ARPads: Mid-air Indirect Input for Augmented Reality

Eugenie Brasier, Olivier Chapuis, Nicolas Ferey, Jeanne Vezien, Caroline Appert

To cite this version:

Eugenie Brasier, Olivier Chapuis, Nicolas Ferey, Jeanne Vezien, Caroline Appert. ARPads: Mid-air Indirect Input for Augmented Reality. 2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), Nov 2020, Porto de Galinhas, Brazil. pp.332-343, 10.1109/ISMAR50242.2020.00060 . hal-02915795

HAL Id: hal-02915795
https://hal.science/hal-02915795v1

Submitted on 16 Aug 2020

ARPads: Mid-air Indirect Input for Augmented Reality

Eugenie Brasier* Olivier Chapuis* Nicolas Ferey† Jeanne Vezien† Caroline Appert*

ABSTRACT

Interacting efficiently and comfortably with Augmented Reality (AR) headsets remains a major issue. We investigate the concept of mid-air pads as an alternative to gaze or direct hand input to control a cursor in windows anchored in the environment. *ARPads* allow users to control the cursor displayed in the headset screen through movements on a mid-air plane, which is not spatially aligned with the headset screen. We investigate a design space for *ARPads*, which takes into account the position of the pad relative to the user's body, and the orientation of the pad relative to that of the headset screen. Our study suggests that 1) indirect input can achieve the same performance as direct input while causing less fatigue than hand raycast, 2) an *ARPad* should be attached to the wrist or waist rather than to the thigh, and 3) the *ARPad* and the screen should have the same orientation.

Keywords: AR headset; Indirect input; Small scale gestures

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

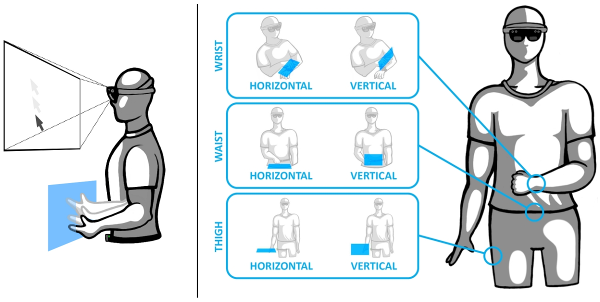


Figure 1: (left) Mid-air indirect input: hand movements on an *ARPad* control the location of the cursor in the AR window (an *ARPad* is an invisible plane in mid-air, which is colored in translucent blue here for illustration purposes). (right) Our design space for *ARPads*: the virtual plane can be located at different positions (wrist, waist or thigh), and can have different orientations (vertical or horizontal).

1 INTRODUCTION

Augmented Reality (AR) headsets make it possible for users to get private, in-situ user interfaces. They can offer personalized exhibits in museums (*e.g.*, [54]), deliver specific instructions in an educational context (*e.g.*, [20]), or display data visualizations related to objects in the environment [66]. Interacting with virtual objects displayed in the headset is typically achieved through gaze and hand gestures performed in front of the headset. For example, the Microsoft Hololens requires users to move their head to adjust the cursor position over the object of interest, and then raise their arm to perform an air-tap hand gesture in the tracking area located ahead of the headset.

Both gaze input and mid-air gestures have drawbacks. Mid-air gestures can quickly get tiring when the arms are in an upward position [5,30], while gaze-based input suffers from precision issues [72] and is prone to accidental selections [33,34]. In the case of AR headsets that estimate gaze based on head position and orientation, the situation is even worse as the object to select needs to be put in the viewport's center, making peripheral (but potentially interesting) objects leave the user's field of view. Furthermore, large gestures performed in mid-air can be socially awkward [1]. So can unnatural head movements. This might result in a mismatch between displayed content that is intended for private consumption and exaggerated movements to interact with this content.

We investigate the concept of *ARPad* as a means to replace or complement the default input channels of AR headsets. An *ARPad* can be seen as a virtual plane in which users perform small scale movements to control the cursor displayed in the headset's screen (Fig. 1-left and Fig. 4), like a trackpad does with a cursor displayed on a laptop's screen.[1] As already demonstrated in the context of remote control of distant vertical screens [43], this indirect way of controlling a cursor enables users to adopt more relaxed postures and perform gestures that are subtle, and thus more acceptable in public [16].

We investigate a *design space* for such indirect input in the context of AR headsets. As illustrated in Fig. 1-right, we consider two dimensions: where the gestures are performed relative to the user's body (wrist, waist or thigh), and what the main orientation of those gestures is (horizontal, vertical). We implement specific designs in this space in an AR headset that has a limited-size viewport (*i.e.*, a Microsoft Hololens 1). We conduct a study to evaluate these designs as well as two direct input baselines (gaze and hand raycast) on both discrete and continuous input tasks. Our study results indicate that: 1) indirect input can achieve the same performance as direct input while causing less fatigue than hand raycast; 2) an *ARPad* is more efficient when positioned close to users' wrist or waist rather than to their thigh; and 3) users perform better with an *ARPad* when its orientation matches that of the headset screen.

2 RELATED WORK

Indirect input is a fundamental concept in HCI that received a lot of attention in the literature about pointing. We only discuss here on works that are closely related to our contribution, focusing on interaction with head-mounted displays and mid-air gestures for interacting with portable devices.

2.1 Interacting with Head-mounted Displays

Usability is a key aspect to consider if we want to facilitate the adoption of AR glasses [39]. Many projects in the literature aim at improving interaction with such devices. As mentioned above, interaction with AR headsets usually relies on gaze input. While fast, gaze input suffers from precision issues [72]. Gaze input thus benefits from the addition of other modalities [40] such as a mouse in a desktop setting [76] or a handheld touch device in a mobile

---

*Université Paris-Saclay, CNRS, Inria, LRI, Orsay France. Email: [brasier | chapuis | appert]@lri.fr

†Université Paris-Saclay, CNRS, LIMSI, Orsay France. Email: [ferey | vezien]@limsi.fr

[1] The parallel is not to be considered literally as a trackpad enables relative cursor control (with clutching actions) while our implementation of an *ARPad* enables absolute control (as a graphics tablet).

setting [69]. Gaze input also suffers from the Midas Touch problem [33, 34], which can cause accidental activation commands in the absence of a proper selection mechanism. Selection with gaze is usually performed using a dwell action. It can also rely on an additional device [33, 72]. When targets are moving, selection can also be achieved with eye pursuit [37, 60].

Mid-air gestures have also received a lot of attention as a means to interact with head-mounted displays. They remove the need for additional devices and allow users to perform *direct manipulation* of objects that are within arm's reach [46]. Mid-air gestures can be complemented with another modality such as speech [59] or gaze [56]. Commercial solutions like the Microsoft Hololens rely on gaze for pointing and air tap gestures for selection. In this specific context of AR headsets, optical sensors for tracking hand gestures are usually mounted on the headset, looking in the direction of users' sight (*e.g.*, [12, 26]). This means that users have to perform gestures at a relatively high height, which quickly causes discomfort and fatigue [30, 57].

When users do not need to keep their hands free, an external device can be used as a controller for Head-mounted Displays. This external device can be a smartwatch [58], a smartphone [8], a tablet [70], or a custom one [64]. The tangibility of an external device usually improves comfort and precision. For instance, a smartphone can act as a trackpad for pointing content displayed on a distant screen [68] or in AR [11]. A spatially tracked mobile device can control 6 DOF for manipulating spatial content [45, 48]. Further adding multitouch input, users can even perform advanced manipulations in VR [70]. Conversely, while portable devices can address weaknesses of HMDs in terms of input, HMDs can also address weaknesses of portable devices in terms of output [22, 77]. For instance, an AR display can enlarge a smartphone's display to accommodate a large information space [53] or to distribute widgets across displays [21].

For tasks performed in the office, users can lay a tablet on their desk and use its soft keyboard for text entry in VR [23]. They can also use a mouse [20] or a physical keyboard [23]. In a mobile context, a smartwatch can be used to interact with remote content: for raycast pointing [58], for indirect pointing [74, 77] or for text input [2]. All these projects advocate using an external input device to interact with HMDs. We argue that bare-hand input can complement such techniques or replace them when an external device is not available. Bare-hand input also has the advantage of making HMDs stand-alone devices, and of leaving users' hands free for other actions.

A hands-free alternative consists in using the user's body as an input device with large-amplitude gestures performed on the whole body [18] or with finger gestures performed on specific parts of the body (*e.g.*, on the forearm [29], on the hand palm [50], on the abdomen [71], and even on the face [42, 65, 75]). Wearable devices also support portability. Dobbelstein *et al.* have proposed wearables for interacting with headset screens with, *e.g.*, a touch-enabled belt [13] and pocket [14]. Finally, foot taps have also been investigated recently as a means to perform menu selection with head-mounted displays [51]. However, these latter solutions require custom sensors, have limited input capabilities and might raise questions regarding social acceptance.

2.2 Overcoming Fatigue of Mid-air Gestures

In light of the above, mid-air gestures remain a very good candidate for interacting with headset screens. However, tracking such gestures in a portable setting is not trivial. Some research prototypes embed cameras in clothes, shoes or jewels. For example, the Gesture Pendant [19] is a wearable device used to control home environments with mid-air gestures. SixthSense [47] extends the Gesture Pendant by combining a projector and a camera in a pendant. It can project digital content in the physical environment in front of the user, and capture gestures that are performed at chest height. ShoeSense [6] augments a shoe with a Leap Motion to track large-amplitude arm and hand gestures to control wearable applications. Such wearable tracking equipment has the advantage of allowing users to perform mid-air gestures with arms in a position lower than when optical sensors are mounted on the headset itself.

OwnerShift [17] is a recent technique for Virtual Reality (VR) headsets that enables users to progressively relax their posture during long-lasting interactions. Two optical sensors are mounted on the headset itself, one oriented along users' sight, the other oriented downward. Users see a virtual representation of their hands, which gets progressively shifted up so as to encourage users to shift their real hands down, smoothly transitioning from overhead to waist. OwnerShift builds upon the set of techniques that play with the rubber hand illusion [7]. For example, the Go-go Interaction Technique virtually extends users' arms to reach distant objects without walking [62]. The converse approach also works: instead of virtually extending users' body, virtual objects can be brought closer in motor space while preserving their positions in visual space [49]. However, AR gives less freedom than VR as it makes it challenging to modify the representation of users' body.

Gunslinger [43] addresses the fatigue problem of mid-air gestures in a more radical manner. Relying on what has been later called *at-your-side gestures* [67] and earlier used for zooming actions in [52], Gunslinger allows users to remote control a large display using small-scale movements that are performed in a relaxed posture. One Leap Motion controller is attached to each thigh, tracking both hands when arms are along the body. Users point with their dominant hand following a trackpad metaphor, and activate commands with specific postures of their non-dominant hand. Gunslinger's support for pointing is very similar to a horizontal *ARPad* attached to users' thigh in our design space, with one notable difference: Gunslinger works in relative mode (with clutching actions) while *ARPads* work in absolute mode.

2.3 Extending Input Space in the Air

Extending the interaction space in the air is a strategy that has been implemented for devices other than head-mounted ones. It was initially investigated for small devices because of the limited size of their interaction surface, which causes occlusion and precision issues. For example, the Gesture Watch [38] allows users to perform mid-air gestures above the watch. Gestures are performed with the dominant hand over the watch worn on the non-dominant arm (the watch is equipped with an array of proximity sensors). The Abracadabra wristwatch [28] extends a watch's sensing area above the surface in the same way, but differs in the sensing technology: the finger from the dominant hand wears a magnet, making it trackable by the watch's magnetometers. Here again, those techniques share some similarities with *ARPads* when they are located on users' wrist with a horizontal orientation.

Building upon the concept of *continuous interaction space*, that considers a touch surface and the space around it as a continuum [44], Chen *et al.* equip a smartphone with a depth camera in order to extend the vocabulary of possible commands with air+touch gestures [10]. The combination of touch and mid-air input has also proven efficient for performing large amplitude slide gestures on a smartphone [4] or on a smartwatch [27].

3 ARPAD DESIGN SPACE

Mid-air indirect input, as a concept, is not novel. But all possible designs have not been systematically investigated. This section describes our *design space* for *ARPads*. This space encompasses techniques from the literature, as well as novel ones that build upon prior work on mid-air gestures for portable devices.

An *ARPad* is an imaginary pad on which users can move their hands to control the position of a cursor displayed in the headset

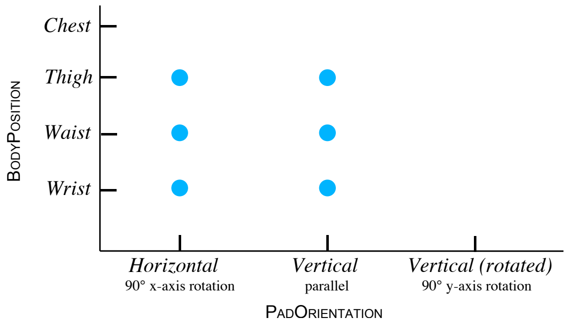


Figure 2: Our design space is structured along two dimensions: where the *ARPad* is located relative to the user's body (BodyPosition), and how the *ARPad* is oriented (PadOrientation). Blue points indicate the six specific configurations that we study.

screen. Like imaginary interfaces [25], users operate *ARPads* with their bare hands and without any visual support. However, *ARPads* differ from the concept envisioned by Gustafson *et al.* where imaginary *interfaces* are operated with direct hand gestures. An *ARPad* is rather an imaginary *device*, which is used for indirect control of a distant screen. The number of possibilities for implementing such a mid-air pad is large. Our design space, illustrated in Fig. 2, focuses on two dimensions that relate to the positioning of *ARPads*. PadOrientation describes the *ARPad*'s position relative to the AR screen as it is key to the definition of indirection. BodyPosition describes the *ARPad*'s position relative to the user's body as ergonomic aspects strongly depend on it.

The first dimension, BodyPosition, is directly motivated by previous work on mid-air gestures for mobile contexts (discussed in the previous section). Tracking mid-air gestures while leaving users free to move in their environment requires attaching optical sensors to them. Depending on which body part the sensors are attached to, users will be able to perform gestures at different locations relative to their body. In our design space, dimension BodyPosition represents the body part close to which users perform mid-air gestures. Here, we specifically consider three body parts: *Thigh*, *Waist* and *Wrist* (Fig. 3). *Thigh* is an obvious candidate as it has already been used for cursor control in the Gunslinger project [43]. Although not focusing on indirect pointing, mid-air gestures relative to users' *Wrist* have also been considered for improving interaction with smartwatches (*e.g.*, [28, 38]). In projects where mid-air gestures were tracked by a pendant [19, 47], users were performing gestures in front of their *Chest*. However, considering empirical results about the fatigue of mid-air gestures when arms are held high in the air [30], *Waist* is a more promising option to study. When gestures are tracked at waist level, users can keep their arms low and should experience less fatigue.

The second dimension, PadOrientation, describes the orientation of the 2D plane within which the pad stands. We consider *Vertical* pads, which correspond to cases where the *ARPad*'s orientation matches that of the headset screen when users look straight ahead. We also consider *Horizontal* pads, which correspond to cases where the *ARPad* is rotated 90° around the x-axis of the headset screen. This configuration is particularly interesting, as we can expect it to be familiar to users who have experience with a physical trackpad on a laptop computer. While we limit our first investigation to these two specific values, other values might be worth investigating. For example, for an *ARPad* located close to users' *Thigh*, we could also consider the case where the *ARPad* is rotated 90° around the y-axis of the headset screen in order to allow for movements parallel to the sagittal plane that might be comfortable. We could even go beyond 2D planes and consider curved surfaces that might better fit the trajectory of limbs around joints.

Figure 3: *Waist*, *Wrist* and *Thigh* body positions for the Leap Motion

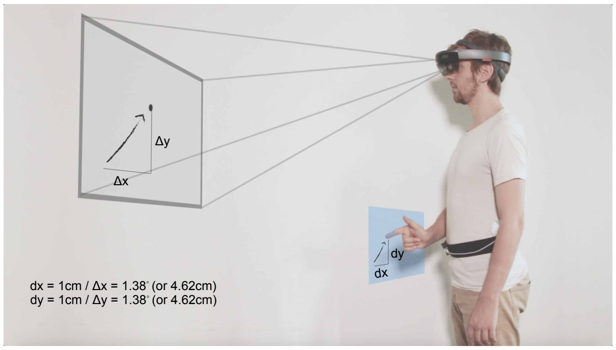


Figure 4: Control display ratio for controlling a cursor displayed in the headset screen through movements on the *ARPad*.

Fig. 5-left shows where the six *ARPad* designs that we investigate stand in our design space. These designs result from crossing values {*Thigh*, *Waist*, *Wrist*} for BodyPosition with values {*Horizontal*, *Vertical*} for PadOrientation.

4 ARPAD PROTOTYPES

We implemented the six *ARPads* identified above using a Microsoft HoloLens and a Leap Motion controller. We relied on Unity (2017.2.4.f1) and the MixedRealityToolkit (2017.4.3.0) to implement our own AR cursor. We reused the `IPointingSource` interface from MRTK, and registered the cursor in the `FocusManager` instance as the main one. The pointer position gets updated each frame depending on what has been sensed with the Leap Motion (without being affected by head motion). Object picking in the AR scene is achieved through raycasting from this pointer.

We rely on a Leap Motion controller for tracking users' hands. Depending on the pad's orientation (BodyPosition), we attach the Leap Motion to users with pieces of velcro tape, as illustrated in Fig. 3. As opposed to Gunslinger [43], which implements relative control with clutching postures, *ARPads* work in absolute mode. We made this choice in order to focus on motor and cognitive aspects involved in indirect control for AR, eliminating any potential issue related to a recognition engine for hand postures. Dimensions of an *ARPad* are 10.3 × 18.3 cm. The LeapMotion provides a 3D coordinate system whose origin is the device itself. We transform coordinates from this cuboid to the *ARPad*'s plane by ignoring the axis that is orthogonal to that plane. This means that an *ARPad* is a cuboid inside which all movements are projected on a plane rather than an actual plane on which users would have to perform unnatural co-planar movements. Informed by pilot studies, we implemented a control display ratio of 1cm:1.38°, which is uniform on both the x- and y-axes (Fig. 4) . This means that a 1cm hand movement will make the cursor move 1.38° in visual angle.

We use an external computer for communication between the Leap Motion and the HoloLens. As the Leap Motion cannot be directly plugged to the HoloLens, it sends data to the computer via USB, which in turn sends them to the HoloLens via sockets.

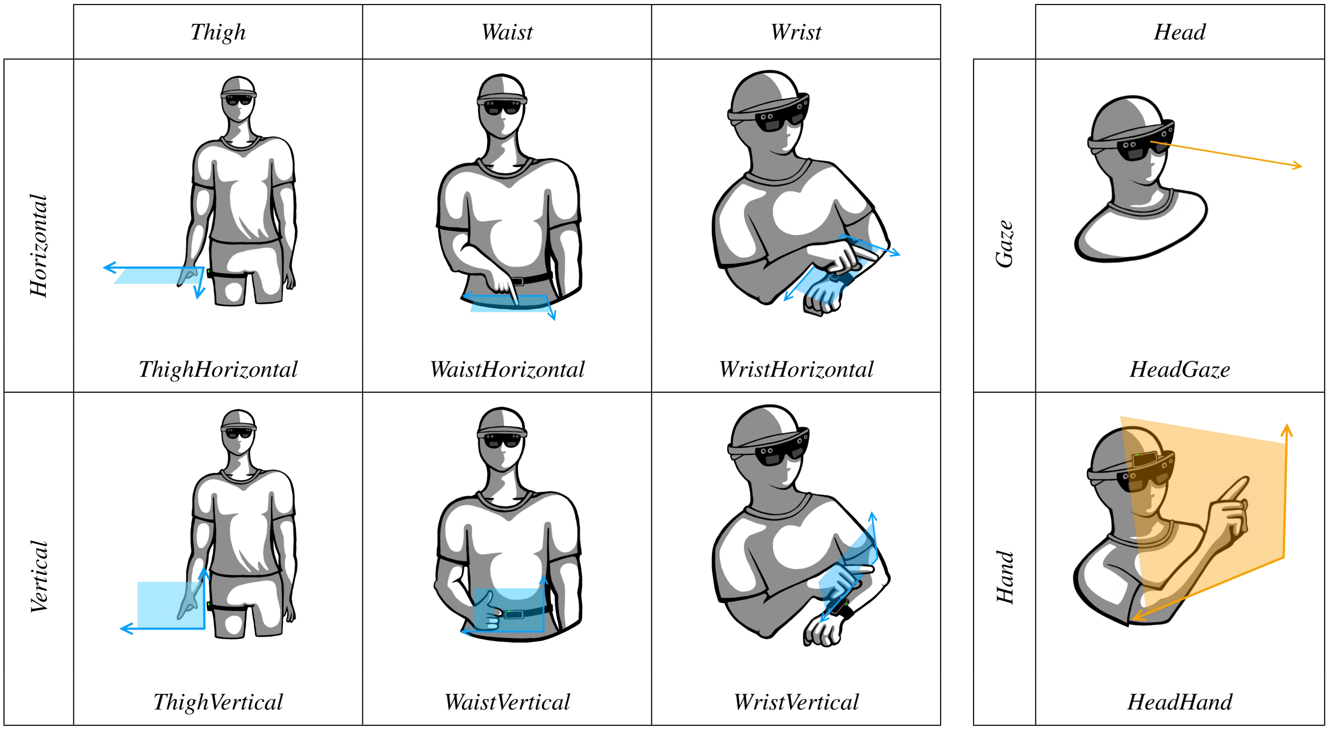


Figure 5: (Left) Indirect input conditions: the six *ARPad* specific configurations from our design space {*Thigh*, *Waist*, *Wrist*} × {*Horizontal*, *Vertical*}. (Right) Direct input conditions: *Gaze* and *Hand* raycast.

5 EXPERIMENT

Our study aims at assessing 1) how indirect input performs in comparison with direct input, and 2) what *ARPad* designs perform best.

5.1 Direct Input Baselines

HeadGaze – An AR headset cursor is usually controlled through gaze, allowing users to directly look at the object they want to select. In current headsets such as the Microsoft HoloLens, gaze orientation is actually estimated based on head orientation. Our first baseline (*HeadGaze*) corresponds to this default cursor control technique.

HeadHand – In our second baseline, users directly control the cursor with their hand. The cursor position results from a raycast along the vector headset-hand. As opposed to indirect techniques, *HeadHand* implements a 1:1 control display ratio to make the cursor position match that of the hand. This corresponds to the *Image-Plane* technique in [35].

5.2 Hypotheses

Our experiment compares the eight techniques illustrated in Fig. 5. Our hypotheses regarding their comparative performance were as follows:

H_1 *Users experience more fatigue with HeadHand than with ARPads*. This first hypothesis is the actual motivation for using indirect input as an alternative to hand direct input. Freehand interaction when holding arms up in the air are tiring [30]. *ARPads* let users keep their arms in positions that are more comfortable than *HeadHand* does.

H_2 *Users perform better with direct techniques than with ARPads*. Visual and motor spaces are not spatially aligned with *ARPads*. Such an indirection can degrade user performance, as is the case for a typing task on a software keyboard in Virtual Reality [23]. We expect that this indirection will also be cognitively demanding when pointing with an *ARPad*, and that it may affect user performance.

H_3 *Vertical ARPads perform better than horizontal ones*. The level of indirection between the motor space and the visual space is even higher when both spaces differ in their orientation. This might degrade user performance even more.

H_4 *An ARPad at Wrist position performs better than at Waist or Thigh positions*. When the *ARPad* is located on the *Wrist*, users can move their non-dominant hand to adjust the plane's location and orientation. This enables bimanual actions where the non-dominant hand sets the context for precise actions performed with the dominant one. Such an asymmetric division of labor might prove efficient [24].

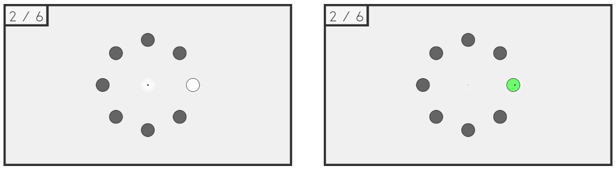
5.3 Participants

16 participants (11 men and 5 women) volunteered for the experiment. All of them were right-handed. Our experiment started with a question about their prior experience with AR and VR headsets. Twelve participants reported that they *never to rarely* used AR headsets, and the other four answered *sometimes to daily*. Eleven participants answered *never to rarely* for VR headsets, and five of them *sometimes to daily*.

5.4 Apparatus

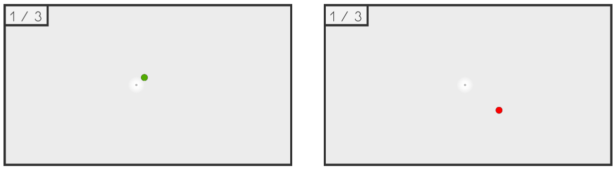
Our experiment ran on a Microsoft HoloLens 1 (30° × 17.5° screen) and a Leap Motion controller (running Orion 4.0.0+52238). Communication between the Leap Motion and the HoloLens was enabled by a Microsoft Surface Book 2 through WiFi / UDP.

The experiment scene was rendered in an AR vertical window of 47.5 × 84.5 cm. It was positioned at a distance of 194cm from the headset when starting an experiment session (*i.e.*, its visual size is 24.57° × 13.96°). The window then remained fixed in space if users moved their head. The control display ratio for *ARPads* in this specific case is thus 1cm:4.62cm. A 1cm hand movement on an *ARPad* makes the cursor move 1.38° in visual angle, which is 4.62cm in the window (Fig. 4).



(a) (b)

Figure 6: The pointing task. (a) The target to acquire is colored white. (b) It turns green when the cursor enters it. Users have to keep the cursor inside it for 500ms to make it disappear.



(a) (b)

Figure 7: The pursuit task. (a) The cursor is green when it is close to the target. (b) As it is getting further from the target, it turns reddish.

A Leap Motion controller has some limitations regarding the range within which it can accurately track users' hands. In particular, the hand must not be too close to the device (*i.e.*, distance should be at least 8cm). We made users aware of this limit by providing some feedback. When users' hand is getting close to the limit, the border of the AR window progressively turns red, indicating that they should adjust their hand position. The operator instructed the participants about how to interpret this visual feedback at the beginning of the experiment.[2]

5.5 Tasks

As in several recent studies (*e.g.*, [17, 40, 51]), our tasks focus on 2D interaction. This does not limit *ARPads* to 2D interactions, however. An *ARPad* can also be used for interacting with 3D content same as mouse input can control 3D software on desktop workstations.

Participants had to perform two types of task with each technique. The first type of task aims at operationalizing discrete interactions. It consists in a typical pointing task implemented according to the ISO 9241-9 standard [32]. The second type of task aims at rather operationalizing continuous interactions that typically occur during direct manipulation of objects or during gesture-based input. Inspired by the study conducted to evaluate OwnerShift [17], we operationalize such continuous tasks with the pursuit tracking task introduced by Poulton [61].

5.5.1 Task 1: Discrete, point-based interactions

Participants perform pointing tasks in diverse directions by acquiring a series of targets that are laid out in a circular manner. As illustrated in Fig. 6, participants acquire eight targets successively. The next target to acquire is colored white. It turns green as soon as the cursor enters it. The participant then has to keep the cursor inside the target for 500ms (*dwell*) to make it disappear. This makes the next target to acquire turn white. We chose to rely on dwell actions for target selection so as not to involve any additional input device or gesture recognition engine that could have introduced some noise.

All targets in a series have the same size. Some series feature *Large* targets (2.36°), *Medium* targets (1.18°) or *Small* targets (0.59°). Targets are laid on a 7.78°-diameter circle.[3] The difficulty of the pointing task is thus an inverse function of target size.

For each elementary pointing task, we measure acquisition time (*i.e.*, the time interval between the last target's disappearance and the time at which the current target is selected), as well as the number of errors (*i.e.*, how many times the cursor leaves the target before it gets selected). As the cursor is located at the center of the scene at the beginning of a series, its distance to the first target is smaller than the diameter of the ring on which targets are laid out. We thus ignore the first target acquisition of all series.

5.5.2 Task 2: Continuous, trajectory-based interactions

Participants have to follow a circular target (0.59° in diameter) that moves in a quasi-random manner within a square of $11.66° \times 11.66°$ (Fig. 7). Participants are instructed to follow this target in order to keep the cursor as close as possible to it. We limited the duration of pursuit tasks to 30 seconds. The target trajectory seemed random to participants, but was actually pre-computed in order to balance the difficulty across participants and techniques. In our experiment design, users had to perform a series of three pursuit tasks with each of the eight techniques. We thus generated three target paths in advance using a sum of four sinusoids for both the x- and -y directions with parameter values reported in [17]. We used the same three paths for each participant × technique block.

For each pursuit task, we consider the mean angular distance between the target and the cursor (we incrementally compute this value at each frame).

5.6 Design and Procedure

We follow a within-subject design with the primary factor TECHNIQUE. Fig. 5 gives an overview of the eight TECHNIQUE conditions. Each indirect technique (*i.e.*, *ARPad*) is a combination of the two secondary factors BODYPOSITION and PADORIENTATION, which correspond to the respective dimensions in our design space. For the two direct techniques, we consider the value of BODYPOSITION to be *Head* but the other secondary factor rather corresponds to the modality: *Gaze* or *Hand*.

Fig. 8 illustrates the presentation order of trials in our experiment. Each participant completed four blocks, one per BODYPOSITION. The presentation order for these blocks was counterbalanced with a Latin Square. There were two subblocks per BODYPOSITION block, one per PADORIENTATION (for BODYPOSITION=*Head*, there were one *Gaze* and one *Hand* subblock). Presentation order between the couple of subblocks was also counterbalanced across participants

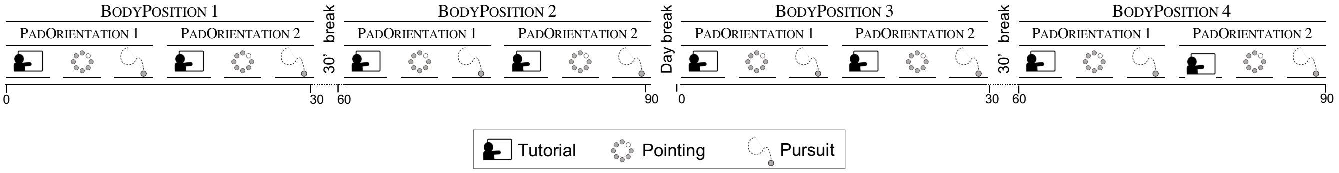


Figure 8: Experiment timeline

---

[2] Such "Out of range" events occurred rarely in our experiment. For Indirect Techniques, it happened: (i) in 3% of pointing trials; (ii) between 0.05% and 1.8% of total time per pursuit trial.

[3] One visual degree corresponds to 3.39cm in the AR vertical window displayed at 194 *cm* from the headset, and to 1.73cm in motor space for indirect techniques. Our conditions thus correspond to 8 *cm*-, 4 *cm*- or 2 *cm*-targets laid out on a 26.5 *cm*-diameter circle in AR window, and to 1.73 *cm*-, 0.87 *cm*- or 0.43 *cm*-targets on a 5.74 *cm*-diameter circle in motor space for indirect techniques.

and body positions. Finally, within each subblock, the presentation order of tasks was always the same: a tutorial session, followed by all pointing tasks and then all pursuit tasks.

In the tutorial session of a block, the operator quickly introduced the technique to be tested. When the technique was an *ARPad*, the operator used a cardboard sheet to illustrate where the pad stood and what its orientation was. Participants then had to point at eight targets, one at each corner of the screen and one at the middle of each side. If needed, participants were free to perform this training session again at will, before actually starting the block. Participants then completed 2×3 pointing series (one series actually consists of eight elementary pointing tasks as detailed above). Pointing series were grouped by three, which were always presented in an increasing order of difficulty: *Large*, *Medium*, and *Small* targets. The first group of three is considered practice, and the other one is used for analysis. Finally, participants performed three pursuit tasks as described above. Here, the first one is considered practice while the other two are for analysis.

We took particular care not to have a transfer of fatigue from one technique to the other. As Fig. 8 illustrates, our experiment was divided into two sessions of 1h30mn each, performed on consecutive days. Each session was also divided into two sub-sessions of 30 minutes, separated by a 30-minute break to let participants rest. They could rest longer if they wanted to. In all cases, they started a BodyPosition block only after having self-assessed themselves as not tired. We counterbalanced presentation order for the two PadOrientation conditions within a block so as not to have always the same transfer direction from one orientation to the other.

Participants had to answer several questionnaires during the course of the experiment. The first one had to be filled in at the very beginning of the procedure. It consisted of the consent form and two questions about their prior experience with head-mounted displays. Then, at the end of each BodyPosition condition, they had to rate the mental demand, the physical demand and the perceived performance of the technique for each of the two pad orientations (using definitions and associated 20-point Likert scales from NASA-TLX). Collecting participants' comparative feedback between techniques is somewhat conflicting with the aim of minimizing transfer of fatigue between conditions. As sessions were run across two separate days, participants might have had difficulties recalling conditions that they tested on the first day. To help participants remember each technique, conditions were illustrated with the pictograms used in Fig. 5. They were displayed in each instruction message all along the experiment, and were printed in the questionnaires.

5.7 Results

We use repeated measures ANOVAs and post-hoc paired t-tests for quantitative analyses (with aggregated data, after having checked that there is no strong violation of normality). We use ART ANOVAs and post-hoc paired Wilcoxon signed-rank tests for qualitative analyses. To decrease the probability of Type I errors, we apply Bonferroni-Holm corrections. Our graphs represent data distribution using notched box plots (notches show the 95% confidence interval for the median). We use R [63] to run our analyses. ANOVAs rely on the `ez` [41] and `ARTool` [36] packages. Post-hoc tests rely on functions `t.test` and `wilcoxsign_test` of the `coin` [31] package, and the `p.adjust` function. Charts rely on the `ggplot2` library [73] for box plots and the `likert` package for Likert graphs.

We first analyze data collected in pointing tasks, then in pursuit tasks. For both tasks, we analyze the effect of Technique to get a comparison between the different *ARPads* and with the direct baselines (*i.e.*, *Head* condition). We then analyze the effect of PadOrientation and BodyPosition for indirect techniques only in order to understand how each of the two dimensions of the design space contributes to their performance. These results help to interpret differences between indirect techniques. Such analyses for indirect techniques only consider a subset of data, as PadOrientation is not crossed with BodyPosition in the *Head* condition. Our experimental design ensures a balanced number of measures across the different groups that we compare in all cases. However, when we analyze data for indirect techniques only, results might have been slightly impacted by the presentation order as the counterbalancing strategy applies to the whole set of technique conditions. We evaluate this impact as low, as the global Latin Square still ensures a balanced presentation order between pairs of indirect techniques. The only impact might come from the presentation of baseline direct techniques in-between two indirect conditions.

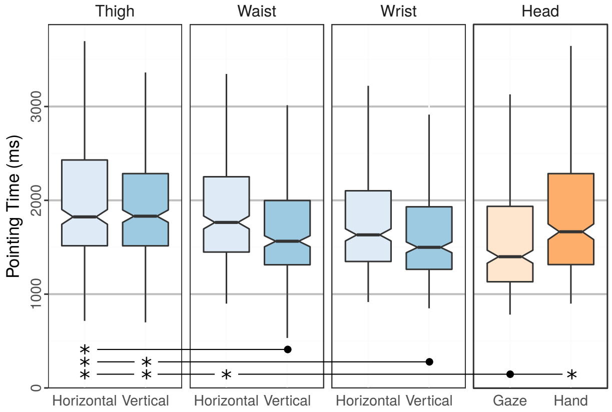


Figure 9: Notched box plot for the pointing time per Technique, grouped by BodyPosition, then by PadOrientation for indirect techniques or modality for direct techniques. Pointing time includes dwell time for selection.

Finally, we analyze participants' feedback through questionnaires.

5.7.1 Pointing Task

Direct and Indirect Techniques (*Time*). Fig. 9 shows pointing time (*Time*) for each Technique, and Fig. 10 shows a breakdown of these data per TargetWidth. An ANOVA for model *Time* ∼ Technique × TargetWidth reveals a significant effect of Technique ($F_{7,105} = 9.02$, $p < 0.001$, $\eta^2_G = 0.16$), and of TargetWidth ($F_{2,30} = 9340$, $p < 0.001$, $\eta^2_G = 0.69$) on *Time*. Moreover, the ANOVA reveals a significant Technique × TargetWidth interaction effect ($F_{14,210} = 3.28$, $p < 0.001$, $\eta^2_G = 0.06$).

Unsurprisingly, TargetWidth has a significant effect on *Time*. The larger the target, the easier the task, as reflected by the shorter pointing time.

HeadGaze has the best completion *Time* on average. It is significantly faster than the two *Thigh* techniques (p's < 0.001), than *WaistHorizontal* ($p = 0.007$), and than *HeadHand* ($p < 0.001$). We then look at the interaction with TargetWidth. For large and medium targets, we have the same results (with similar p's). However, for small targets, *HeadGaze* is only significantly faster than *ThighHorizontal* ($p = 0.017$) and *HeadHand* ($p = 0.033$). This suggests that *HeadGaze* suffers from an increase in pointing difficulty, which is in line with previous work that discusses the precision issues users face with gaze input [33, 34, 72].

WristVertical ranks second regarding average completion *Time*. *WristVertical* is significantly faster than the two *Thigh* techniques (p's $= 0.010$). Looking closer at the interaction with TargetWidth, we can see that, for small targets, *WristVertical* becomes significantly faster than *HeadHand* ($p = 0.021$). Here again, we can see that users have relatively more trouble with direct techniques than with indirect techniques for tasks that require more precision. As *HeadHand* is implemented based on a raycast along the headset-hand vector, it might also have suffered from involuntary head movements.

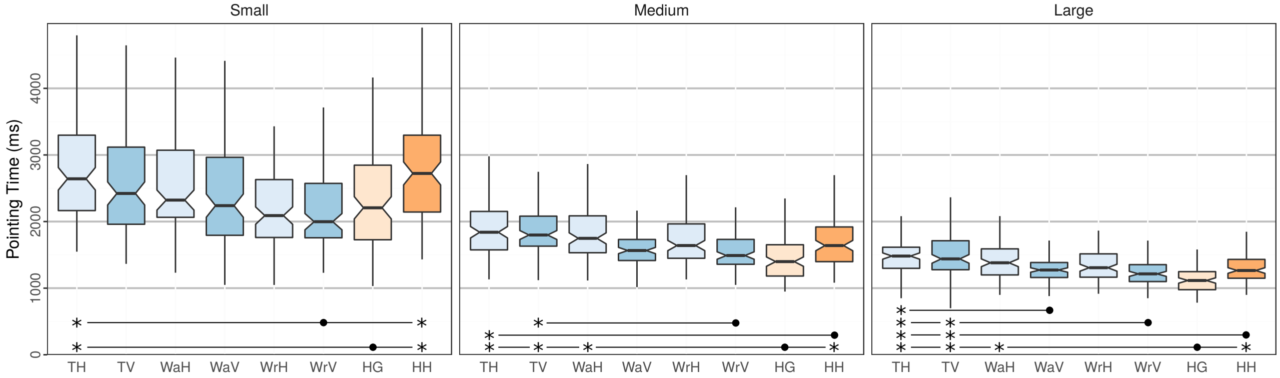


Figure 10: Notched box plots for the pointing time per TECHNIQUE × TARGETWIDTH condition.

TH: *ThighHorizontal*, TV: *ThighVertical*, WaH: *WaistHorizontal*, WaV: *WaistVertical*, WrH: *WristHorizontal*, WrV: *WristVertical*, HG: *HeadGaze*, HH: *HeadHand*

WaistVertical ranks third regarding average completion *Time*. But it is significantly faster than *ThighHorizontal* only ($p{=}0.013$). Furthermore, *WaistVertical*'s advantage is for large targets only ($p{=}0.008$). Tests do not reveal any significant difference for medium and small.

Overall, these results partially support H_2, which states that direct techniques outperform indirect ones. *HeadGaze* actually performs very well, but *HeadHand* does not perform well. However, the *ARPad* that has the best average completion time (*Wrist* × *Vertical*) is not significantly slower than *HeadGaze*. This suggests that indirect input can achieve performance close to that of direct input with gaze.

Indirect Techniques only (*Time*). Considering data for indirect techniques in isolation, we can analyze the effect of PADORIENTATION with an ANOVA according to model: *Time* ∼ BODYPOSITION × PADORIENTATION × TARGETWIDTH. We observe the same significant effect of TARGETWIDTH ($F_{2,30}{=}279$, $p{<}0.001$, $\eta^2_G{=}0.65$), but we also observe a significant effect of both BODYPOSITION ($F_{2,30}{=}12.6$, $p{<}0.001$, $\eta^2_G{=}0.10$) and PADORIENTATION ($F_{1,15}{=}10.1$, $p{=}0.003$, $\eta^2_G{=}0.04$). We also have one significant interaction effect: BODYPOSITION × TARGETWIDTH ($F_{4,60}{=}4.02$, $p{=}0.003$, $\eta^2_G{=}0.03$).

Our study suggests that participants are significantly faster with *Vertical ARPads* than with *Horizontal* ones (H_3). Moreover, t-tests (corrected $n = 3$) reveal a strict order of performance between the three body positions: *Wrist* outperforms *Waist* ($p{=}0.04$), which in turn outperforms *Thigh* ($p{=}0.04$). Regarding the BODYPOSITION × TARGETWIDTH interaction effect, t-tests (correction $n = 9$) show that the superiority of *Wrist* over *Waist* is significant for small targets only. But, overall, ***ARPads* seem to perform best when they are positioned at users' wrist** (H_4).

Direct and Indirect Techniques (*Precision*). To select a target, participants had to keep the cursor for at least 500ms within the target's boundaries (dwell). We counted a *TargetLeave* (TL) event each time the cursor went out of the target, before successfully selecting it. The number of such events provides an indicator of users' precision with the different techniques, as it increases with the difficulty of stabilizing the cursor.

For this analysis, we consider the full set of collected data (direct and indirect techniques). With averages of 1.71 and 1.36 respectively, *HeadGaze* and *HeadHand* have a significantly higher number of TL events than indirect techniques (p's<0.001), which have an average of TL=0.58 by elementary pointing task. Vertical *ARPads* at the *Thigh* position have a slightly higher number of such events (0.74) than other *ARPads*, and actually significantly more than *ARPads* at the *Wrist* location (p's$=0.013$), which have an average of TL=0.46.

This suggests that **the two direct techniques, *HeadGaze* and *HeadHand*, suffer from precision issues**, which probably explains why their comparative performance decreases with small targets.

Fitts' Law. Fig. 11 shows how collected data fit Fitts' law for each technique. We can observe that the point for small targets is systematically above the linear regression line, and even more

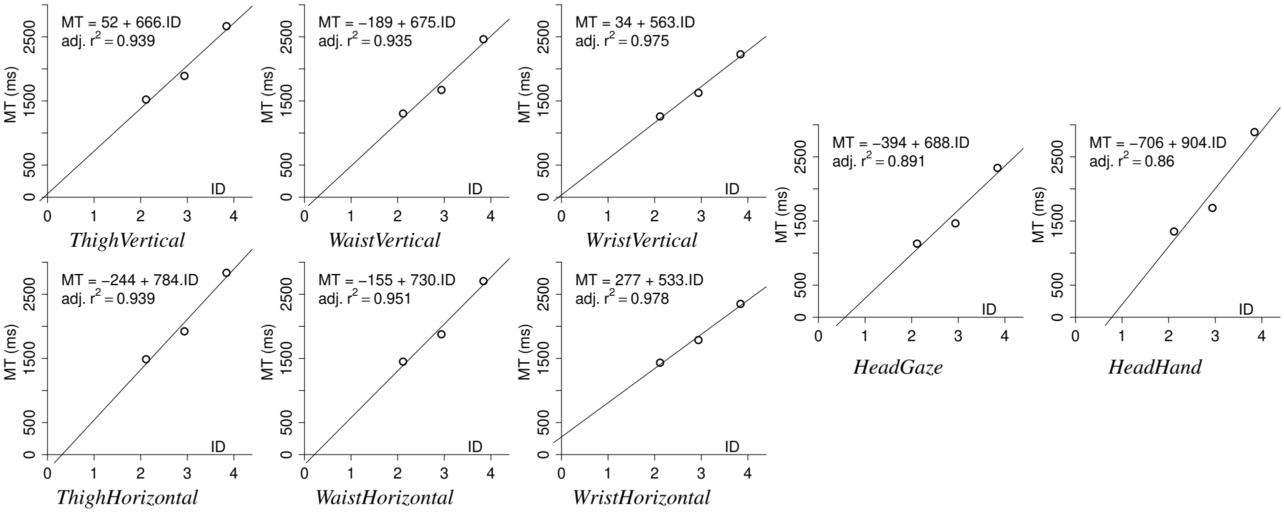


Figure 11: Linear regression analysis with respect to Fitts' law (Movement Time as a function of Index of Difficulty) for all techniques.

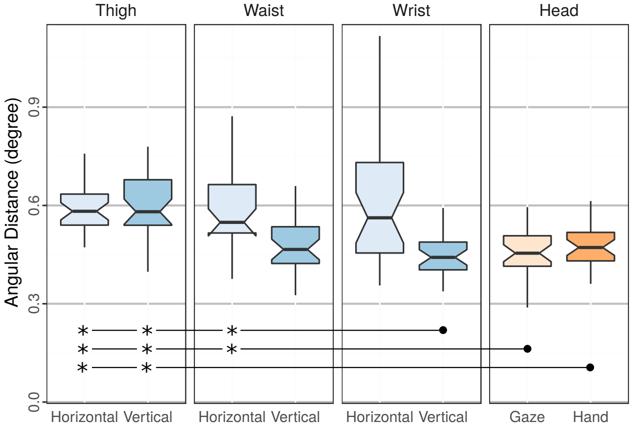


Figure 12: Notched box plot for the average angular distance in *degree* between the cursor and the target during the pursuit. Results are grouped by TECHNIQUE.

above the line that would pass through the other two points (large and medium targets). This indicates a "small target effect" [9], with small targets being more difficult to acquire than what Fitts' law predicts. This effect is stronger for the direct techniques than it is for the indirect techniques, and it is particularly strong for *HeadGaze*. This accounts for the low adjusted r^2 for the direct techniques, as well as the high (*i.e.*, bad) slope of their regression lines. Again, these results are consistent with the precision issues that users encounter with direct techniques. Interestingly, we can also notice that *WristHorizontal* has the lowest (*i.e.*, best) slope, but a very high intercept. This suggests that users might have a bit of trouble in initiating a movement with this technique, but that it likely better scales with the task difficulty.

5.7.2 Pursuit Task

For the pursuit task, our measure is the average angular distance between the target and the cursor (*AngularDistance*) over a 30s trial.

Direct and Indirect Techniques (*AngularDistance*). An ANOVA for the model *AngularDistance* $\sim$ TECHNIQUE reveals a significant effect of TECHNIQUE ($F_{7,105} = 7.65$, $p < 0.001$, $\eta^2_G = 0.27$). Fig. 12 shows the comparative performance of techniques. **The best performing techniques are *HeadGaze*, *WristVertical*, *HeadHand*, and *WaistVertical*.** Corrected t-tests ($n = 28$) do not reveal any significant difference between these four techniques. This suggests that *HeadHand* performs comparatively better for continuous tasks than it does for pointing tasks.

Pairwise t-tests also reveal a few interesting differences. First, the two *ARPads* at *Thigh* perform significantly worse than the two direct techniques and than *WristVertical* (p's<0.01). Second, *WaistHorizontal* performs significantly worse than both *WristVertical* ($p=0.015$) and *HeadGaze* ($p=0.031$).

Indirect Techniques only (*AngularDistance*). To analyze the effect of PADORIENTATION, we filter out data for indirect techniques only (as we did for pointing tasks above). An ANOVA for model *AngularDistance* $\sim$ BODYPOSITION × PADORIENTATION reveals a significant effect of PADORIENTATION ($F_{1,15} = 10.8$, $p = 0.005$, $\eta^2_G = 0.10$) and a significant BODYPOSITION × PADORIENTATION interaction ($F_{2,30} = 6.63$, $p = 0.004$, $\eta^2_G = 0.09$), but no effect of BODYPOSITION ($F_{2,30} = 1.70$, $p = 0.200$, $\eta^2_G = 0.04$). Pairwise t-tests reveal where the interaction comes from: *Vertical ARPads* perform better than *Horizontal* ones only for *Waist* ($p=0.033$) and *Wrist* ($p=0.011$) body positions. Whatever their orientation, ***ARPads* located at the thigh perform particularly poorly for continuous tasks.** But, overall, **our study suggests that *Vertical ARPads* are more efficient than *Horizontal* ones** (H_3).

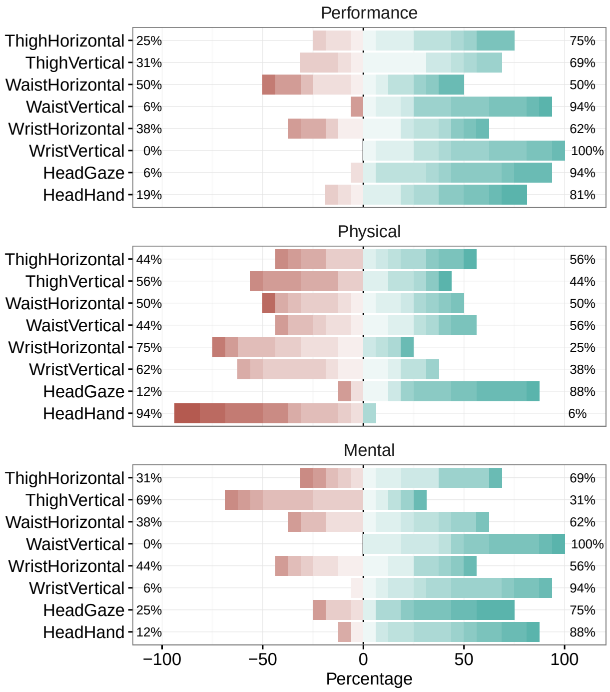


Figure 13: Participants' grades for the following three NASA-TLX scales: perceived performance, physical demand, and mental demand. The plot relies on color coding and binning: each score is assigned a color, and the color patch's size is proportional to the ratio of times this score was given. Right (green) is better.

5.7.3 Qualitative Results

Fig. 13 shows participants' qualitative feedback for each technique along the following scales: perceived performance, physical demand, and mental demand. ART ANOVAs reveal a significant effect of TECHNIQUE for each of these scales: $F_{7,105} = 4.97$, $p < 0.001$ for perceived performance, $F_{7,105} = 9.70$, $p < 0.001$ for physical demand, and $F_{7,105} = 6.69$, $p < 0.001$ for mental demand.

Participants' perceived performance is consistent with quantitative data. They feel most efficient with *WristVertical*, *WaistVertical* and *HeadGaze*. However, few of the differences are statistically significant. Only *WristVertical* and *HeadGaze* are graded significantly better than *ThighVertical* ($p=0.040$ and $p=0.014$).

Grades associated with physical demand are in line with H_1: ***HeadHand* is the most tiring technique on average**. *HeadHand* is actually significantly more demanding than all other techniques (p's<0.038) except *WristHorizontal* ($p=0.098$). Regarding physical demand, the only other significant differences are about *HeadGaze*, which is graded as less tiring than *WristHorizontal* ($p=0.005$) and *WristVertical* ($p=0.022$). Average grades for *Wrist* techniques are actually high. This is interesting: while participants were fast and precise with *WristVertical*, they also found it quite tiring.

Regarding mental demand, participants found it particularly demanding to interact with *ThighVertical*, which has has been graded significantly worse than both direct techniques (p's<0.037), *WaistVertical* ($p=0.013$), and *WristVertical* ($p=0.011$). Also, for *Waist* and *Wrist*, horizontal *ARPads* have been graded significantly more demanding than vertical ones ($p=0.021$ and $p=0.030$, respectively).

Regarding PADORIENTATION, we also asked participants about their preferred orientation for each BODYPOSITION. Fig. 14 shows participants' answers. While preferences are unclear between the two *Thigh* techniques ($p=0.210$), they preferred *Vertical* for both *Waist* ($p<0.001$) and *Wrist* ($p=0.021$). We also asked them about their preferences between the two direct techniques: they all preferred *Gaze* over *Hand*.

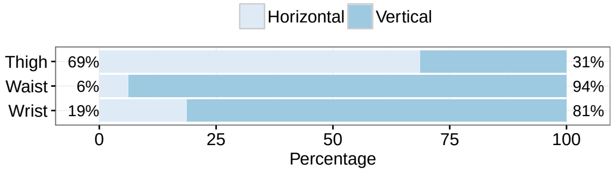


Figure 14: Preferred PADORIENTATION per BODYPOSITION for indirect techniques (for direct techniques, all participants consistently preferred *Gaze* over *Hand*).

The results of the above two paragraphs are in line with H_3. The larger the difference between the visual space and the motor space, the less usable an *ARPad* is (more tiring and less liked). However, the *Thigh* seems to be an exception: *ThighVertical* received a very poor grade for mental demand and there is no clear PADORIENTATION preference between the two *ARPads*. This echoes participants' feedback during the experiment: seven of them mentioned having trouble understanding how the cursor's position was mapped to hand movements in both *Thigh* conditions. This might be because of *ARPads*' lateral location in comparison with *Wrist* and *Waist* positions which are more centered on the sagittal plane. For example, one participant mentioned that they were expecting vertical *ARPads* at the *Thigh* to be parallel to the sagittal plane (*i.e.*, *Vertical (rotated)* in our design space – Fig. 2). We designed it this way to be consistent across body positions, but it might be interesting to reconsider our design for the *Thigh* position. For example, we could position the Leap Motion to make it track in front of the thigh as opposed to sideward, as proposed by two participants.

6 IMPLICATIONS FOR DESIGN

Data collected in our experiment support most of our hypotheses, and bring additional insights from which we can derive guidelines about implementing indirect input for AR headsets.

6.1 Direct Input vs. Indirect Input

First, users actually perform better on average with direct gaze input than with any of the other techniques we have tested. However, well-designed *ARPads* allow users to reach a similar level of performance overall. In particular, our study showed that direct input suffers from precision issues, and that indirect input might be better suited to interaction with small objects.

Second, as we hypothesized, direct hand input is very tiring. However, our study also showed that direct hand input (*HeadHand*) performs quite well for continuous tasks, better than it does for pointing tasks. It thus remains a good candidate to consider for brief manipulations. However, it quickly causes fatigue, so its use should be limited to manipulations that do not last too long. To give a rough estimate, pursuit tasks in our experiment lasted only 30s but already caused much fatigue. Direct hand input should definitely be considered neither for long manipulations of objects nor for repetitive tasks.

6.2 Guidelines for the Design of Indirect Input

The data collected yield guidelines for practitioners regarding *ARPads*'s position as well as their orientation.

Users' performance with *Thigh* was poor in comparison to *Waist* and *Wrist*. Comparison between *Waist* and *Wrist* is more nuanced. Participants performed slightly better with *ARPads* located at *Wrist* for discrete tasks. However, qualitative feedback also showed that they found them more tiring than *ARPads* at *Waist* position. Furthermore, one limitation of the *Wrist* position is that it prevents users from performing bi-manual interactions for, *e.g.*, controlling two cursors. *ARPads* at *Wrist* are good candidates for short-duration interactions, and they might be particularly relevant for some situations such as, *e.g.*, remote control in a seated position. But, overall, *Waist* seems to be the best compromise. Users were efficient and felt comfortable with this body position.

Participants performed better and expressed a clear preference for *Vertical ARPads* over their *Horizontal* counterparts. Despite experience with trackpads on laptops, a difference in orientation between visual and motor spaces seems to have a negative impact.

7 CONCLUSION AND FUTURE WORK

Interaction with HMDs usually benefits from an external device. Bare-hand input can complement device-based techniques when users want to keep their hands free or when they do not have access to an external device. However, implementing bare-hand input that does not cause too much fatigue is challenging. We contribute guidelines for such cases by investigating a design space for *ARPads*. We report on a study evaluating some *ARPad* designs and how they compare with direct input baselines. In our study, gaze input performed very well but suffered from precision issues. Participants' performance with *ARPads* suggest that well-designed indirect input can achieve the same level of performance as direct gaze input. In addition, *ARPads* are particularly relevant for long-lasting or precise interactions as they minimize fatigue and cope with precision issues related to gaze input.

We do not advocate for replacing direct input with indirect input, however. Direct input actually performed well in our study and is probably more intuitive. We rather believe that both types of input should co-exist, allowing users to freely choose the type of input that best suits their context and task. For example, both types of input could be combined in bi-manual techniques where one hand performs direct object selection, and the other hand performs long or fine parameter adjustments using indirect input. Users could also decide to rely on indirect input when they want their interactions be more subtle [3] or rely on direct input when, on the contrary, they want them be transparent to others [15]. Supporting the coexistence of both requires implementing mode switching techniques. Such mode switches could be implicit, relying on hand location relative to headset orientation, in the spirit of the Gaze-Shifting technique for pen input [55].

Finally, this preliminary study suggests directions for future work. First, *ARPads* tested in this study implement absolute control with a headset that has a limited-size viewport. Considering absolute control was necessary to test the effect of an *ARPad*'s position on users' performance without any confound from a posture recognition engine (for *e.g.*, clutching actions for relative control). A larger viewport would require considering one or several of the following strategies: increasing the CD gain, enlarging *ARPads*, or implementing a clutching mechanism. Second, complementing *ARPads* with subtle audio or haptic feedback could help users get a better sense of where the pad stands in the air. This is particularly relevant when relying on technology such as the Leap Motion, which has tracking limitations. Third, our informal observations suggest that users do not adopt planar movements when interacting with an *ARPad*. A fine-grained analysis of users' movements could help refine the pads' orientation, or even consider pad surfaces that are not planar. Finally, in our experiment, participants were interacting with graphical objects that were displayed in a world-anchored window that was in front of them. It would be interesting to study cases where users are not well positioned in front of the window they are interacting with.

ACKNOWLEDGMENTS

This research was partly supported by project VizGest (CNRS PEPS S2IH - 2018). We wish to thank the participants of our experiment. We also wish to thank Emmanuel Pietriga and all ILDA members for their feedback.

REFERENCES

[1] D. Ahlström, K. Hasan, and P. Irani. 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, pp. 193–202. ACM, 2014. doi: 10.1145/2628363.2628381

[2] S. Ahn, S. Heo, and G. Lee. Typing on a smartwatch for smart glasses. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces*, ISS '17, pp. 201–209. Association for Computing Machinery, 2017. doi: 10.1145/3132272.3134136

[3] F. Anderson, T. Grossman, D. Wigdor, and G. Fitzmaurice. Supporting subtlety with deceptive devices and illusory interactions. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pp. 1489–1498. ACM, 2015. doi: 10.1145/2702123.2702336

[4] C. Arslan, Y. Rekik, and L. Grisoni. E-Pad: Large display pointing in a continuous interaction space around a mobile device. In *Proceedings of the Conference on Designing Interactive Systems*, DIS '19, pp. 1101–1108. ACM, 2019. doi: 10.1145/3322276.3322284

[5] M. Bachynskyi, G. Palmas, A. Oulasvirta, J. Steimle, and T. Weinkauf. Performance and ergonomics of touch surfaces: A comparative study using biomechanical simulation. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pp. 1817–1826. ACM, 2015. doi: 10.1145/2702123.2702607

[6] G. Bailly, J. Müller, M. Rohs, D. Wigdor, and S. Kratz. ShoeSense: A new perspective on gestural interaction and wearable applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pp. 1239–1248. ACM, 2012. doi: 10.1145/2207676.2208576

[7] M. Botvinick and J. Cohen. Rubber hands 'feel' touch that eyes see. *Nature*, 391(6669):756, 1998. doi: 10.1038/35784

[8] R. Budhiraja, G. A. Lee, and M. Billinghurst. Using a HHD with a HMD for mobile AR interaction. In *IEEE International Symposium on Mixed and Augmented Reality*, ISMAR '13, pp. 1–6. IEEE, 2013. doi: 10.1109/ISMAR.2013.6671837

[9] O. Chapuis and P. Dragicevic. Effects of motor scale, visual scale, and quantization on small target acquisition difficulty. *ACM Trans. Comput.-Hum. Interact.*, 18(3), 2011. doi: 10.1145/1993060.1993063

[10] X. A. Chen, J. Schwarz, C. Harrison, J. Mankoff, and S. E. Hudson. Air+Touch: Interweaving touch & in-air gestures. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pp. 519–525. ACM, 2014. doi: 10.1145/2642918.2647392

[11] Y. Chen, K. Katsuragawa, and E. Lank. Understanding viewport- and world-based pointing with everyday smart devices in immersive augmented reality. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, pp. 1–13. Association for Computing Machinery, 2020. doi: 10.1145/3313831.3376592

[12] A. Colaço, A. Kirmani, H. S. Yang, N.-W. Gong, C. Schmandt, and V. K. Goyal. Mime: Compact, low power 3D gesture sensing for interaction with head mounted displays. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, UIST '13, pp. 227–236. ACM, 2013. doi: 10.1145/2501988.2502042

[13] D. Dobbelstein, P. Hock, and E. Rukzio. Belt: An unobtrusive touch input device for head-worn displays. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pp. 2135–2138. ACM, 2015. doi: 10.1145/2702123.2702450

[14] D. Dobbelstein, C. Winkler, G. Haas, and E. Rukzio. PocketThumb: A wearable dual-sided touch interface for cursor-based control of smart-eyewear. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(2):9:1–9:17, 2017. doi: 10.1145/3090055

[15] B. Ens, T. Grossman, F. Anderson, J. Matejka, and G. Fitzmaurice. Candid interaction: Revealing hidden mobile and wearable computing activities. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, UIST '15, pp. 467–476. ACM, 2015. doi: 10.1145/2807442.2807449

[16] B. Ens, A. Quigley, H.-S. Yeo, P. Irani, T. Piumsomboon, and M. Billinghurst. Counterpoint: Exploring mixed-scale gesture interaction for AR applications. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, CHI EA '18, pp. LBW120:1–LBW120:6. ACM, 2018. doi: 10.1145/3170427.3188513

[17] T. Feuchtner and J. Müller. Ownershift: Facilitating overhead interaction in virtual reality with an ownership-preserving hand space shift. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, UIST '18, pp. 31–43. ACM, 2018. doi: 10.1145/3242587.3242594

[18] B. Fruchard, E. Lecolinet, and O. Chapuis. Impact of semantic aids on command memorization for on-body interaction and directional gestures. In *Proceedings of the International Conference on Advanced Visual Interfaces*, AVI '18, pp. 14:1–14:9. ACM, 2018. doi: 10.1145/3206505.3206524

[19] M. Gandy, T. Starner, J. Auxier, and D. Ashbrook. The gesture pendant: A self-illuminating, wearable, infrared computer vision system for home automation control and medical monitoring. In *Proceedings of the 4th IEEE International Symposium on Wearable Computers*, ISWC '00, pp. 87–94. IEEE, 2000. doi: 10.1109/ISWC.2000.888469

[20] R. Gervais, J. Frey, and M. Hachet. Pointing in spatial augmented reality from 2D pointing devices. In *Proceedings of the 15th IFIP TC 13 International Conference on Human-computer Interaction*, INTERACT'15, pp. 381–389. Springer-Verlag, 2015. doi: 10.1007/978-3-319-22723-8_30

[21] J. Grubert, M. Heinisch, A. Quigley, and D. Schmalstieg. Multifi: Multi fidelity interaction with displays on and around the body. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pp. 3933–3942. Association for Computing Machinery, 2015. doi: 10.1145/2702123.2702331

[22] J. Grubert, M. Kranz, and A. Quigley. Challenges in mobile multi-device ecosystems. *mUX: The Journal of Mobile User Experience*, 5(1):1–22, 2016.

[23] J. Grubert, L. Witzani, E. Ofek, M. Pahud, M. Kranz, and P. O. Kristensson. Text entry in immersive head-mounted display-based virtual reality using standard keyboards. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 159–166, 2018.

[24] Y. Guiard. Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. *Journal of motor behavior*, 19(4):486–517, 1987. doi: 10.1080/00222895.1987.10735426

[25] S. Gustafson, D. Bierwirth, and P. Baudisch. Imaginary interfaces: Spatial interaction with empty hands and without visual feedback. In *Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology*, UIST '10, pp. 3–12. ACM, 2010. doi: 10.1145/1866029.1866033

[26] T. Ha, S. Feiner, and W. Woo. Wearhand: Head-worn, rgb-d camera-based, bare-hand user interface with visually enhanced depth perception. In *IEEE International Symposium on Mixed and Augmented Reality*, ISMAR '14, pp. 219–228. IEEE, 2014. doi: 10.1109/ISMAR.2014.6948431

[27] J. Han, S. Ahn, K. Park, and G. Lee. Designing touch gestures using the space around the smartwatch as continuous input space. In *Proceedings of the ACM International Conference on Interactive Surfaces and Spaces*, ISS '17, pp. 210–219. ACM, 2017. doi: 10.1145/3132272.3134134

[28] C. Harrison and S. E. Hudson. Abracadabra: Wireless, high-precision, and unpowered finger input for very small mobile devices. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology*, UIST '09, pp. 121–124. ACM, 2009. doi: 10.1145/1622176.1622199

[29] C. Harrison, D. Tan, and D. Morris. Skinput: Appropriating the body as an input surface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pp. 453–462. ACM, 2010. doi: 10.1145/1753326.1753394

[30] 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, 2014. doi: 10.1145/2556288.2557130

[31] T. Hothorn, K. Hornik, M. A. van de Wiel, and A. Zeileis. Implementing a class of permutation tests: The coin package. *Journal of Statistical Software*, 28(8):1–23, 2008.

[32] ISO. 9241-9 Ergonomic requirements for office work with visual display terminals (VDTs)-Part 9: Requirements for non-keyboard input

devices. *International Organization for Standardization*, 2000.

[33] 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, 1990. doi: 10.1145/97243.97246

[34] R. J. K. Jacob. Eye tracking in advanced interface design. In W. Barfield and T. A. Furness, III, eds., *Virtual Environments and Advanced Interface Design*, pp. 258–288. Oxford University Press, New York, NY, USA, 1995. doi: 10.1016/0160-9327(96)88424-9

[35] R. Jota, M. A. Nacenta, J. A. Jorge, S. Carpendale, and S. Greenberg. A comparison of ray pointing techniques for very large displays. In *Proceedings of Graphics Interface 2010*, GI '10, pp. 269–276. Canadian Information Processing Society, 2010.

[36] M. Kay and J. O. Wobbrock. *ARTool: Aligned Rank Transform for Nonparametric Factorial ANOVAs*, 2020. R package version 0.10.7. doi: 10.5281/zenodo.594511

[37] M. Khamis, C. Oechsner, F. Alt, and A. Bulling. VRpursuits: Interaction in virtual reality using smooth pursuit eye movements. In *Proceedings of the International Conference on Advanced Visual Interfaces*, AVI '18, pp. 18:1–18:8. ACM, 2018. doi: 10.1145/3206505.3206522

[38] J. Kim, J. He, K. Lyons, and T. Starner. The gesture watch: A wireless contact-free gesture based wrist interface. In *Proceedings of the 11th IEEE International Symposium on Wearable Computers*, ISWC '07, pp. 1–8. IEEE, 2007. doi: 10.1109/ISWC.2007.4373770

[39] M. Koelle, A. El Ali, V. Cobus, W. Heuten, and S. C. Boll. All about acceptability?: Identifying factors for the adoption of data glasses. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, CHI '17, pp. 295–300. ACM, 2017. doi: 10.1145/3025453.3025749

[40] M. Kytö, B. Ens, T. Piumsomboon, G. A. Lee, and M. Billinghurst. Pinpointing: Precise head- and eye-based target selection for augmented reality. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, CHI '18, pp. 81:1–81:14. ACM, 2018. doi: 10.1145/3173574.3173655

[41] M. A. Lawrence. *ez: Easy Analysis and Visualization of Factorial Experiments*, 2016. R package version 4.4-0.

[42] D. Lee, Y. Lee, Y. Shin, and I. Oakley. Designing socially acceptable hand-to-face input. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, UIST '18, pp. 711–723. ACM, 2018. doi: 10.1145/3242587.3242642

[43] M. Liu, M. Nancel, and D. Vogel. Gunslinger: Subtle arms-down mid-air interaction. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, UIST '15, pp. 63–71. ACM, 2015. doi: 10.1145/2807442.2807489

[44] N. Marquardt, R. Jota, S. Greenberg, and J. A. Jorge. The continuous interaction space: Interaction techniques unifying touch and gesture on and above a digital surface. In *Proceedings of the 13th IFIP TC 13 International Conference on Human-computer Interaction - Volume Part III*, INTERACT'11, pp. 461–476. Springer-Verlag, 2011. doi: 10.1007/978-3-642-23765-2_32

[45] A. Millette and M. J. McGuffin. DualCAD: Integrating augmented reality with a desktop GUI and smartphone interaction. In *IEEE International Symposium on Mixed and Augmented Reality*, ISMAR-Adjunct '16, pp. 21–26. IEEE, 2016. doi: 10.1109/ISMAR-Adjunct.2016.0030

[46] M. R. Mine, F. P. Brooks, Jr., and C. H. Sequin. Moving objects in space: Exploiting proprioception in virtual-environment interaction. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, pp. 19–26. ACM & Addison-Wesley, 1997. doi: 10.1145/258734.258747

[47] P. Mistry and P. Maes. Sixthsense: A wearable gestural interface. In *ACM SIGGRAPH ASIA 2009 Sketches*, SIGGRAPH ASIA '09, pp. 11:1–11:1. ACM, 2009. doi: 10.1145/1667146.1667160

[48] P. Mohr, M. Tatzgern, T. Langlotz, A. Lang, D. Schmalstieg, and D. Kalkofen. Trackcap: Enabling smartphones for 3d interaction on mobile head-mounted displays. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, pp. 1–11. Association for Computing Machinery, 2019. doi: 10.1145/3290605.3300815

[49] R. A. Montano Murillo, S. Subramanian, and D. Martinez Plasencia. Erg-O: Ergonomic optimization of immersive virtual environments. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, pp. 759–771. ACM, 2017. doi: 10.1145/3126594.3126605

[50] F. Müller, N. Dezfuli, M. Mühlhäuser, M. Schmitz, and M. Khalilbeigi. Palm-based interaction with head-mounted displays. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*, MobileHCI '15, pp. 963–965. ACM, 2015. doi: 10.1145/2786567.2794314

[51] F. Müller, J. McManus, S. Günther, M. Schmitz, M. Mühlhäuser, and M. Funk. Mind the tap: Assessing foot-taps for interacting with head-mounted displays. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, CHI '19, pp. 477:1–477:13. ACM, 2019. doi: 10.1145/3290605.3300707

[52] M. Nancel, J. Wagner, E. Pietriga, O. Chapuis, and W. Mackay. Mid-air pan-and-zoom on wall-sized displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pp. 177–186. ACM, 2011. doi: 10.1145/1978942.1978969

[53] E. Normand and M. J. McGuffin. Enlarging a smartphone with ar to create a handheld vesad (virtually extended screen-aligned display). In *2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 123–133, 2018.

[54] I. Pedersen, N. Gale, P. Mirza-Babaei, and S. Reid. More than meets the eye: The benefits of augmented reality and holographic displays for digital cultural heritage. *J. Comput. Cult. Herit.*, 10(2):11:1–11:15, 2017. doi: 10.1145/3051480

[55] K. Pfeuffer, J. Alexander, M. K. Chong, Y. Zhang, and H. Gellersen. Gaze-Shifting: Direct-Indirect input with pen and touch modulated by gaze. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, UIST '15, pp. 373–383. ACM, 2015. doi: 10.1145/2807442.2807460

[56] K. Pfeuffer, B. Mayer, D. Mardanbegi, and H. Gellersen. Gaze + Pinch interaction in virtual reality. In *Proceedings of the 5th Symposium on Spatial User Interaction*, SUI '17, pp. 99–108. ACM, 2017. doi: 10.1145/3131277.3132180

[57] J. S. Pierce, A. S. Forsberg, M. J. Conway, S. Hong, R. C. Zeleznik, and M. R. Mine. Image plane interaction techniques in 3D immersive environments. In *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, I3D '97, pp. 39–ff. ACM, 1997. doi: 10.1145/253284.253303

[58] K. Pietroszek, L. Tahai, J. R. Wallace, and E. Lank. Watchcasting: Freehand 3d interaction with off-the-shelf smartwatch. In *2017 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 172–175, 2017.

[59] T. Piumsomboon, D. Altimira, H. Kim, A. Clark, G. Lee, and M. Billinghurst. Grasp-Shell vs Gesture-Speech: A comparison of direct and indirect natural interaction techniques in augmented reality. In *International Symposium on Mixed and Augmented Reality*, ISMAR '14, pp. 73–82. IEEE, 2014. doi: 10.1109/ISMAR.2014.6948411

[60] T. Piumsomboon, G. Lee, R. W. Lindeman, and M. Billinghurst. Exploring natural eye-gaze-based interaction for immersive virtual reality. In *IEEE Symposium on 3D User Interfaces*, 3DUI '17, pp. 36–39. IEEE, 2017. doi: 10.1109/3DUI.2017.7893315

[61] E. C. Poulton. *Tracking skill and manual control*. Academic press, 1974.

[62] I. Poupyrev, M. Billinghurst, S. Weghorst, and T. Ichikawa. The Go-go interaction technique: Non-linear mapping for direct manipulation in VR. In *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology*, UIST '96, pp. 79–80. ACM, 1996. doi: 10.1145/237091.237102

[63] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2018.

[64] H. Saidi, M. Serrano, P. Irani, C. Hurter, and E. Dubois. On-body tangible interaction: Using the body to support tangible manipulations for immersive environments. In *Proceedings of the 17th IFIP TC 13 International Conference on Human-computer Interaction*, INTERACT'19, pp. 471–492. Springer, 2019.

[65] M. Serrano, B. M. Ens, and P. P. Irani. Exploring the use of hand-to-face input for interacting with head-worn displays. In *Proceedings of the 32nd Annual ACM Conference on Human Factors in Computing Systems*, CHI '14, pp. 3181–3190. ACM, 2014. doi: 10.1145/2556288.

2556984

[66] R. Sicat, J. Li, J. Choi, M. Cordeil, W.-K. Jeong, B. Bach, and H. Pfister. DXR: A toolkit for building immersive data visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):715–725, 2019. doi: 10.1109/TVCG.2018.2865152

[67] S. Siddhpuria, K. Katsuragawa, J. R. Wallace, and E. Lank. Exploring at-your-side gestural interaction for ubiquitous environments. In *Proceedings of the Conference on Designing Interactive Systems*, DIS '17, pp. 1111–1122. ACM, 2017. doi: 10.1145/3064663.3064695

[68] S. Siddhpuria, S. Malacria, M. Nancel, and E. Lank. Pointing at a distance with everyday smart devices. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, pp. 1–11. Association for Computing Machinery, 2018. doi: 10.1145/3173574.3173747

[69] S. Stellmach and R. Dachselt. Look & Touch: Gaze-supported target acquisition. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pp. 2981–2990. ACM, 2012. doi: 10.1145/2207676.2208709

[70] H. B. Surale, A. Gupta, M. Hancock, and D. Vogel. TabletInVR: Exploring the design space for using a multi-touch tablet in virtual reality. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, CHI '19, pp. 13:1–13:13. ACM, 2019. doi: 10.1145/3290605.3300243

[71] D.-B. Vo, E. Lecolinet, and Y. Guiard. Belly gestures: Body centric gestures on the abdomen. In *Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational*, NordiCHI '14, pp. 687–696. ACM, 2014. doi: 10.1145/2639189.2639210

[72] C. Ware and H. H. Mikaelian. An evaluation of an eye tracker as a device for computer input. In *Proceedings of the SIGCHI/GI Conference on Human Factors in Computing Systems and Graphics Interface*, CHI '87, pp. 183–188. ACM, 1987. doi: 10.1145/29933.275627

[73] H. Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2009.

[74] D. Wolf, J. J. Dudley, and P. O. Kristensson. Performance envelopes of in-air direct and smartwatch indirect control for head-mounted augmented reality. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 347–354, 2018.

[75] K. Yamashita, T. Kikuchi, K. Masai, M. Sugimoto, B. H. Thomas, and Y. Sugiura. CheekInput: Turning your cheek into an input surface by embedded optical sensors on a head-mounted display. In *Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology*, VRST '17, pp. 19:1–19:8. ACM, 2017. doi: 10.1145/3139131.3139146

[76] S. Zhai, C. Morimoto, and S. Ihde. Manual and gaze input cascaded (MAGIC) pointing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '99, pp. 246–253. ACM, 1999. doi: 10.1145/302979.303053

[77] F. Zhu and T. Grossman. Bishare: Exploring bidirectional interactions between smartphones and head-mounted augmented reality. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, pp. 1–14. Association for Computing Machinery, 2020. doi: 10.1145/3313831.3376233