Performance Envelopes of In-Air Direct and Smartwatch Indirect Control for Head-Mounted Augmented Reality

Dennis Wolf[*]
University of Cambridge
Ulm University

John J. Dudley[†]
University of Cambridge

Per Ola Kristensson[‡]
University of Cambridge

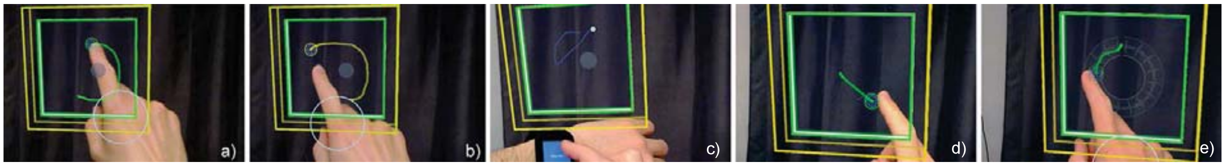


Figure 1: A view of the experiment interface with the two input techniques and three tasks (target acquisition (a-c), goal crossing (d), and circular steering (e)) investigated in the paper. The direct in-air cursor follows the hand position and leaves a green trail to indicate proximity to the ideal selection plane (a) and a yellow trail to prompt the user to adjust the cursor depth (b). The indirect smartwatch cursor is controlled with the smartwatch (c).

Abstract

The scarcity of established input methods for augmented reality (AR) head-mounted displays (HMD) motivates us to investigate the performance envelopes of two easily realisable solutions: indirect cursor control via a smartwatch and direct control by in-air touch. Indirect cursor control via a smartwatch has not been previously investigated for AR HMDs. We evaluate these two techniques for carrying out three fundamental user interface actions: target acquisition, goal crossing, and circular steering. We find that in-air is faster than smartwatch ($p < 0.001$) for target acquisition and circular steering. We observe, however, that in-air selection can lead to discomfort after extended use and suggest that smartwatch control offers a complementary alternative.

Keywords: Fitts' law; steering law; goal crossing; in-air selection; smartwatch; indirect cursor; AR; augmented reality

Index Terms: Human-centered computing—Human computer interaction (HCI)—Interaction paradigms—Mixed / augmented reality

1 Introduction

We envisage a future in which the smartphone is replaced by an interactive near-eye display (NED). The need to preserve mobility and interactivity within the physical world suggests that such devices will deliver augmented reality (AR) experiences rather than purely virtual ones. To enable this vision it is imperative that users are supplied with appropriate techniques for making selections and manipulating interface elements in AR. These techniques must not only be fast and accurate but must also accommodate other usability considerations such as comfort and social acceptability. This future vision motivates research into how both established and novel input techniques perform in the context of head-mounted AR devices. We anticipate that the ideal solution will not be provided by a single input technique but by a suite of complementary alternatives with fluid transitions between techniques.

In this paper we evaluate and compare direct hand-tracking based cursor control and selection with an indirect alternative based on a smartwatch. In the context of this paper we refer to the hand-tracking based technique as *in-air*. The in-air technique is intentionally closely aligned to the physical act of touching, sliding and dragging real world objects. The cursor position is directly controlled by movement of the hand. The smartwatch based technique leverages the familiarity of a touch-screen but at a reduced form factor. Importantly, the smartwatch interaction technique as explored here is indirect in the sense that the touch point on the screen is mapped to a corresponding point in the AR environment.

Among the multitude of possible interaction techniques suited to AR, we examine the smartwatch and in-air techniques for primarily two reasons. First, in both techniques the user's hands remain unencumbered; leaving them free to interact with the environment. Second, we hypothesise that the two techniques are *complementary*: in-air movements are high amplitude but low precision and smartwatch inputs are low amplitude but high precision. In addition to the two reasons listed above, both techniques can be immediately delivered through currently available consumer hardware.

This paper reports on the evaluation of the in-air and smartwatch techniques for three well established fundamental user interface actions: target acquisition, goal crossing and circular steering. These fundamental *actions* combine in designs to form high-level interaction *tasks* (that is, sequences of actions and decisions). Actions provide an experimentally controllable and standardised basis for comparing the two techniques. The evaluation focuses on comparing performance, that is, time (with error controlled), using established mathematical models of human performance for target acquisition, goal crossing, and circular steering. In addition, we report participants' qualitative responses when carrying out the experiment to capture indications of additional factors, such as comfort.

Understanding the performance envelope, that is, the strengths and weaknesses of an interaction technique over typical interface element sizes, provides valuable design insight and can aid both manual design of AR interfaces and be fed into a user interface optimization process, for example by incorporating the human performance model parameters from our experiment into objective functions.

In summary, our contributions are:

1. A validation of smartwatches as indirect control devices for HMD AR.
2. An exploration of the performance envelopes and model parameters for three fundamental user interface actions using in-air direct and smartwatch indirect control.
3. A set of design implications distilled from the experiment.

[*] e-mail: dennis.wolf@uni-ulm.de
[†] e-mail: jjd50@cam.ac.uk
[‡] e-mail: pok21@cam.ac.uk

2018 IEEE Conference on Virtual Reality and 3D User Interfaces
18-22 March, Reutlingen, Germany
978-1-5386-3365-6/18/$31.00 ©2018 IEEE

2 Related Work

The in-air technique examined in this work represents an extension upon a well established body of research. Historically, most in-air interactions with virtual content have relied on external sensing, requiring markers [7] and/or input devices that encumber the user [25]. Mobile, hand-held devices are nevertheless effective and indeed necessary in certain contexts. Users will naturally prefer a mobile, hand-held input device over tethered cables [4]. We do not argue that hand-held devices have no place with AR HMDs but rather that their use may be precluded in certain use cases, for instance, for a technician who must frequently interact with different physical objects.

Marker-less hand-tracking as supported by the Microsoft HoloLens is a major step forward towards free-hand interaction with mixed virtual-physical environments. The user remains mobile and unencumbered thanks to egocentric sensing. However, in-air interaction based on the physical hand position is not always suitable or preferable. For example, a user may seek to interact with objects that are out of reach or distant. As a consequence, there is potential value in virtual pointer techniques that augment the hand position in some way. For example, a pointer could be presented as a ray extending the user's hand [8] or a virtual hand as defined by Poupyrev et al. [20]. Poupyrev et al. [20] showed that virtual pointers can be efficient to select distant targets, but suffer a loss of accuracy with increasing distance.

An alternative to augmenting the physical hand position through virtual pointers is the use of additional complementary input devices. Ohta et al. [19] proposed smartwatch-based navigation and interaction in a virtual shop to facilitate the shopping process for disadvantaged users, for example senior citizens. AR representations of products could be selected and viewed from various angles via direct selection on the smartwatch touchscreen. To increase selection precision and enable interaction with distant displays, Hartmann et al. [11] proposed to reintroduce mouse-based interaction for tabletops, while Mane et al. [17] argue that precise interactions in a virtual screen should be performed on a two-dimensional touchscreen. Touchscreen controlled cursors have already been applied in the public screen context for multiple users [18], single user [21], and presentation slides [6]. Huo et al. [13] demonstrate the potential value of using a touchscreen to draw projected lines onto physical surfaces. Similarly, we argue that in-air direct selection can be meaningfully complemented by indirect smartwatch-based cursor selection.

Rohs and Oulasvirta [23] explore the performance of "peephole" pointing in the context of magic lens interfaces for large displays. When there is no background context and the peephole content is generated dynamically, they found that Fitts' law does hold. However, when background context is present, they observe that Fitts' law does not accurately model the performance of target selection using a magic lens interface. They hypothesize that this is due to the disruption in the visual feedback loop as the lens is moved over the background scene. In response, they propose a two-part model that separates the physical and virtual stages of the pointing task which improves the movement time prediction model. They subsequently validate that this model holds in a real-world AR task in which the user must target buildings using the "peephole" interface [24]. The investigation of Rohs and Oulasvirta is an interesting example of complex interactions introduced by the human visual and motor control system and highlights the value of examining new and fundamentally different forms of selection control.

3 Interface Actions and Performance Models

We examine three user interface actions: target acquisition, goal crossing, and circular steering. Each of these actions have been extensively investigated in the literature and robust human performance models have been derived for each of them. We also highlight the relevance of these different interface actions from an interaction design perspective.

3.1 Target Acquisition

Target acquisition is the act of moving to and then designating (e.g. selecting) a desired target. Fitts' law predicts that the Movement Time (MT) required to make a selection is linearly proportional to the Index of Difficulty (ID) of the target. Index of Difficulty is a non-dimensional metric describing the relative difficulty associated with the geometry of the selection interface. The relationship between (MT) and (ID) can be expressed as

$$MT = a + bID, \tag{1}$$

where a and b are linear regression coefficients. According to the Shannon formulation of Fitts' law [16], ID is defined as

$$ID = \log_2\left(\frac{D}{W} + 1\right), \tag{2}$$

where D is the distance to the target, and W is the width of the target. Throughput (TP) is often used as a summative measure representing the efficiency of the target acquisition method, i.e. an efficient method accommodates difficult targets nearly as well as easy targets. Throughput is computed as the inverse of the regression line gradient, that is, $TP = 1/b$.

Target acquisition can be thought of as an abstraction of the basic task of pressing a button. Even typing on a keyboard can be framed as a sequential target acquisition task [5]. Target acquisition thus represents a fundamental interface action.

3.2 Goal Crossing

Goal crossing is the act of intersecting the boundaries of an object, for example entering a button. While acquiring small targets can be time-consuming, crossing targets are only limited in one dimension, that is, the target width perpendicular to cursor movement. Accot and Zhai [2] discovered that the mathematical formulation for the movement time model of target acquisition (Fitts' law), also models goal crossing, that is, crossing targets. If W is target width and D the distance to the target then the movement time to cross the target is modelled by Equations 1 and 2.

Goal crossing as an action can be exploited to support rapid sequential selection of interface elements. However, crossing is not widely used in modern 2D interfaces despite its demonstrated potential. This is in part due to the fact that the majority of crossing research derives from pen-based interaction contexts. For example, Apitz and Guimbretière [3] demonstrate a simple drawing application designed to exploit the fluidity and speed of crossing interactions with a pen on a tablet. Apitz and Guimbretière [3] also motivate the use of the crossing action as a means to accommodate the typically noisy trajectories associated with pen input.

FlowMenu [10] is another example of a feature rich interface element built to exploit crossing actions. FlowMenu supports the fluid navigation of hierarchical menus with minimal motion trajectories. As a consequence, menu elements can remain small and the interactions inherently accommodate small display sizes. This example has particular relevance to the AR scenarios examined in this paper for two reasons. First, the display region of currently available AR NEDs is relatively narrow, limiting the size and placement of visual interface elements. Second, using a smartwatch as an input surface is constrained by the small screen size limiting the shape and size of potential trajectories.

Luo and Vogel [15] evaluate crossing-based selection on a touch input surface. They highlight that the value of a crossing-based interface is only realised when continuous selections can be chained together, and that, "if [there are] many discrete crossing actions, the benefit is lost" [15]. It is thus important to be aware of the influence that usage context may have on the practical benefits of a particular interface action.

3.3 Circular Steering

Circular steering is a trajectory-based interaction where a cursor is navigated through a circular tunnel without leaving its boundaries. Accot and Zhai [1] found that the movement time for steering through a circular tunnel of a given radius R and width W can be predicted by

$$MT = a + bID_C, \tag{3}$$

where circular Index of Difficulty (ID_C) is defined as

$$ID_C = \frac{2\pi R}{W}. \tag{4}$$

Circular steering throughput, TP_C is computed as the inverse of the regression line gradient, $TP_C = 1/b$.

The linear equivalent of circular steering is frequently encountered in the navigation of hierarchical menus. Navigating through a circular tunnel is a reflection of the general ability to follow a curved trajectory. Bounded curved trajectories may be encountered in dial type interface elements or as part of other selection operations such as lassoing [26]. Circular steering is significantly more difficult than an equivalent linear steering configuration due to the additional coordination required [1].

4 Technique Evaluation

This section describes the experimental setup and procedure used to evaluate the in-air and smartwatch input techniques for target acquisition, goal crossing, and circular steering. The experiment was a within-subjects design with 20 participants (10 female) recruited via convenience sampling. Participants were aged between 19 and 60 (mean = 33, sd = 12.3). No participant had prior experience with an AR HMD and only four had previously used a smartwatch.

4.1 Apparatus

The system for the experiment consisted of three components:

- A *Microsoft HoloLens* served as the interactive near-eye display. In addition to providing the interface display environment, the device also provided the coarse hand tracking functionality necessary for the in-air direct cursor control technique. The HoloLens application also acted as a server for receiving and handling the touch events reported by the smartwatch.
- A *Sony Smartwatch 3* running on Android 6.0.1 (API 23) with a screen resolution of 320 × 320 pixels on a 28.7 × 28.7 mm display. The smartwatch ran a client application that registered touch events and reported these to the HoloLens over a TCP connection.
- A dedicated wireless router provided the TCP communication layer between the smartwatch and the HoloLens.

4.2 Interface for Experiment

The interface for the experiment was presented at a distance of approximately 0.5 m from the user and had an apparent real world size of approximately 200 × 200 mm. This size was chosen as it approximately represented the maximum size of an interface that completely fits within the HoloLens' display region when presented at a distance comfortably reachable by the user.

The HoloLens provides coarse hand tracking and the reported position was used to approximate the location of the index finger. It is important to note that the hand tracking does not provide any articulation information and so the index finger location is only approximated. A cursor, referred to subsequently as the *index cursor*, is placed at this location. The user can control the index cursor by moving their hand within view of the headset. Note the feasible tracking region is considerably larger than the display region and so tracking loss was not an issue in this experiment.

A three-dimensional gesture pane was implemented to enable continuous in-air gestures rather than just simple touches. This gesture pane provides feedback to the user on the deviation of the index cursor from the central plane. This feedback helps restrict the user to performing gestures in a fixed plane (within an allowed tolerance). When the index cursor was inside the gesture pane and within the required tolerance of the central plane, the user was presented with a green line trail (see Figure 1a and Figure 2). If the index cursor exceeded an intermediary tolerance threshold, the user was presented with a yellow line trail (see Figure 1b), indicating that they should adjust their depth position to stay within the ideal gesture pane.

All touch events on the smartwatch were sent via a TCP socket to the HoloLens and rendered on the same gesture pane described above. A cursor indicated the most recent touch event and a trailing path of fixed point length was shown (see Figure 3). The average cursor update interval was 30 ms, which allowed users a smooth selection with an average delay between touch event and cursor update of 50 ms.

Every successful selection in both input methods was confirmed by a sound. All selections were calculated by detecting three-dimensional collisions between the index cursor and the target mesh (in-air selection) or by collapsing all three-dimensional pointer coordinates onto a two dimensional interface where a simple boundary check was performed (smartwatch selection).

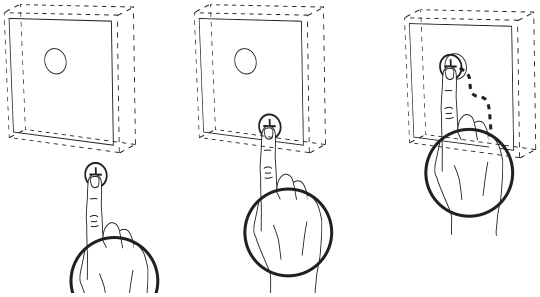


Figure 2: Direct in-air cursor. The current index cursor position is illustrated by a circle and a cross. A trail of fixed length displays the recently traced path. In the experiment interface the color of the cursor trail was used to indicate proximity to the ideal gesture pane and to cue participants to regulate their depth (green trail: within tolerance, yellow trail: adjust depth to meet tolerance).

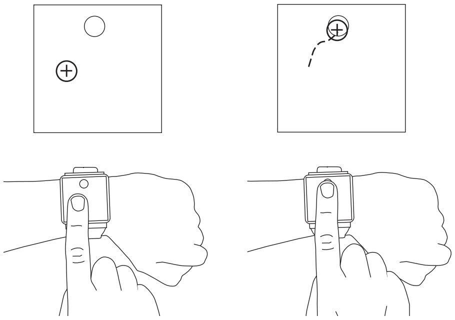


Figure 3: Indirect smartwatch cursor. The most recent touch point is displayed as a cursor. A trail of fixed length indicates the recently traced path.

4.3 Procedure

We conducted a within-subjects experiment, using a factorial design with two conditions:

- **SMARTWATCH**: Indirect cursor control based on smartwatch touch surface.
- **IN-AIR**: Direct cursor control based on coarse hand motion tracking.

All tasks used lift-off to indicate a selection. The reason for a lift-off metaphor lies in the nature of both conditions. Triggering targets in-air based on collision alone would lead to the "Midas touch problem" [14] and select every interface item on the cursor path when used in combination with a rich user interface. Therefore, an explicit selection has to be made by leaving the gesture pane.

Further, the smartwatch condition offers no visual feedback in the interface when there is no touch event. As a consequence, selecting targets on an instantaneous touch alone would render a very low success rate and would not comply with Fitts' law behaviour. Instead, we allow the user to move the cursor, while contact with the touch surface is maintained, and designate an explicit selection by the lift-off event.

The dependent variable was Movement Time (*MT*). Each condition was tested for user interface actions: target acquisition, goal crossing, and circular steering. The order of the tasks was allocated with a $3{\times}3$ Latin square. Target parameters were balanced with a balanced Latin square.

Prior to commencing the experiment, participants were encouraged to familiarise themselves with each selection technique and task. During the familiarisation phase, participants were free to repeat any task until they felt comfortable with their performance. The order of conditions was balanced with the first condition alternating for each consecutive participant. All tasks were completed in one condition before proceeding to the next condition. We encouraged all participants to take off the HMD between each condition to rest their eyes and relax their arms.

Participants performed the experiment while seated and facing a dark flat-colour background. The smartwatch was worn on the non-dominant hand. Selection was performed with the index finger of the dominant hand for both the in-air and smartwatch conditions. Targets were only dismissed upon a correct selection.

4.4 Target Acquisition

The *ID* range and values examined in the target acquisition and goal crossing tasks were chosen from within the bounding constraints of the interface size and minimum feasible target size. The target widths (W = 20, 30, 40, and 50 pixels) and target distances (D = 70, 120, 170, 220, and 270 pixels) are expressed in pixels relative to the smartwatch screen (with a resolution of 320×320 pixels) and define 20 unique IDs ranging from 1.26 to 3.86 bits. All 20 W-D conditions were balanced with a balanced Latin square and participants performed 22 trials within each condition.

For a valid selection, participants had to release the cursor within the target (see Figure 4). As releasing the cursor in the in-air condition was a three-dimensional task, early pilot studies showed that selections would often fail due to the cursor leaving the target during a release. Therefore, a distinction between in-air and smartwatch selection was introduced:

- **In-Air selection:** Targets are selected by moving the hand to generate a collision between the index cursor and the target. To select the next target, the hand must first be retracted from the gesture pane before reentering.
- **Smartwatch selection:** Targets are selected by placing the cursor within the target and then lifting the finger from the touch surface. Lifting the finger while the cursor is outside of the target is a selection error even if the target was previously intersected.

We discuss the implications of this distinction in selection behaviour later in Section 6.3.

4.5 Goal Crossing

Goal crossing was identical for both smartwatch and in-air selection. Participants were asked to start their selection in the centre of the interface, cross the target, and release without returning to the centre again (see Figure 5). Missing the target or crossing in the opposite direction (that is, from outside to inside) did not count as a valid selection and had to be repeated again. The index cursor had to remain within the ideal gesture pane during the actual intersection while crossing the goal.

4.6 Circular Steering

The circular steering task is illustrated in Figure 6. The tunnel opening was rotated by 90 degrees between each trial to cover four different starting positions. The necessity of a tolerance at the tunnel opening was determined in early pilot studies. This is a deviation from the original task description in Accot and Zhai [1] due to the uncertainty inherent in in-air and smartwatch-based selection.

For a valid circular steering action, the cursor path must satisfy the following criteria: 1) the first cursor position that lies within the tunnel has to be close to the opening (with a tolerance of 50°); 2) the first cursor position that lies outside of the tunnel after exiting it has to be close to the opening (with a tolerance of 50°); and 3)

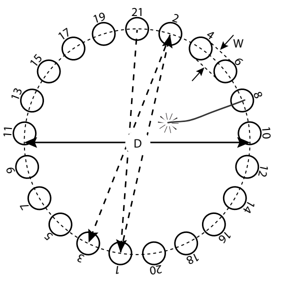


Figure 4: Target acquisition task. Selection can start at any position but must end within the target (smartwatch) or pass through the target (in-air).

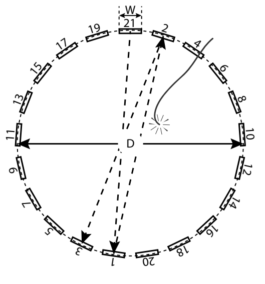


Figure 5: Goal crossing task. Selection has to start within the circle and end after crossing a goal for a valid trial. In-air cursor must stay in the target plane.

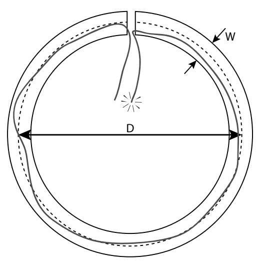


Figure 6: Circular steering task. Selection starts by entering the tunnel opening, steering through the path in a specified direction, then leaving though the same tunnel opening. Entry/exit through the opening has a tolerance of 50°. The circular Index of Difficulty is the quotient of tunnel circumference and tunnel width.

the tunnel has to be steered in the direction indicated by the arrows displayed on the tunnel (clockwise or counter-clockwise).

A different range of index of difficulty values were chosen for the circular steering task due to the different definition of circular index of difficulty, ID_C. According to Equation 4, ID_C is a quotient of tunnel circumference and tunnel width. Therefore, the lowest ID_C that can be defined without the tunnel overlapping with itself is given for $W = 2R$, resulting in an ID_C of π (note that ID_C is not expressed in terms of bits). The tunnel width (W = 30, 40, 50, and 60 pixels) and tunnel circumference (D = 200, 350, 500, 650, and 800 pixels) define 20 unique ID_C values ranging from 3.33 to 26.67. This approximates the range chosen by Accot and Zhai [1] in their original work on circular steering. In our experiment, each D-W condition consisted of only four trials due to the time consuming nature of this particular task.

5 Results

This section summarizes the performance of the in-air and smartwatch techniques across the three examined interface actions.

5.1 Target Acquisition

The average movement time across participants in all IDs was 886 ± 87 ms (one standard deviation) for In-Air and 1275 ± 162 ms for Smartwatch. Repeated measures analysis of variance revealed that the difference was statistically significant ($F_{1,19} = 145.01, \eta_p^2 = 0.884, p < 0.001$). Figure 7a shows average movement time (MT) as a function of Index of Difficulty (ID) across participants. The throughput (TP) was 1.75 bit/s for Smartwatch and 4.17 bit/s for In-Air target acquisition.

5.2 Goal Crossing

The average movement time was 1168 ± 452 ms for In-Air and 1269 ± 250 ms for Smartwatch. A repeated measures analysis of variance did not reveal a significant difference ($F_{1,19} = 1.390, \eta_p^2 = 0.068, p = 0.253$). Figure 7b shows average MT as a function of ID. The throughput was 4.00 bit/s for Smartwatch and 9.09 bit/s for In-Air goal crossing.

5.3 Circular Steering

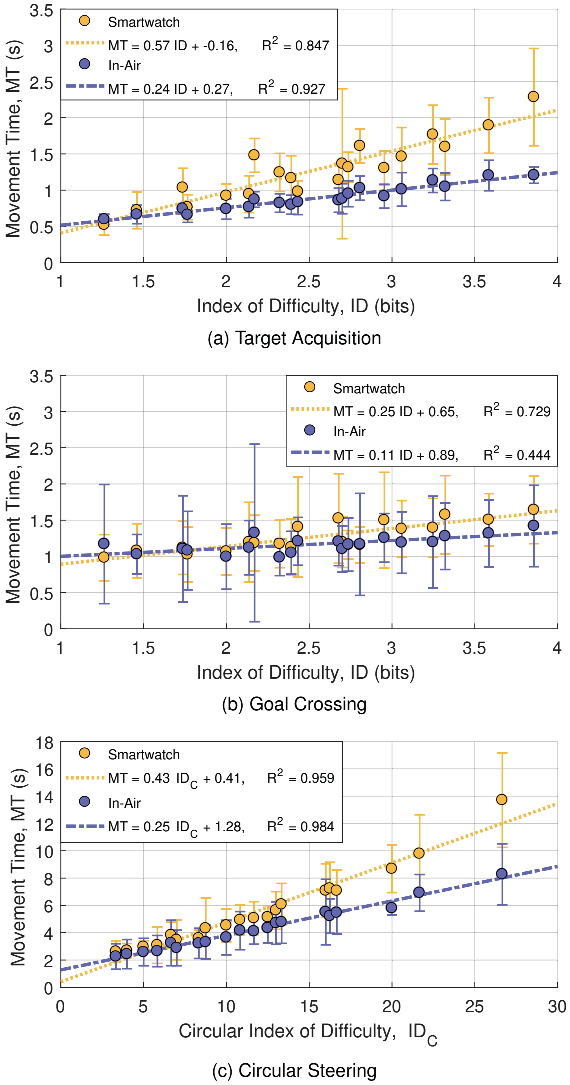
The average movement time was 4276 ± 1217 ms for In-Air and 5575 ± 1088 ms for Smartwatch. A repeated measures analysis of variance revealed that the difference was statistically significant ($F_{1,19} = 44.857, \eta_p^2 = 0.702, p < 0.001$). Figure 7c shows average MT as a function of ID_C. The throughput (TP_C) for Smartwatch circular steering and In-Air circular steering was 2.33 s^{-1} and 4.00 s^{-1} respectively.

5.4 Agreement with Performance Models

Figure 7 shows the linear regression models for target acquisition, goal crossing, and circular steering. Model fits are calculated using the coefficient of determination (R^2), which is the proportion of the variance in movement time explained by index of difficulty or circular index of difficulty. Overall the model fits are high, in particular for the Fitts' law target acquisition task and the circular steering task. The fit is reasonable for Smartwatch goal crossing ($R^2 = 0.729$) but quite poor for In-Air ($R^2 = 0.444$). We conjecture the poor model fit for In-Air goal crossing is due to the difficulty of adjusting the depth and the position of the hand simultaneously. This conjecture is supported by the high intercept value of 0.89 (see Figure 7b).

6 Discussion

The results suggest that In-Air selection is consistently more efficient than Smartwatch. This is also reflected in the significantly faster movement times for In-Air in target acquisition and circular steering.



(a) Target Acquisition

(b) Goal Crossing

(c) Circular Steering

Figure 7: Across-participants, mean movement time MT versus index of difficulty ID. Error bars show ±1 standard deviation.

Interpreted in the context of the previously related studies, the throughputs determined in target acquisition for In-Air (4.17 bit/s) and Smartwatch (1.75 bit/s) are approximately consistent with values reported in [9] (8.05 bit/s on a tabletop display with direct-touch and 4.35 bit/s for mouse-based selection). It is reasonable to expect that In-Air performance would be considerably worse than a physical tabletop alternative given the coarse hand tracking provided by the HoloLens in addition to the less ergonomic alignment of the interface selection plane.

The throughput TP_C in the circular steering task (Smartwatch: 2.33 s^{-1}, In-Air: 4.00 s^{-1}) was also in the range reported by Accot and Zhai [1] where the circular steering was performed with several physical devices (mouse (5.5 s^{-1}), tablet (5.4 s^{-1}), trackpoint (3.7 s^{-1}), trackball (3.0 s^{-1}), touchpad (2.5 s^{-1})).

An interesting result was the comparatively similar performance of the two techniques in the goal crossing task. We observed high variance in the measured movement times and no significant overall difference between the two techniques. The throughput for In-Air was approximately double that for Smartwatch, however, the model fit is relatively poor.

An alternative perspective on the results presented in Figure 7, however, is that at low *ID* values the performance of the two techniques converge. If interface elements can be held in these *ID* ranges, other usability considerations might dominate which technique is most suitable. This finding highlights the value of investigating the performance envelope of a diverse range of interaction techniques over a wide range of *ID* values.

A potential factor contributing to the generally worse performance of SMARTWATCH is the additional cognitive demand imposed by the indirect control of the cursor. The user must learn the mapping between the cursor position and the touch location on the smartwatch screen. The fact that the touchscreen is so small means that the scaling required is comparatively large. Although this scaling remains consistent, the scaling up of small movements may significantly exacerbate errors and negatively affect usability. In the context of this study, however, we sought to remove any potential learning affect associated with the control techniques by providing a familiarisation period. Nevertheless, the potential effect of the additional cognitive demand associated with indirect cursor control cannot be eliminated without extensive use. This could mean that the difference in performance observed between the two techniques may decrease with increasing use.

In summary, in-air selection appears to show a definite performance advantage over smartwatch input. However, while conducting the experiment, we received consistent feedback from participants that the in-air technique was fatiguing and uncomfortable after prolonged use, which indicates that actual AR interfaces should limit the frequency of in-air selections to a comfortable value. In contrast, participants generally found smartwatch input comfortable and relaxing to use.

Although untested specifically in this study, participants also expressed their concerns about social acceptance of in-air interactions. By comparison, the social acceptability (at least in the immediate future) of the smartwatch technique is more favourable. The smartwatch technique aligns closely with three of the four key reasons for liking a gesture as proposed by Rico and Brewster [22]: subtle movement, similar to existing technology, and looks or feels similar to everyday actions.

6.1 Integrating Complementary Selection Actions

The performance envelopes of in-air direct and smartwatch indirect control suggest that the two techniques may be complementary. The user can fluidly transition between techniques since each relies on a different control mechanism. The in-air index cursor is always available to manipulate within-reach interface elements while a hand is inside the tracking volume of the NED. The indirect cursor is activated explicitly by placing a finger on the smartwatch touch surface. A minor complication is the fact that indirect cursor control is bound to a designated plane, i.e. cursor movement is restricted to the currently focused interface.

We now describe two hypothetical scenarios in which the techniques deliver complementary functionality supported by fluid mode transitions. In one scenario, the user selects a button on a distant AR menu using the smartwatch-based indirect cursor. A new interface opens in front of the user which is in comfortable reach. The user chooses to switch to in-air direct cursor control for speed. This transition is seamless as the user need only lift their finger off the smartwatch and place their hand inside the NED's field-of-view to activate the index cursor. Alternatively, the user can continue to use the indirect cursor since the focus has been moved to the new interface. At this point, the optimal selection technique is a user choice and depends on personal preference, current fatigue level, and task complexity.

In another scenario, the user is modifying the visual appearance (for example, colour, scale, orientation) of an object placed in the AR scene by interacting with buttons on a context menu. The user initially opts to use the in-air cursor to quickly and approximately adjust the appearance towards desired settings. At some point, however, the user's focus switches to fine adjustment as they seek to accommodate the spatial context and physical scene's background into the visual aesthetics of the object. Consequently, the user prefers to adjust appearance settings using the smartwatch indirect cursor so that their arm is not occluding or otherwise disrupting their view of the mixed-reality scene. The transition from in-air to smartwatch is smooth as the context menu is already active and the user must simply locate the watch on their wrist. The resting finger position facilitates small gestures and the smartwatch cursor occludes only the widget itself.

6.2 An Illustrative Example of Performance Modelling

We now briefly illustrate the process of exploiting knowledge of the performance envelope for each technique to evaluate alternative interface design decisions. Consider an AR application involving creating and/or placing various virtual objects in the space, such as primitive shapes, text boxes, 3D line drawings. The application designer desires to provide a simple context menu that can be displayed depending on the current object in focus. This menu will allow the user to adjust basic object appearance and perform simple actions such as move and delete. The context menu will thus contain four buttons as shown in Figure 8: *Size*, *Colour*, *Move* and *Delete*.

To minimise scene occlusion, the designer wishes to hide the four radial buttons by default but is happy for the single centre *Edit* button to remain visible above the virtual object placed in the space. The designer wants to estimate the theoretical performance of two alternative interactions with the context menu to check whether they are both worth implementing for physical testing. The two alternative interactions are described briefly below:

- **In-air direct touch:** The context menu will be activated by directly touching the centre *Edit* button. One of the four context menu buttons will then be selected by direct touch with the index cursor.
- **Smartwatch crossing:** The context menu will be activated by an initial touch on the smartwatch screen while visually focused on the target object via a gaze cursor. One of the four context menu buttons will then be selected using the smartwatch by an outward crossing trace.

We define selection time (*ST*) as the total time required to make a selection from the context menu. This can be expressed as

$$ST = MT + AT, \tag{5}$$

where activation time (*AT*) is the time required to open the context menu and movement time (*MT*) is the time required to select one of the four menu buttons.

The designer sizes the centre *Edit* button to be approximately 40 mm in diameter. Based on these dimensions, a rough estimate for

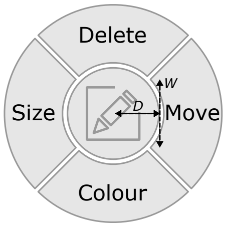


Figure 8: Anticipated layout of a radial context menu for an AR application. The application designer seeks to determine the theoretical performance of two alternative interaction schemes.

activation time using in-air selection is obtained by assuming a hand moving from an initial position by the user's side to the *Edit* button—a movement amplitude of approximately 500 mm. This yields an activation time of approximately 1.17 s based on the derived target acquisition model for in-air selection. The activation time using the smartwatch is simply the time required to place a finger on the touch surface. The designer builds a simple application and measures the activation times in a self-experiment to be 0.75 s in a seated position and 0.72 s in a standing position.

Upon activation, the movement time for each interaction technique can be estimated using the target acquisition model for in-air selection and using the goal crossing model for smartwatch selection. The designer approximates the target width W based on the chord length of the internal edge of a radial button to be 28 mm and the movement distance D to be 20 mm (half of the *Edit* button width). Movement time is then calculated to be 0.46 s and 0.84 s for in-air and smartwatch respectively. The resulting total selection times are thus 1.63 s for in-air and 1.59 s for smartwatch (in a seated position).

Based on these rough estimates, the designer is happy that the difference in performance does not render any of the techniques redundant and unworthy of physical evaluation. Clearly there are many other factors which may influence these results but the approximations provide the designer with some confidence that their intended interaction techniques are at least theoretically valid.

6.3 Limitations and Future Work

Selecting targets in a seated position with controlled lighting conditions could have favoured the in-air selection technique. The interactive NED vision painted in the introduction motivates future research involving an ecological evaluation of these techniques in a walking scenario in the wild. Changing lighting conditions and movement might shift the balance between both techniques and favour the indirect input on a smartwatch due to tactile sensation feedback and a resting, more subtle hand posture.

While no fatigue values were measured to compare individual tasks, higher fatigue was observed during the steering task which could result from the long selection time necessary [12]. Furthermore, the reported fatigue during in-air selection could be the result of the artificial set-up and the high number of trials inherent to a performance evaluation. The physical strain could be less prominent in an actual application. Nevertheless, the reported in-air performance can serve as a baseline to design and evaluate more ergonomic selection techniques. We hypothesise that improved hand tracking would likely increase performance and might also serve to reduce fatigue. This requires further investigation with alternative hand tracking hardware.

Having established the performance envelopes for in-air and smartwatch-based selection and cursor control, it is now possible to design a variety of content generation, annotation and editing interfaces containing user interface widgets leveraging the fundamental user interface actions of target acquisition, goal crossing or circular steering. Future work will deploy these two techniques for use in a practical AR interface task and evaluate their performance. Such an investigation would help to better articulate the relative benefits of the two techniques in a typical usage scenario. The empirically determined models presented in this paper can also provide the basis for simulations and predictive models of human performance for hypothetical interface designs as illustrated in the previous section. In addition, they can be incorporated into objective functions for automatic user interface optimisation methods.

6.4 Implications for Design

The findings of this work have several design implications for AR interfaces. These are summarized below:

- **Interaction Fatigue:** Within-reach interfaces can benefit from the high input speed of direct in-air selection, but suffer from fatigue effects. Thus, the application designer should ideally ensure such interactions are short and sparse, for example selecting a sub-view from a menu or one of a few elements in an interface.

- **Interaction at Distance:** Out-of-reach interfaces and tasks with high complexity and duration can plausibly benefit from the resting hand position and haptic feedback of indirect smartwatch selection. This is desirable for dragging tasks with multiple targets or continuous cursor interactions, such as scrolling down a long list of items. In contrast to more conventional 2D interfaces in which the user position can be readily inferred, the AR interface designer must consider the likely relative position of the user and how this might impact the most appropriate interaction technique.

- **Modal Fluidity:** The complementary strengths of both techniques suggest a combined usage, where mid-air selection is applied to select sub-menus or large widgets and smartwatch selection is applied to perform fine-grained interaction. Since both techniques do not require a hand-held device, the transition between both is smooth and fast. Designers should both support and exploit a high degree of modal fluidity to ensure users can choose input techniques according to their needs.

- **Context Sensitivity:** The preferable input technique can be highly context-specific. When interacting in a working scenario, for example, handling documents on a virtual desktop in the office, performance can be the main concern of users. During similar interactions in a public environment, unobtrusiveness can have a higher priority. Designers can accommodate such context sensitivity through a better awareness of the likely usage scenarios of their application and/or by providing users with a choice in selection of an interaction technique.

7 Conclusions

The experiment presented in this paper is a step towards the creation of non-encumbering interaction techniques for augmented reality applications. Two techniques, direct in-air selection and indirect smartwatch-based selection, were evaluated for three fundamental user interface actions: target acquisition, goal crossing, and circular steering. In-air direct control was significantly faster than smartwatch indirect control in the target acquisition and circular steering actions. The results demonstrate that Fitts' law and the steering law model these two tasks well for these circumstances. The goal crossing performance difference between conditions was marginal and the relatively low R^2 value suggests that the goal crossing model does not accurately model performance for the in-air input method.

Qualitative feedback from participants suggests that user comfort and social acceptance of in-air interaction can influence user preferences for interaction techniques. Such usability concerns may indeed dominate at low *ID* values at which performance levels converge.

This paper establishes the performance envelopes of two complementary interaction techniques, well suited to many AR applications. The performance envelopes in this paper can aid manual performance-conscious design of AR user interfaces and the model parameters can be used to guide user interface optimisation algorithms by incorporating the user interface action models in objective functions. Ultimately, the vision of an interactive NED replacing the smartphone will likely be achieved through a suite of diverse input techniques with various strengths and weaknesses.

Acknowledgments

This work was supported by EPSRC (grant number EP/N010558/1) and the Trimble Fund. Part of this work was conducted within the Transregional Collaborative Research Centre SFB/TRR 62 Companion-Technology of Cognitive Technical Systems funded by the German Research Foundation (DFG).

References

[1] J. Accot and S. Zhai. Performance evaluation of input devices in trajectory-based tasks: an application of the steering law. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '99, pp. 466–472. ACM, 1999.

[2] J. Accot and S. Zhai. More than dotting the i's—foundations for crossing-based interfaces. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pp. 73–80. ACM, 2002.

[3] G. Apitz and F. Guimbretière. CrossY: A Crossing-based Drawing Application. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*, UIST '04, pp. 3–12. ACM, New York, NY, USA, 2004.

[4] H. Benko, E. W. Ishak, and S. Feiner. Collaborative mixed reality visualization of an archaeological excavation. In *Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality*, pp. 132–140. IEEE Computer Society, 2004.

[5] X. Bi, Y. Li, and S. Zhai. FFitts Law: Modeling Finger Touch with Fitts' Law. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pp. 1363–1372. ACM, New York, NY, USA, 2013.

[6] D. Chattopadhyay, K. O'Hara, S. Rintel, and R. Rädle. Office Social: Presentation interactivity for nearby devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '16, pp. 2487–2491. ACM, 2016.

[7] K. Dorfmuller-Ulhaas and D. Schmalstieg. Finger tracking for interaction in augmented environments. In *Proceedings of the IEEE and ACM International Symposium on Augmented Reality*, pp. 55–64, 2001.

[8] A. O. S. Feiner. The flexible pointer: An interaction technique for selection in augmented and virtual reality. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*, UIST'03, pp. 81–82, 2003.

[9] C. Forlines, D. Wigdor, C. Shen, and R. Balakrishnan. Direct-touch vs. mouse input for tabletop displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pp. 647–656. ACM, New York, NY, USA, 2007.

[10] F. Guimbretiére and T. Winograd. Flowmenu: Combining command, text, and data entry. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology*, UIST '00, pp. 213–216. ACM, New York, NY, USA, 2000.

[11] B. Hartmann, M. R. Morris, H. Benko, and A. D. Wilson. Augmenting interactive tables with mice & keyboards. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology*, UIST '09, pp. 149–152. ACM, 2009.

[12] J. D. Hincapié-Ramos, X. Guo, P. Moghadasian, and P. Irani. Consumed endurance: A metric to quantify arm fatigue of mid-air interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pp. 1063–1072. ACM, New York, NY, USA, 2014.

[13] K. Huo, Vinayak, and K. Ramani. Window-shaping: 3D design ideation by creating on, borrowing from, and looking at the physical world. In *Proceedings of the Eleventh International Conference on Tangible, Embedded, and Embodied Interaction*, TEI '17, pp. 37–45. ACM, New York, NY, USA, 2017.

[14] R. J. K. Jacob. What you look at is what you get: Eye movement-based interaction techniques. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '90, pp. 11–18. ACM, New York, NY, USA, 1990.

[15] Y. Luo and D. Vogel. Crossing-based selection with direct touch input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pp. 2627–2636. ACM, New York, NY, USA, 2014.

[16] I. S. MacKenzie. Fitts' law as a research and design tool in human-computer interaction. *Humam-Computer Interaction*, 7(1):91–139, Mar. 1992.

[17] M. Mine, A. Yoganandan, and D. Coffey. Making VR work: Building a real-world immersive modeling application in the virtual world. In *Proceedings of the 2Nd ACM Symposium on Spatial User Interaction*, SUI '14, pp. 80–89. ACM, New York, NY, USA, 2014.

[18] B. A. Myers, H. Stiel, and R. Gargiulo. Collaboration using multiple PDAs connected to a PC. In *Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work*, CSCW '98, pp. 285–294. ACM, New York, NY, USA, 1998.

[19] M. Ohta, S. Nagano, S. Takahashi, H. Abe, and K. Yamashita. Mixed-reality shopping system using HMD and smartwatch. In *Proceedings of the 2015 ACM International Symposium on Wearable Computers*, pp. 125–128. ACM, 2015.

[20] I. Poupyrev, T. Ichikawa, S. Weghorst, and M. Billinghurst. Egocentric object manipulation in virtual environments: Empirical evaluation of interaction techniques. *Computer Graphics Forum*, 17(3):41–52, 1998.

[21] U. Rashid, J. Kauko, J. Häkkilä, and A. Quigley. 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*, pp. 495–498. ACM, 2011.

[22] J. Rico and S. Brewster. Usable gestures for mobile interfaces: Evaluating social acceptability. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pp. 887–896, 2010.

[23] M. Rohs and A. Oulasvirta. Target acquisition with camera phones when used as magic lenses. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pp. 1409–1418. ACM, New York, NY, USA, 2008.

[24] M. Rohs, A. Oulasvirta, and T. Suomalainen. Interaction with magic lenses: Real-world validation of a fitts' law model. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pp. 2725–2728. ACM, New York, NY, USA, 2011.

[25] D. Schmalstieg, A. Fuhrmann, and G. Hesina. Bridging multiple user interface dimensions with augmented reality. In *Proceedings of the IEEE and ACM International Symposium on Augmented Reality*, pp. 20–29, 2000.

[26] S. Yamanaka, W. Stuerzlinger, and H. Miyashita. Steering through sequential linear path segments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '17, pp. 232–243. ACM, New York, NY, USA, 2017.