Push or Pinch? Exploring Slider Control Gestures for Touchless User Interfaces

Kieran Waugh
University of Glasgow
Glasgow, Scotland
k.waugh.1@research.gla.ac.uk

Mark McGill
University of Glasgow
Glasgow, Scotland
Mark.McGill@glasgow.ac.uk

Euan Freeman
University of Glasgow
Glasgow, Scotland
Euan.Freeman@glasgow.ac.uk

ABSTRACT

Touchless gesture interfaces enable user interaction without the need for direct physical contact. Recently there has been increased interest in deploying such interfaces over concerns about touchscreen sterility. Many touchless displays use gestures that mimic pointer-based interactions, with a 'cursor' mapped to finger position, that users activate by 'pushing' their finger forwards. Mid-air pushing with a virtual cursor is fine for discrete interactions like button activation; however, users have difficulty exerting control over continuous interactions like sliding and scrolling, because it is challenging to keep the hand at a consistent depth when gesturing. We investigate interaction techniques for slider control that use alternative mode switches between (un)pressed states. Our findings show that pinch gestures are preferred by most users and offer a faster alternative for acquiring control of sliders, as pinching has two clearly defined states and avoids the ambiguous use of depth for delineating input states.

CCS CONCEPTS

• **Human-centered computing → Gestural input**.

KEYWORDS

Mid-Air Gestures, Touchless Interaction, Slider Control

ACM Reference Format:
Kieran Waugh, Mark McGill, and Euan Freeman. 2022. Push or Pinch? Exploring Slider Control Gestures for Touchless User Interfaces. In *Nordic Human-Computer Interaction Conference (NordiCHI '22), October 8–12, 2022, Aarhus, Denmark.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3546155.3546702

1 INTRODUCTION

Touchless gesture interfaces allow users to interact with displays using mid-air hand movements and poses, making it possible to provide input without direct contact with an input device (e.g., touchscreen). Touchless gesture interaction is compelling because it creates new opportunities for interaction, from casual interactions with distal devices [29] to more focused interactions that leverage the dexterous capabilities of our hands [32]. More recently, touchless interaction has seen increased interest due to concerns over the cleanliness and sterility of shared touchscreens [12], with concerns heightened by the Covid-19 pandemic [23] leading to users avoiding touching input devices [28]. Touchless interaction offers a more hygienic alternative to shared touchscreen use because it eliminates surface contact [8]. With touchscreens being so widely used across numerous domains of public display (e.g., retail, transport, tourism), touchless interaction could soon be reaching a wider audience and so addressing usability concerns is a timely concern.

In an attempt to facilitate rapid deployment of touchless interaction, solutions were devised to effectively retrofit touchless gesture sensing to existing touchscreen devices, to meet growing demands for more hygienic interaction. Examples of this include Ultraleap TouchFree [21] and Intel RealSense Touchless [16]. These allow users to control a virtual cursor by moving their hands in mid-air, with activation enabled by an 'air push' gesture [22], i.e., moving the hand towards the display to enter a 'pressed' state. Other commercial gesture interfaces make similar use of virtual cursors for touchless input, e.g., Microsoft's Xbox Kinect also used hand motion for cursor control, with a similar push gestures used to select items beneath the cursor.

This 'air push' interaction paradigm mimics traditional pointer- and touch-based input, which can simplify integration with existing systems, reduce software costs, and maintain consistency and familiarity for users. However, whilst this works well for discrete input actions like targeting and activating buttons, it can be challenging for users to interact with continuous input widgets like sliders and scrollbars [37]. Cursor instability, ambiguity between 'pressed' and 'unpressed' states, and inadvertent cursor displacement whilst the hand pushes forward, all make it difficult for users to acquire and maintain control over these continuous interactions.

In this work, we investigate alternative interaction techniques for continuous cursor control in a touchless user interface. We focus, in particular, on controlled slider input. Sliding and scrolling are fundamental interactions encountered in public display domains, e.g., for browsing lists of transit destinations or menu items, for entering numerical quantities, or for navigating timelines. We investigate 'pinch' and 'dwell' gestures as an alternative to air push for activating scroll widgets in a touchless user interface. These gestures have previously been used for button activation and could also be effective for continuous slider interactions because: (i) they may reduce target slips by allowing the hand to remain stationary during selection, and (ii) they offer more clearly defined state transitions (i.e., pinching two fingers together and pausing for fixed duration, respectively), which mitigates the inherent ambiguity of using hand depth for moving between (un)pressed interface states.

We present an experiment that evaluates slider control with four hand-based activation methods: air push, cursor dwell, and two variations of finger pinching. Participants completed a variety of

NordiCHI '22, October 8–12, 2022, Aarhus, Denmark

slider-based tasks including selection from two types of alphanumerical list and an object scaling task. Our findings show that pinch gestures offer a promising alternative for acquiring and maintaining control over slider widgets in a touchless user interface. Pinching worked well as a mode switch because it has two clearly defined states and does not rely on hand depth as an ambiguous 'pressing' delimiter. Pinching was highly preferred and led to faster slider handle acquisition and reduced time-to-target. For touchless interactions where continuous control is necessary, pinch gestures can be used to complement the air push interaction paradigm for more effective and confident touchless input. As touchless technology reaches a wider audience of public display users, more expressive input techniques like these can help improve the usability of fundamental UI operations.

2 BACKGROUND

2.1 Touchless Cursor-Based Interaction

Touchless interfaces can use a variety of mid-air gesture types and interaction paradigms to enable distal interaction with on-screen content. In this work, we focus on hand-based touchless interfaces that use a user-controlled 'cursor' to target and interact with graphical user interface widgets, like buttons, sliders, and scrollbars. These cursor-based interfaces are the predominant type of touchless interface for public displays, e.g., as supported by Ultraleap TouchFree [21], Intel RealSense Touchless [16], and Microsoft's Xbox Kinect. Gesture-controlled cursors are typically mapped to part of the hand (e.g., the centre of the palm or index fingertip), such that hand movements result in corresponding cursor movements. This allows users to target interface elements by moving their hand so that the cursor hovers on top, analogous to a mouse pointer. Targeted interface elements are then typically activated through a discrete gesture, e.g., *pushing* the hand towards the screen [22] or *tapping* downwards with an extended finger [6]. We call this the air push interaction paradigm, named after the primary input gesture for Ultraleap TouchFree [22].

In an air push interface, users control the cursor through hand movements in a region in front of the screen. This region is not visible to users and lacks other sensory feedback, which can cause usability issues including uncertainty about where to perform gestures [10, 11] and how to initiate interaction [8], lack of confidence during cursor control [9], and diminished feelings of control and responsiveness during button activation gestures [6]. These issues arise over ambiguity about the role that hand depth plays in the interaction. The relationship between horizontal and vertical hand movements is more clear as these result in congruent and highly visible cursor movements on the 2D screen. However, users get limited feedback about motions towards the display to inform their understanding about how this third dimension is (or is not) used for input to differentiate between pressed and unpressed states. There is also the risk that forward movement results in hands moving *too close* to the input sensor, leading to loss of tracking during the push gesture. In addition to the ambiguity over depth in air push gestures, usability issues also arise from incidental hand movements during the push gesture itself. If push gestures are not perpendicular to the display, incidental translation can cause *target slips* [2], where the cursor moves and falls off the target widget before there is enough forward movement to enter the pressed state.

Efforts have been made to address these issues: Appropriate signage [19] and peripheral visual cues [10, 11] have been successful in guiding users to adopt better hand position so that sensing quality is improved for more reliable input recognition. Rhythmic path mimicry [3, 5], where users replicate the movement of a screen-represented patterns (with their hand) have shown benefits for large display interaction and multiple user support. Novel haptic feedback devices like wearable vibrotactile displays [9] and ultrasound haptic displays [6, 30, 35] have been used to give extra sensory feedback so users can feel when they are 'pressing' touchless interface controls. Others have addressed issues like target slips and false-positive gesture recognition through alternative selection techniques, based on temporal coupling rather than spatial targeting [4, 10, 24, 34]. For the most part, however, air push works well enough for straightforward button pressing interactions. However, our earlier work found that users encounter difficulties during more complex operations like sliding and scrolling [37].

2.2 Touchless Slider Control

To better understand the problem with sliding and scrolling, it helps to think of how slider widgets behave in a typical air push user interface. Users must first target the slider handle with the virtual cursor, then perform the air push gesture to take control of it. With the hand still in the pressed state (c.f., holding down a mouse button with the pointer over a slider handle), users must then translate the slider to the desired target position before then releasing the handle by returning to the unpressed state. We see this as an instance of *steering* [1] a cursor in mid-air, as users must maintain control of the slider whilst moving the cursor along the slider trajectory. The added constraint of maintaining the 'pressed' state increases the difficulty of this task as users must maintain hand depth [17].

When using an air push gesture to take control of a slider handle, the usability issues we discussed earlier for button pressing remain relevant and are made worse by the need to maintain continuous control while steering, as opposed to a button press action that simply enacts a discrete event. We therefore consider alternative interaction techniques for taking control of a slider, with a goal of reducing the use of hand depth as an input constraint.

One alternative is the use of *cursor dwell*, where targeted widgets are selected by hovering the cursor in place for a brief period of time. Dwell can improve cursor-based interactions by effectively reducing gesture input to two degrees of freedom, as hand movements only result in horizontal/vertical cursor translation. A notable example of touchless cursor dwell can be found in the original Microsoft Xbox Kinect, which used a dwell time of approx. 1500ms for button activation. Others have used much shorter dwell times, e.g., 600ms [14]. We include dwell in our experiment because it has been widely used for touchless button activation and could be a promising alternative for acquiring control of touchless slider handles, since it does not require hand displacement to enact selection.

As an alternative to air push as a 'mode switch' gesture, we consider the use of other hand gestures for acquiring control of the slider handle. Modern touchless input sensors can fully track fingertip positions and joint articulations, allowing the recognition of complex hand postures with many degrees of freedom [32]. Candidate hand gestures for slider handle selection are those that can be performed with minimal hand translation or rotation, since minimal cursor movement is desired. We chose *pinch gestures* for this purpose, where the index finger and thumb are brought together. Pinch gestures are easy to detect with optical tracking systems [38] and have been widely used as a mode switch, e.g., for selection in mixed reality gesture interfaces [33]. We extend this to consider their use as a mode switch for acquiring control of a slider handle for continuous sliding interactions: i.e., users pinch to take control of the slider, move their hand to change the slider position, then release the pinch gesture to relinquish control.

Cursor dwell and pinch gestures are compelling alternatives to the air push gesture for continuous cursor-based interactions like sliding. These approaches allow the user's hand to remain steady when acquiring control of the slider handle and they avoid the ambiguous use of hand depth as a mode switch, both of which may help address the usability issues and user frustration with touchless sliding operations [37]. They also complement the air push gesture vocabulary, as dwelling in place and pinching are not meaningful actions for input. Our aim in this paper is to experimentally compare **dwell** and **pinch gestures** to **air push** as a baseline, to better understand how slider control technique affects usability and task performance across a range of touchless slider tasks. Our main contribution is a formative look at how continuous interactions can be improved for touchless public display interfaces.

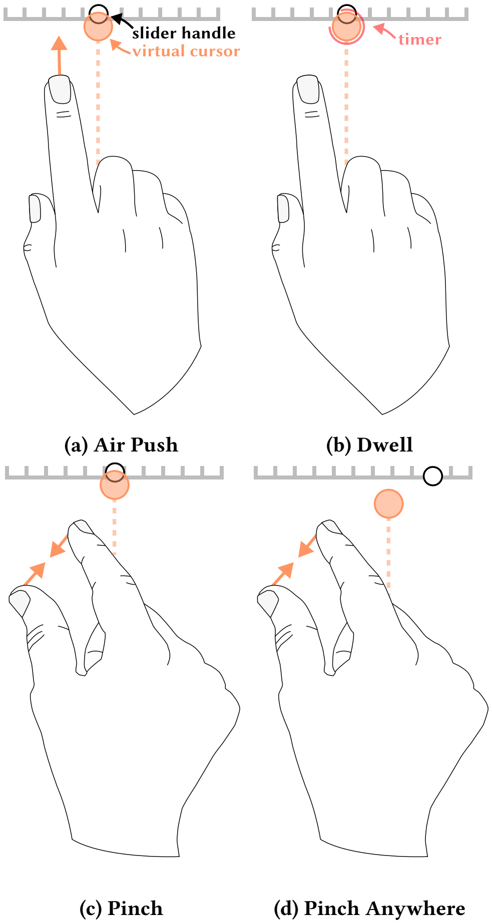
3 EXPERIMENT

3.1 Study Design

The aim of this experiment was to evaluate the usability and task performance of the slider control techniques introduced previously. We investigated four techniques in this experiment: **Air Push** (baseline technique that replicates the behaviour of Ultraleap TouchFree [21]), **Dwell** (cursor dwell technique), **Pinch** (using finger-to-thumb pinching as a mode switch gesture), and **Pinch Anywhere** (a variant of pinch that automatically grabs the slider handle without the cursor overlapping it), as illustrated in Figure 1.

We included the pinch anywhere variant because finger pinching is not already a meaningful action in the air push interaction paradigm, and so could potentially be used exclusively for taking control of move-able user interface elements (like slider and scroll handles), so long as the cursor is nearby. Whilst we expected this might be better than pinch (with the cursor over the slider handle), we included both in our experiment to better understand the benefits (or otherwise) of allowing slider control without cursor overlap.

We chose three task types for this experiment, all involving selections made using a horizontal slider of fixed length. Two tasks involved moving the slider handle to a designated target, selecting either a **digit** from 0–9 or a **letter** from A–Z. These tasks give insight into the effect of target width (i.e., 10 targets vs 26 more narrow targets) and represent the slider selections one might expect to find in a public display (e.g., selecting numerical quantities at a supermarket checkout, or filtering the start letter of destinations on a ticket machine). The third task involved using the slider to change the **scale** of a square to match another shown on screen. We included the scale task as a more difficult alternative, requiring users to select a near pixel-perfect slider position versus having wider targets. Task contexts requiring increased precision imply increased movement time [25], which we expected to increase the difficulty of maintaining control of the slider. Figure 2 shows screenshots of each task type.



(a) Air Push

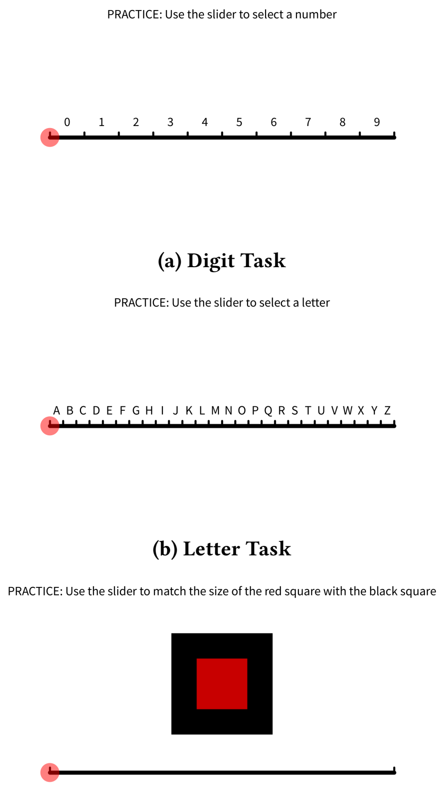
(b) Dwell

(c) Pinch

(d) Pinch Anywhere

Figure 1: Illustration of the four approaches for taking control of the slider handle. The cursor position was mapped to the centre of the palm. Note that while these images show an extended index finger, this was not required; users could gestures with any posture.

We used a within-subject experiment design with slider control technique and task type as independent variables, giving twelve conditions (four techniques × three task types). Condition order was randomised using a Latin Square, and participants completed twelve selection trials for each condition. All tasks started with the slider in the left-most position of the bar. For the digit and letter tasks, we used fixed sets of targets (randomised per participant) such that no selections were made within the first three slider targets (i.e., 0–2 and A–C), so there was always some distance between the start and target positions.



PRACTICE: Use the slider to select a number

(a) Digit Task

PRACTICE: Use the slider to select a letter

(b) Letter Task

PRACTICE: Use the slider to match the size of the red square with the black square

(c) Scale Task

Figure 2: Screenshots showing each task type as shown to the participants. The semi-transparent red circle is the slider handle. For the digit and letter tasks, slider ticks were rendered to indicate target width. For the scale task, participants had to use the slider to scale the size of the semi-transparent red shape so that it matched the size of the black shape.

3.2 Study Procedure

Participants sat in front of a table with a 27" display, Leap Motion sensor, and wireless keyboard placed in front of them. Following Ultraleap's recommendations [20], the sensor was placed 15cm below and 10cm in front of the display. We angled the sensor so that it pointed 15° towards the participant, so they did not have to place their hand immediately above the sensor and could gesture from a more relaxed posture (to improve sensing quality and reduce fatigue [15]).

Before completing the experimental trials, participants were introduced to the slider control gestures and were given a chance to complete as many practice trials as they wanted for each technique and task type. For each trial, instructions were presented on screen (e.g., *"Use the slider to select the letter N"*). Every trial started with participants' hands by the side of their body, so they began each task by reaching in front of the Leap Motion sensor. Participants had no time constraints and were allowed to release the slider and make corrections if wanted (rather than just being allowed one slider movement). Each trial ended when the participant pressed the spacebar on the keyboard in front of them. Following each block of tasks for each interaction technique, participants completed the NASA-TLX survey [13] as a measure of task workload. We also asked participants to rate their agreement with statements about various aspects of slider control, for each technique, using a seven point Likert scale. The following statements were used: Q1 *"I could easily move the slider to a specific point"*, Q2 *"'I felt as if it was easy to keep the slider activated"*, and Q3 *"The slider felt responsive to my movements"*. At the end, participants were asked to rank the slider control techniques in order of preference, then were asked to discuss why they ranked them in that order and to reflect on their experience.

3.3 Measurements

We measured the overall **task time**, the **time to acquire slider** handle, and the time taken for the slider cursor to first land on the target (**time to target**). We included these other portions of task time because they would give insight into how quickly different slider acquisition techniques allow users to obtain control of the slider handle, and how these may affect the initial *ballistic* phase of cursor movement [26] as they begin moving the slider towards the target.

We captured the final position of the cursor when the slider widget was released, which was then used to calculate the horizontal **error distance** in pixels between the cursor and intended target (for the digit and letter tasks, this was the central of the target). Additionally, we measured **task success** based on whether the final cursor had acquired the intended target; for the shape scaling task, which required a much greater degree of precision (i.e., pixel perfect match), the task was considered a success if the cursor was within a 25 pixel threshold of the intended target position. Finally, we measured task workload using NASA-TLX along with the answers to the agreement statements and participant rankings (as highlighted above). Our experiment data are available via the Zenodo open research data platform [36].

3.4 Implementation

We implemented our experimental software using the Java version of the Leap Motion SDK. Our software ran full-screen on a 27" display (2560 × 1440 px, 65 × 49.1 cm) and bespoke visual feedback was rendered instead of relying on existing OS visualisations or a GUI toolkit, to enable customisation of cursor presentation and appearance. Each task screen showed single horizontal slider bar (Figure 2) that occupied 55% of the screen width, for a real width of 35.75 cm. A circular on-screen cursor was mapped to the centre of the palm of the front-most hand; we coupled cursor movement with palm position, rather than finger position, as this allowed finger movements without affecting cursor position (e.g., for the pinch gestures). Visual feedback was given through the cursor appearance, e.g., changing colour when hovering over the slider handle, and the dwell timer was rendered as a circular progress bar around the cursor edge (Figure 1).

For the air push gesture, we used a moving average of palm distance from the screen and a threshold to check for large forward movements to trigger a push; likewise, an opposing 'pull' would disengage the slider. For the dwell gesture, we used an 800ms dwell timer for slider handle selection. Pilot testing found this to offer an ideal balance giving users sufficient time to prevent unintended selection but without slowing the interaction too much. For the pinch gesture variants, we used the distance between index fingertip and thumb tip to determine when the user was pinching.

3.5 Participants

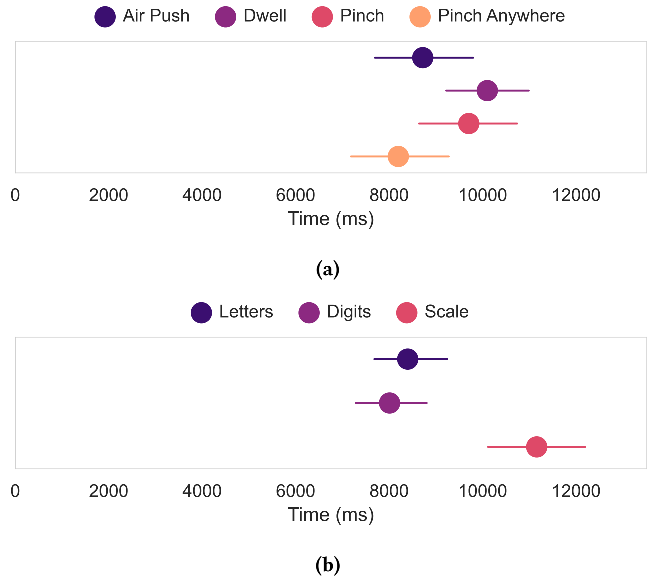
We recruited 15 participants (7 male, 8 female, mean age 27.7 years, SD 5.61 years, 14 right handed and 1 left handed) through institution mailing lists and poster advertising. Participants were compensated £10 for taking part in the study, which lasted for one hour. Ethics approval for this study was provided by our institution ethics committee.

4 RESULTS

In the following section we report the results of the user study and provide statistical analysis on these findings. We used a repeated-measures ANOVA with post-hoc t-tests via the pinguoin python package. For this study an alpha (α) of 0.05 was used.

4.1 Task Performance

4.1.1 Task Time. Mean task time was 9188 ms (SD 3620 ms, 95% CI [8655, 9721] ms), as shown in Figure 3 for each technique (left) and task (right). A repeated-measures ANOVA found a significant main effect of **technique** on time: $F(3,\ 39)\ =\ 4.2, p\ =\ .02,\ \eta_p{}^2 = .25$, and a significant main effect of **task type** on time: $F(2,\ 26)\ =\ 16.2, p\ <\ .001, \eta_p{}^2 = .56$. There was no significant interaction effect: $F(6,\ 78)\ =\ .6, p\ =\ .6$.



(a)

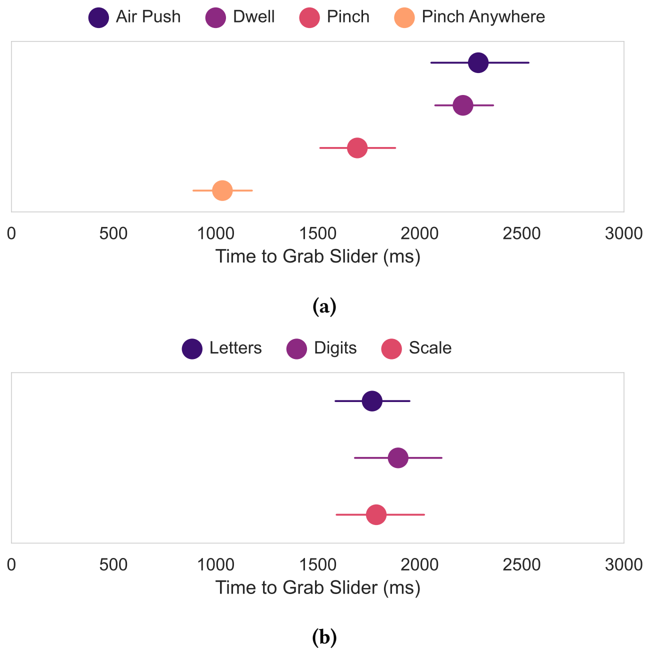
(b)

Figure 3: Mean task time for each interaction technique (a) and task type (b). Error bars show 95% CIs.

Post hoc t-tests found that task time was significantly lower for Air Push than Dwell (t = 3.02, p = .01, d = .67), and significantly lower for Pinch Anywhere than Dwell ($t\ =\ 2.82, p\ =\ .01, d\ =\ .65$) and Pinch ($t\ =\ 2.63, p\ =\ .02, d\ =\ .50$). None of the other pairwise comparisons were significantly different (all $t\ \leq\ 2.2, p\ \geq\ .05$).

Post hoc t-tests found task time was significantly longer for Scale than the Letter ($t\ =\ 6.17, p\ =\ .001, d = 1.1$) and Digit ($t\ =\ 3.84, p\ =\ .002, d = 1.1$) tasks. There was no significant difference between Letter and Digit ($t\ =\ .41, p\ =\ .69$).

4.1.2 Time to Acquire Slider. Mean time to acquire the slider handle was 1816 ms (SD 793 ms, 95% CI [1699, 1933] ms), as shown in Figure 4 for each technique (left) and task (right). A repeated-measures ANOVA found a significant main effect of **technique** on time to acquire slider handle: $F(3, 39)\ =\ 53.8, p\ <\ .001, \eta_p{}^2 = .81$. There was no significant main effect of **task type** ($F(2, 26)\ =\ 1.17, p\ =\ .33$) and there was no significant interaction effect ($F(6, 78)\ =\ 1.74, p\ =\ .12$).



(a)

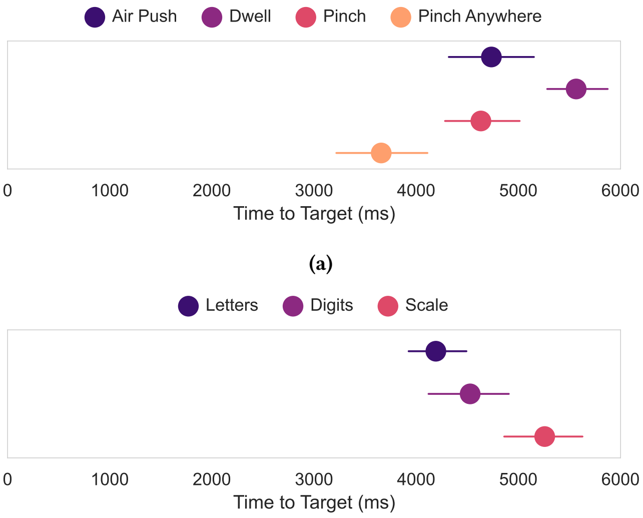
(b)

Figure 4: Mean time to acquire control of the slider handle for each technique (a) and task type (b). Error bars show 95% CIs.

Post hoc t-tests found that time to acquire slider handle was significantly lower for Pinch Anywhere than all other techniques (all $t\ \geq\ 6.42, p\ \leq\ .001, d\ \geq\ 1.89$), and was significantly lower for Pinch than Air Push ($t\ =\ 4.26, p\ =\ .001, d\ =\ 1.17$) and Dwell ($t\ =\ 3.78, p\ =\ .002, d\ =\ 1.27$). There was no significant difference between Air Push and Dwell ($t\ =\ .63, p\ =\ .54$).

4.1.3 Time To Reach Target. Mean time to reach the target was 4659 ms (SD 1488 ms, 95% CI [4440, 4878] ms) as shown in Figure 5 for each technique (left) and task (right). A repeated-measures ANOVA found a significant main effect of **technique** on time to reach target: $F(3,\ 39)\ =\ 22.05, p\ <\ .001,\ \eta_p{}^2 = .63$, and a significant main effect of **task type** on time to reach target: $F(2,\ 26)\ =$

16.6, $p < .001$, $\eta_p{}^2 = .56$. There was no significant interaction: $F(6, 78) = .93, p = .48$.



(a)

(b)

Figure 5: Mean time to acquire target for each interaction technique (a) and task type (b). Error bars show 95% CIs.

Post hoc t-tests found that time to target was significantly lower for Pinch Anywhere than all other techniques (all $t \geq 4.02, p \leq .001, d \geq .96$). Time to target was also significantly higher for Dwell than Pinch ($t = 3.66, p = .003, d = 1.02$) and Air Push ($t = 3.74, p = .002, d = 1.09$). There was no significant difference between Pinch and Air Push ($t = .03, p = .97$).

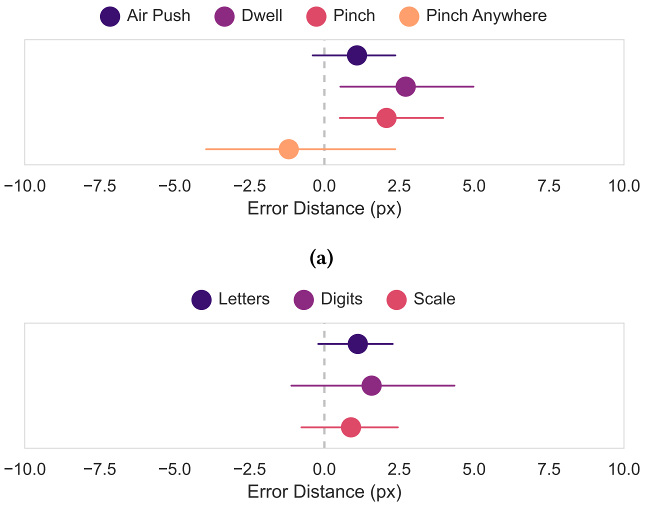
Post hoc t-tests found time to target was significantly longer for Scale than the Letter ($t = 8.3, p = .001, d = 1.21$) and Digit ($t = 3.09, p = .009, d = .77$) tasks. There was no significant difference between Letter and Digit ($t = 1.66, p = .12$).

4.1.4 Error Distance. Mean error distance was 1.19 px/4.7 mm (SD 7.8 px/30.7 mm, 95% CI [0.09, 2.29] px) as shown in Figure 6 for each technique (left) and task (right). A repeated-measures ANOVA found no significant main effect of **technique** ($F(3, 39) = 2.74, p = .06$) and no significant main effect of **task** ($F(2, 26) = .11, p = .86$).

4.1.5 Task Success Rate. Mean task success rate was 96.5% (SD 9 %, 95% CI [95, 98] %), as in Table 1.

Table 1: Task success rate for each technique and task type.

	Digits	**Letters**	**Scale**	***Mean***
Air Push	100%	99.2%	97.7%	98.9%
Dwell	98.5%	95.7%	94.6%	96.3%
Pinch	100%	97.9%	92.7%	96.8%
Pinch Anywhere	95.5%	94.7%	90.5%	93.7%
Mean	98.5%	96.9%	93.9%	96.5%



(a)

(b)

Figure 6: Mean error distance for each interaction technique (a) and task type (b). Error bars show 95% CIs. Note that negative error distance means the slider handle ended before the centre of the target.

4.2 Task-Load Index

Overall TLX score was computed from the NASA-TLX survey responses using the 'raw TLX' method [13]. Mean overall TLX score was 47 out of 100 (SD 15, 95% CI [43.2, 50.8]), as shown in Figure 7. Friedman's test found a significant main effect of **technique** on overall TLX score: $\chi^2 = 8.4, p = .04$. Post hoc Nemenyi comparisons found no significant differences between techniques (all $p \geq .09$).

In addition to overall TLX score, we analysed the effect of technique on the six individual TLX components. Friedman's test found significant main effects of **technique** on mental demand ($\chi^2 = 12.40, p = .006$), effort level ($\chi^2 = 8.76, p = .03$) and frustration ($\chi^2 = 11.10, p = .01$); there was no significant effect on physical load ($\chi^2 = 6.44, p = .09$), temporal demand ($\chi^2 = 4.18, p = .2$) and perceived performance ($\chi^2 = 1.16, p = .8$).

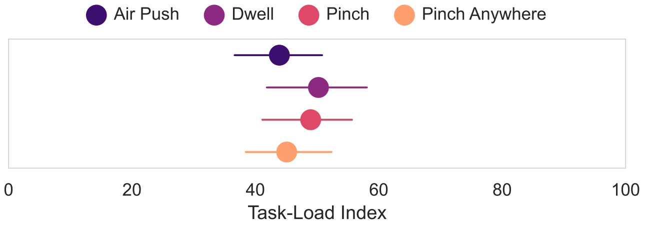


Figure 7: Mean Task-Load Index score for each interaction technique. Error bars show 95% CIs.

Post hoc Nemenyi tests found that mental demand was lower for Air Push compared to Dwell ($p = .02$) and Pinch ($p = .02$). None of the other pairwise comparisons were significantly different (all $p \geq .09$).

Post hoc Nemenyi tests found that frustration was significantly higher for Pinch compared to Air Push ($p = .001$). None of the other pairwise comparisons were significantly different (all $p \geq .06$).

Post hoc Nemenyi tests found no significant differences for perceived effort level (all p ≥ .09).

4.3 Survey Results

Table 2 shows an overview of the Likert scale results for agreement with the statements on slider control. For Q1 (*"I could easily move the slider to a specific point"*), Friedman's test found a significant main effect of technique: $\chi^2 = 11.0, p = .012$. Post hoc Nemenyi tests found no significant differences (all $p \geq .07$). Participants generally agreed that all four techniques enabled them to easily control the slider.

For Q2 (*"I felt as if it was easy to keep the slider activated"*), Friedman's test did not find a significant main effect of technique: $\chi^2 = 6.9, p = .074$. Participants generally agreed that Air Push and Pinch Anywhere enabled them to stay in control of the slider, although responses for Dwell and Pinch were closer to the neutral response.

For Q3 (*"The slider felt responsive to my movements"*), Friedman's test found a significant main effect of technique: $\chi^2 = 9.1, p = .03$. Post hoc Nemenyi tests found no significant differences (all $p \geq .15$). Participants generally agreed that all four techniques were responsive to their gestures.

Figure 8 shows the distribution of preference ranks for the techniques. Friedman's test found a significant difference in the rankings: $\chi^2(3) = 14.6, p = .002$. Post hoc Nemenyi tests found that Pinch Anywhere was ranked significantly more highly than Dwell ($p = .01$) and Pinch ($p = .006$). No other significant differences were found (all $p \geq .19$).

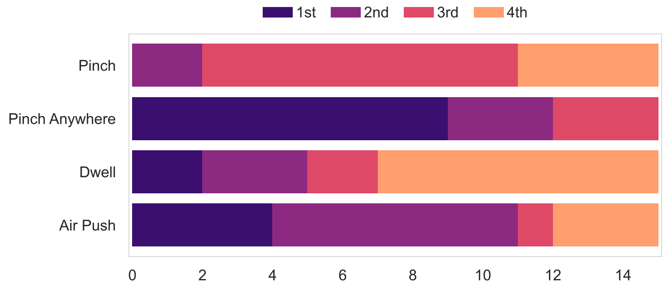


Figure 8: Preference ranking distribution for each interaction technique.

We asked participants to discuss their preferences for the interaction techniques and to reflect on the touchless slider tasks they had just completed. Pinch Anywhere was the most preferred with nine participants ranking it first and nobody ranking it in last place. When asked to explain why, most noted that it felt the easiest to use. Participants could perform the pinch gesture without being over the slider handle ("*it was quick as my hand could be anywhere*" [P15], "*less demanding as I could do it anywhere*" [P10]), which meant they could take control "*without making an effort*" [P12] and "*focus on the task instead of the slider*" [P2]. Perhaps unsurprisingly, nobody ranked the basic Pinch technique first, as it was highly similar to Pinch Anywhere but without the freedom to assume slider control over a wider region.

Air Push was second most preferred, with four participants ranking it first. These four felt the activation method was the simplest to perform (e.g., "*it was easiest to select*" [P14] and "*felt natural*" [P11]). Those who ranked it last seemed to have the opposite experience, suggesting it was unclear how to (dis)engage the handle and that "*it was frustrating [moving] my finger forward and back*" [P6]. Dwell was a divisive technique; five participants ranked it within their two most favourite techniques, although eight ranked it their least favourite. This difference in opinions seems to be driven by whether or not participants felt the dwell period slowed them down: whereas some felt remaining steady "*made it easier to select*" [P4, P8], others found this "*more tiring*" [P10] and "*more time consuming*" [P12, P14].

5 DISCUSSION

5.1 Acquiring Slider Control

It took significantly less time for users to acquire control of the slider handle when using Pinch Anywhere (1033 ms) and Pinch (1694 ms), than when using Air Push (2286 ms) and Dwell (2212 ms), as shown in Figure 4. Pinching seems to be a faster 'mode switch' than the Air Push gesture, as both Pinch and Air Push required users to precisely guide the cursor to the slider handle before they could take control, yet there was almost 600 ms difference between them. Since the time to acquire slider control only includes time taken to target and activate the slider handle, we believe pinching could be a promising complementary alternative to Air Push for activating buttons as well, as this phase of the interaction was similar to button pressing.

It was interesting that there was no significant difference in slider acquisition time between Air Push and Dwell, as we did not subtract the dwell period (800 ms) from the timing data. We observed participants having difficulty enacting the Air Push gesture over the slider widget (similar to our earlier findings [37]), which perhaps explains why the Dwell technique did not take longer for initially acquiring slider control. The dwell period did have a noticeable effect across the entire task duration however, as the dwell period had a compound effect when accidental slider release meant further dwell periods were necessary to regain control. Whilst some participants did not mind the dwell period initially, repeated dwelling caused frustration, which may explain why it was ranked the least preferred by most.

5.2 Pinching 'Anywhere' vs Above the Cursor

Participants largely preferred the Pinch Anywhere technique because it allowed them to take control of the slider from a convenient and comfortable hand position. This minimised initial hand movement as users were not required to move the cursor to the slider and could quickly gain control. This led to users acquiring slider control almost 700 ms faster (on average) than the standard Pinch technique (which was the next fastest). We wondered about the extent to which users took advantage of the freedom to take control

Table 2: Summary of Likert scale responses. Note responses were on a scale from 1 ("Strongly Disagree") to 7 ("Strongly Agree").

	Air Push		Dwell		Pinch		Pinch Anywhere	
Question	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Q1 Move	5.5	1.1	4.9	1.4	4.5	1.5	5.2	1.0
Q2 Activated	5.3	1.0	4.3	1.8	4.0	1.4	4.8	1.4
Q3 Responsive	5.7	0.8	4.7	1.2	4.7	1.2	5.3	1.4

without precisely targeting the cursor. Figure 9 shows the distribution of x-axis coordinates for the cursor, when users gained control of the slider handle (mean start point was at 42% of the slider bar width). We can see that users often took control 'anywhere' within proximity to the slider, with starting points distributed across the full width of the slider bar, and some even grabbing the handle outwith the slider width.

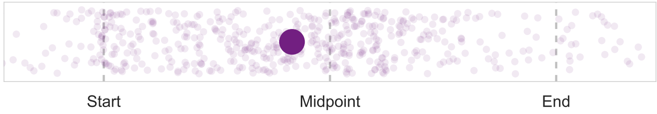


Figure 9: Scatterplot showing distribution of cursor x-axis coordinate when the user first acquired control of the slider handle via the Pinch Anywhere gesture. The x-axis shows the slider start, mid, and end points. The large dot shows mean start position.

Regardless of where the hand is positioned when the user takes control, they still need to make the same hand movements to translate the slider handle. Care must be taken to stay within tracking range of the sensor. For example, if a participant 'pinches' the slider handle with the cursor near the end of the slider bar, further sideways movement will move their hand even further from the centre of the tracking space, away from the "sweetspot" for tracking [11]. This could lead to tracking difficulties and, indeed, seemed to cause some issues with the Pinch Anywhere technique. If a user took control of the slider with the cursor near the end, moving the full length of the slider would take their hand approximately 53.7cm to the right of the sensor centre (see Section 3.4), which is approaching the limitations of Leap Motion tracking range. As Figure 9 shows, some people even took control *beyond* the end of the slider, placing them in a range where sensing quality is not optimal. Designers need to consider the physical size of their touchless UI and widget placement on-screen, with respect to tracking range, if allowing users to acquire control of widgets 'anywhere'. This was not an issue for the other techniques here, as the need to explicitly target the cursor meant the centre of the slider bar was always aligned with the centre of the sensor field of view.

5.3 Enhanced Cursor Targeting and Translation

Despite the user experience benefits and faster acquisition times for techniques that allow control of the slider handle from 'anywhere', precise cursor-based selection methods can still achieve good overall task performance. Air Push may have taken longer to gain slider control and first reach the target region of the slider, yet the overall task time was good and this method had the highest overall task success rate (99% vs the mean of 96.5%). We believe this is largely due to the movement dynamics employed by users when using Air Push. The need for precise targeting, hand stability and consistent hand depth led users to making slower and more controlled hand movements, which meant they were less likely to lose control of the slider (e.g., "*I had to move very slowly to avoid falling off the circle and losing control*" [P3]). This is consistent with Kattinakere's model of mid-air steering [17], which shows that the need to maintain consistent depth leads to slower movement to maintain precision. In contrast, the relaxed constraints of Pinch Anywhere may have led to less precision during the slider tasks, resulting in imprecise initial movements followed by additional corrective movements that are reflected through the overall task time.

A compelling topic for future work could be to explore methods of enhancing cursor-based targeting for continuous slider interactions, to make it easier to move a cursor to the handle and then keep it in place during the slider control gesture. Cursor assistance techniques for other input devices could also be adapted for touchless interfaces. For example, *gravity wells* [27], *sticky widgets* [31] and *pseudo-haptic holes* [18] all modulate the control:display ratio in various ways so that cursors 'snap' and 'stick' to targets as they pass by. Methods like these could aid targeting slider handles and 'stickiness' could help account for cursor instability caused by arm fatigue [15] or incidental motion during the Air Push gesture. Such improvements could also enhance other touchless widget interactions, like targeting and pressing buttons. However, caution is needed to avoid negatively affecting the user's sense of control over the system. Too much assistance can reduce the sense of agency experienced by users [7], and this may well be the case for methods that result in incongruence between a person's hand movements and the corresponding cursor movements on screen [39].

5.4 Pinching Multiple Sliders

Our findings show the usability benefits of reducing the need for precise spatial positioning when taking control of continuous interaction widgets like sliders. The Pinch Anywhere gesture was feasible because pinching is not otherwise a meaningful action within the air push interaction paradigm, so does not interfere with other input operations. This was fine within the context of this experiment where we only had a single slider bar for users to interact with, although there may be situations where a touchless interface has multiple sliders (or widgets of similar class).

In such situations, we would recommend imposing rules of proximity so that the widget nearest to the cursor responds to pinch (or other) activation gestures. Similar approaches have been used in other touchless interfaces. For example, we previously used single-handed finger counting gestures to allow section from up to five buttons (e.g., extending two fingers to select the second target) [9]. To allow selection from more targets, they grouped buttons into sets of five, so that finger counting gestures were applied to whichever group was closest to the cursor. Similar approaches could be used for slider acquisition, still giving users plenty of space for making selections (e.g., as in Figure 10).

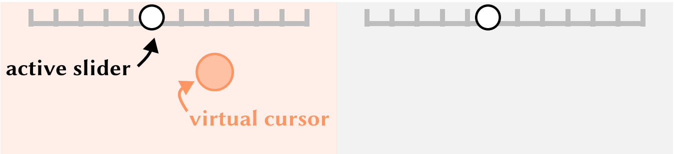


Figure 10: Control gestures like Pinch Anywhere could be used in touchless interfaces with multiple sliders (or similar move-able widgets) by mapping the gesture to whichever widget is closest to the cursor, e.g., the leftmost slider in this example.

6 CONCLUSION

Touchless user interfaces are finding a wider audience across more application domains through Air Push technology, and so it is important to explore ways of improving basic user interface operations to improve usability and user satisfaction. Many touchless user interfaces use the Air Push interaction paradigm, where users control an on-screen cursor and activate interface elements by moving their hand towards the screen to enter a 'pressed' state. Air Push gestures work well for button pressing operations, however may not be ideal for continuous operations, where users need to maintain that pressed state whilst steering to a target (like sliding). We investigated alternative gestures for taking control of a slider using different mode switching techniques that avoid pushing motions. Our findings suggest that pinching gestures offer a promising alternative for continuous interaction with widgets like sliders, as pinching offers two clearly defined states and avoids forward hand movements during activation. Since pinching is not a meaningful action in the air push interaction paradigm, it can even be used for immediate access to move-able widgets like sliders (i.e., our Pinch Anywhere method). Our formative look at touchless sliding shows the potential benefits of introducing new gestures to facilitate continuous interactions, taking advantage of the capabilities of mid-air hand tracking to sense more expressive gestures.

REFERENCES

[1] Johnny Accot and Shumin Zhai. 1997. Beyond Fitts' Law: Models for Trajectory-Based HCI Tasks. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems* (Atlanta, Georgia, USA) *(CHI '97)*. Association for Computing Machinery, New York, NY, USA, 295–302. https://doi.org/10.1145/258549.258760

[2] Stephen Brewster. 2002. Overcoming the lack of screen space on mobile computers. *Personal and Ubiquitous computing* 6, 3 (2002), 188–205. https://doi.org/10.1007/s007790200019

[3] Marcus Carter, Eduardo Velloso, John Downs, Abigail Sellen, Kenton O'Hara, and Frank Vetere. 2016. PathSync: Multi-User Gestural Interaction with Touchless Rhythmic Path Mimicry. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 3415–3427. https://doi.org/10.1145/2858036.2858284

[4] Christopher Clarke, Alessio Bellino, Augusto Esteves, Eduardo Velloso, and Hans Gellersen. 2016. TraceMatch: A Computer Vision Technique for User Input by Tracing of Animated Controls. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (Heidelberg, Germany) *(UbiComp '16)*. Association for Computing Machinery, New York, NY, USA, 298–303. https://doi.org/10.1145/2971648.2971714

[5] Christopher Clarke and Hans Gellersen. 2017. MatchPoint: Spontaneous Spatial Coupling of Body Movement for Touchless Pointing. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (Québec City, QC, Canada) *(UIST '17)*. Association for Computing Machinery, New York, NY, USA, 179–192. https://doi.org/10.1145/3126594.3126626

[6] Patricia Ivette Cornelio Martinez, Silvana De Pirro, Chi Thanh Vi, and Sriram Subramanian. 2017. Agency in Mid-air Interfaces. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. Association for Computing Machinery, New York, NY, USA, 2426–2439. https://doi.org/10.1145/3025453.3025457

[7] David Coyle, James Moore, Per Ola Kristensson, Paul C. Fletcher, and Alan F. Blackwell. 2012. I did that! Measuring Users' Experience of Agency in their own Actions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '12*. ACM Press, 2025–2034. http://dl.acm.org/citation.cfm?id=2208350

[8] Sean Cronin, Euan Freeman, and Gavin Doherty. 2022. Investigating Clutching Interactions for Touchless Medical Imaging Systems. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. Association for Computing Machinery, New York, NY, USA, 14 pages. https://doi.org/10.1145/3491102.3517512

[9] Euan Freeman, Stephen Brewster, and Vuokko Lantz. 2014. Tactile Feedback for Above-Device Gesture Interfaces: Adding Touch to Touchless Interactions. In *Proceedings of the International Conference on Multimodal Interaction - ICMI '14*. ACM, 419–426. https://doi.org/10.1145/2663204.2663280

[10] Euan Freeman, Stephen Brewster, and Vuokko Lantz. 2016. Do That, There: An Interaction Technique for Addressing In-Air Gesture Systems. In *Proceedings of the 34th Annual ACM Conference on Human Factors in Computing Systems - CHI '16*. Association for Computing Machinery, 2319–2331. https://doi.org/10.1145/2858036.2858308

[11] Euan Freeman, Dong-Bach Vo, and Stephen Brewster. 2019. HaptiGlow: Helping Users Position their Hands for Better Mid-Air Gestures and Ultrasound Haptic Feedback. In *2019 IEEE World Haptics Conference (WHC)*. IEEE, 289–294. https://doi.org/10.1109/WHC.2019.8816092

[12] Charles P. Gerba, Adam L. Wuollet, Peter Raisanen, and Gerardo U. Lopez. 2016. Bacterial contamination of computer touch screens. *American Journal of Infection Control* 44, 3 (2016), 358–360. https://doi.org/10.1016/j.ajic.2015.10.013

[13] Sandra G. Hart and Lowell E. Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In *Human Mental Workload*, Peter A. Hancock and Najmedin Meshkati (Eds.). Advances in Psychology, Vol. 52. North-Holland, 139–183. https://doi.org/10.1016/S0166-4115(08)62386-9

[14] Khalad Hasan, David Ahlström, and Pourang Irani. 2013. Ad-Binning: Leveraging around Device Space for Storing, Browsing and Retrieving Mobile Device Content. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. Association for Computing Machinery, New York, NY, USA, 899–908. https://doi.org/10.1145/2470654.2466115

[15] Juan David Hincapié-Ramos, Xiang Guo, Paymahn Moghadasian, and Pourang Irani. 2014. Consumed Endurance: A Metric to Quantify Arm Fatigue of Mid-Air Interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 1063–1072. https://doi.org/10.1145/2556288.2557130

[16] Intel. 2021. Intel RealSense Touchless Control Software. https://www.intelrealsense.com/introduction-to-intel-realsense-touchless-control-software Last Accessed 24/04/2022.

[17] Raghavendra S. Kattinakere, Tovi Grossman, and Sriram Subramanian. 2007. Modeling Steering within Above-the-Surface Interaction Layers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '07)*. Association for Computing Machinery, New York, NY, USA, 317–326. https://doi.org/10.1145/1240624.1240678

[18] Anatole Lécuyer, Jean-Marie Burkhardt, and Laurent Etienne. 2004. Feeling bumps and holes without a haptic interface: the perception of pseudo-haptic textures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '04*, Vol. 6. 239–246. http://dl.acm.org/citation.cfm?id=985723

[19] Hannah Limerick. 2020. Call to interact: communicating interactivity and affordances for contactless gesture controlled public displays. In *Proceedings of the 9TH ACM International Symposium on Pervasive Displays (PerDis '20)*. Association for Computing Machinery, New York, NY, USA, 63–70. https://doi.org/10.1145/3393712.3395338

[20] Ultraleap Ltd. 2021. Camera Placement. https://docs.ultraleap.com/touchfree-user-manual/camera-placement.html Last Accessed 25/04/2022.

[21] Ultraleap Ltd. 2021. TouchFree For Kiosks and Digital Signage. https://www.ultraleap.com/enterprise/touchless-experiences/touchfree-solution Last Accessed 25/04/2022.

[22] Ultraleap Ltd. 2021. Touchless Interactive Kiosks: The Air Push Gesture. https://www.ultraleap.com/company/news/blog/touchless-interactions/ Last Accessed 25/04/2022.

[23] Ville Mäkelä, Jonas Winter, Jasmin Schwab, Michael Koch, and Florian Alt. 2022. Pandemic Displays: Considering Hygiene on Public Touchscreens in the Post-Pandemic Era. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. Association for Computing Machinery, New York, NY, USA, 12 pages. https://doi.org/10.1145/3491102.3501937

[24] Sylvain Malacria, Eric Lecolinet, and Yves Guiard. 2010. Clutch-Free Panning and Integrated Pan-Zoom Control on Touch-Sensitive Surfaces: The Cyclostar Approach. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Atlanta, Georgia, USA) *(CHI '10)*. Association for Computing Machinery, New York, NY, USA, 2615–2624. https://doi.org/10.1145/1753326.1753724

[25] Regan L. Mandryk and Calvin Lough. 2011. The Effects of Intended Use on Target Acquisition. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, BC, Canada) *(CHI '11)*. Association for Computing Machinery, New York, NY, USA, 1649–1652. https://doi.org/10.1145/1978942.1979182

[26] David E Meyer, Richard A Abrams, Sylvan Kornblum, Charles E Wright, and JE Keith Smith. 1988. Optimality in human motor performance: ideal control of rapid aimed movements. *Psychological review* 95, 3 (1988).

[27] Ian Oakley, Marilyn Rose McGee, Stephen Brewster, and Philip Gray. 2000. Putting the Feel in 'Look and Feel'. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (The Hague, The Netherlands) *(CHI '00)*. Association for Computing Machinery, New York, NY, USA, 415–422. https://doi.org/10.1145/332040.332467

[28] Jennifer Sarah Pearson, Gavin Bailey, Simon Robinson, Matt Jones, Tom Owen, Chi Zhang, Thomas Reitmaier, Cameron Steer, Anna Carter, Deepak Sahoo, and Dani Kalarikalayil Raju. 2022. Can't Touch This: Rethinking Public Technology in a COVID-19 Era. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. https://doi.org/10.1145/3491102.3501980

[29] Henning Pohl and Roderick Murray-Smith. 2013. Focused and Casual Interactions: Allowing Users to Vary Their Level of Engagement. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. Association for Computing Machinery, New York, NY, USA, 2223–2232. https://doi.org/10.1145/2470654.2481307

[30] Ismo Rakkolainen, Euan Freeman, Antti Sand, Roope Raisamo, and Stephen Brewster. 2020. A Survey of Mid-Air Ultrasound Haptics and Its Applications. *IEEE Transactions on Haptics* 14 (2020), 2–19. Issue 1. https://doi.org/10.1109/TOH.2020.3018754

[31] Malcolm E. Rodgers, Regan L. Mandryk, and Kori M. Inkpen. 2006. Smart sticky widgets: Pseudo-haptic enhancements for multi-monitor displays. In *Proceedings of Smart Graphics*. 194–205. http://www.springerlink.com/index/U526825K74855577.pdf

[32] Srinath Sridhar, Anna Maria Feit, Christian Theobalt, and Antti Oulasvirta. 2015. Investigating the Dexterity of Multi-Finger Input for Mid-Air Text Entry. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 3643–3652. https://doi.org/10.1145/2702123.2702136

[33] Hemant Bhaskar Surale, Fabrice Matulic, and Daniel Vogel. 2019. Experimental Analysis of Barehand Mid-Air Mode-Switching Techniques in Virtual Reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3290605.3300426

[34] Eduardo Velloso, Marcus Carter, Joshua Newn, Augusto Esteves, Christopher Clarke, and Hans Gellersen. 2017. Motion Correlation: Selecting Objects by Matching Their Movement. *ACM Trans. Comput.-Hum. Interact.* 24, 3, Article 22 (apr 2017), 35 pages. https://doi.org/10.1145/3064937

[35] Dong-Bach Vo and Stephen Brewster. 2015. Touching the Invisible: Localizing Ultrasonic Haptic Cues. In *Proceedings of World Haptics Conference 2015 - WHC '15*. IEEE, 368 – 373. https://doi.org/10.1109/WHC.2015.7177740

[36] Kieran Waugh, Mark McGill, and Euan Freeman. 2022. *User Study Data for "Push or Pinch? Exploring Slider Control Gestures for Touchless User Interfaces"*. https://doi.org/10.5281/zenodo.6779013

[37] Kieran Waugh and Judy Robertson. 2021. Don't Touch Me! A Comparison of Usability on Touch and Non-touch Inputs. In *Human-Computer Interaction – INTERACT 2021*, Carmelo Ardito, Rosa Lanzilotti, Alessio Malizia, Helen Petrie, Antonio Piccinno, Giuseppe Desolda, and Kori Inkpen (Eds.). Springer International Publishing, Cham, 400–404.

[38] Andrew D. Wilson. 2006. Robust Computer Vision-Based Detection of Pinching for One and Two-Handed Gesture Input. In *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology* (Montreux, Switzerland) *(UIST '06)*. Association for Computing Machinery, New York, NY, USA, 255–258. https://doi.org/10.1145/1166253.1166292

[39] Regine Zopf, Vince Polito, and James Moore. 2018. Revisiting the link between body and agency: Visual movement congruency enhances intentional binding but is not body-specific. *Scientific Reports* 8, 1 (2018), Article 196. https://doi.org/10.1038/s41598-017-18492-7