 2005. Distant Freehand Pointing and Clicking on Very Large, High Resolution Displays. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology (UIST '05)*. ACM, New York, NY, USA, 33–42. DOI:http://dx.doi.org/10.1145/1095034.1095041

38. Florian Vogt, Justin Wong, Barry A Po, Ritchie Argue, S Sidney Fels, and Kellogg S Booth. 2004. Exploring collaboration with group pointer interaction. In *Computer Graphics International, 2004. Proceedings*. IEEE, 636–639.

39. Ka-Ping Yee. 2003. Peephole Displays: Pen Interaction on Spatially Aware Handheld Computers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '03)*. ACM, New York, NY, USA, 1–8. DOI: http://dx.doi.org/10.1145/642611.642613