Pointing All Around You: Selection Performance of Mouse and Ray-Cast Pointing in Full-Coverage Displays

Julian Petford
University of St Andrews
St Andrews, UK
jp438@st-andrews.ac.uk

Miguel A. Nacenta
University of St Andrews
St Andrews, UK
mans@st-andrews.ac.uk

Carl Gutwin
University of Saskatchewan
Saskatchewan, Canada
gutwin@cs.usask.ca

ABSTRACT

As display environments become larger and more diverse – now often encompassing multiple walls and room surfaces – it is becoming more common that users must find and manipulate digital artifacts not directly in front of them. There is little understanding, however, about what techniques and devices are best for carrying out basic operations above, behind, or to the side of the user. We conducted an empirical study comparing two main techniques that are suitable for full-coverage display environments: mouse-based pointing, and ray-cast 'laser' pointing. Participants completed search and pointing tasks on the walls and ceiling, and we measured completion time, path lengths and perceived effort. Our study showed a strong interaction between performance and target location: when the target position was not known a priori the mouse was fastest for targets on the front wall, but ray-casting was faster for targets behind the user. Our findings provide new empirical evidence that can help designers choose pointing techniques for full-coverage spaces.

ACM Classification Keywords

H.5.2. Information Interfaces and Presentation: Input Devices and strategies; H.5.1. Information Interfaces and Presentation: Artificial, augmented and virtual realities

Author Keywords

Pointing; Targeting; Immersive Spaces; Full-coverage Displays; Multi-display environments; Laser Pointing

INTRODUCTION

Displays and projectors are becoming cheaper and easier to deploy for daily computing. It is not uncommon to see offices and working environments where a large proportion of the space in front of the user is covered with displays or projectable spaces. A likely scenario for future work and home environments is one where all or most of the surfaces inside a room can display digital information [41]. In fact, everyday spaces that are covered fully or almost fully by displays are starting to be well supported by toolkits (e.g., RoomAlive [54], ASPECTA [39]) that enable transforming regular rooms into spaces where most of the surrounding physical space can be used to display or overlay digital information. The potential benefits of these spaces include extending the digital workspace into the physical world, augmenting world objects with digital information and capabilities, and taking advantage of natural human abilities such as spatial memory, use of landmarks, and the sensitivities of different areas of the visual field [20, 39].

Many of the applications that take advantage of Full-Coverage Displays (FCDs) will require the ability to interact with objects that appear in the environment. For example, a sticky-note application where the user can distribute digital notes to any location in their office will require selecting notes, deleting notes and moving notes from one place to another (e.g., from the desk to the entrance).

Although input is a critical element in the design and possible success of these environments, and despite the breadth of research already carried out on different input techniques, we still have little information about the design of input in this kind of environment. For example, would a pointing gesture be faster than a mouse-based interaction? Furthermore, what will be the characteristics of targeting movements when the possible targets can be in any location such as behind the user or on the ceiling?

In this research we seek to better understand the performance and comfort of pointing interactions in full-coverage display environments. We carried out an experiment in a five-surface display space where we compared a mouse-based interaction technique with a ray-casting pointing technique. For a set of pointing tasks (where participants pointed to a known target) and search+pointing tasks (where the participant had to first find the target before pointing to it) that spanned the full range of angles in a room, we measured completion time, path lengths, perceived effort, and technique preferences. The main contributions of this paper are the results derived from our study, specifically:

- Mouse pointing is fastest for front wall tasks while ray-cast pointing is faster for every other surface (if the target position is not known a priori).
- Ray-casting interaction supports better overlap of search and targeting tasks.

CHI 2018, April 21–26, 2018, Montreal, QC, Canada

ISBN 978-1-4503-5620-6/18/04…$15.00
DOI: https://doi.org/10.1145/3173574.3174107

- Different walls result in different pointing times, with interesting asymmetries.
- Models based on angular measures explain pointing performance better than their linear counterparts.

We analyze and discuss these results and their implications for the design of input in environments with full-coverage displays. Our work contributes a new and more nuanced understanding of how pointing and visual search performs in large immersive spaces, and our results can help designers choose appropriate techniques for different kinds of tasks in these novel environments.

RELATED WORK

Pointing is a pervasive activity in most graphical interfaces. It is therefore not surprising that research in pointing, which encompasses input device comparisons (e.g., [27]) and the modeling of time and accuracy for pointing times (e.g., [11, 13, 26, 46]), is a dominant stream of HCI research. Of particular interest for spaces with full-coverage Displays is previous research on the modeling and performance of input options in large displays, multi-display environments (MDEs), and immersive environments. We also review existing work on targeting techniques that use special sensors such as gaze trackers, and relevant work in modeling angular pointing gestures.

Pointing in MDEs and Large Displays

The mouse has been the dominant input device for PCs and laptops, with direct touch or multi-touch also becoming important in approximately the last decade. However, researchers identified early that mouse and touch might not be ideal when trying to control interfaces that have large or multiple displays. One obvious solution is the use of ray-casting techniques in their multiple variants such as laser pointers [7, 29, 35, 37], the detection of fingers or hands for pointing [45, 52], or the seamless combination of indirect input with direct input or pen input [16, 38]. Ray-casting techniques are easily understood by users and provide a convenient solution to the problem of reach. However, they have also been shown to be susceptible to tremor and parallax [23, 29]. Performance with this kind of technique has been modeled but, to our knowledge, only in environments with large front displays and not when the displays surround the user (e.g., [23, 24]).

An alternative solution is to provide mouse input, which is known to have better throughput than in-air interaction [10] and avoids, to a large extent, issues of precision and parallax associated with angular control. However, plain mouse input for spaces that are not flat requires mappings that are aware of the physical space. Examples of these kind of mappings have been proposed in the literature [32, 33, 53, 55] and have shown performance improvements for MDEs but, to our knowledge, have not been tested in spaces that surround the user.

Pointing in Virtual Reality and Augmented Reality

Pointing and object selection has also received substantial attention in the virtual reality community [3]. Many of the findings are analogous to those on 2D surfaces, for example, smaller objects are generally harder to target and it takes longer to target objects that are at an angle [49]. The results in the augmented reality domain show also that Fitts' models apply (to different degrees) for pointing to objects in the real environment through small screens [43, 44]. However, we are unaware of any studies that have looked at the space surrounding the user, in virtual, augmented reality or elsewhere. Most studies are constrained to selection tasks in the area in front of the participants [2], are limited to short angles due to the display or environment (e.g., small volumetric displays [19], non-immersive fish tank VR [49]), or chose to only investigate narrow angular distances even when targeting tasks took place in the physical environments [44]. Therefore, we do not know much yet about the performance of targeting motions at large angles that surround the user.

Pointing with advanced sensors

Other researchers have proposed techniques that take advantage of other potentially useful information from the user, the environment, or mobile devices such as head orientation [50], gaze location [47], foot taps [18], or the orientation of a smart device [56, 57].

Pointing Performance in MDEs

Since 1954 Fitts' Law [15] and its various forms (e.g., [25, 46]) have been widely used for the prediction of performance in pointing tasks and to assess the efficiency of different targeting techniques and devices. By fitting one of these variants to the performance of real users we can estimate a small set of parameters that predict targeting time for a wide range of target sizes and distances. Generally these models have been applied to examine performance of input devices such as the computer mouse (e.g., [14]), pointers (e.g., [10]), and direct touch (e.g., [15, 17]) for small screens. Other models of performance exist based on, for example, the decomposition of the targeting tasks into two phases [28], but Fitts' law models and their variants are still dominant in HCI since they are adequately descriptive and simple.

More recently, the increased availability of large wall screens has led to research in the comparison and modeling of techniques for large display input [34, 24, 23, 4]. It is important to highlight that large displays have implications for input that go beyond simple size. For example, targeting in large flat displays means that targets of the same size and shape cover different angles of the visual field depending on whether they are in front of the user or in the periphery [23], and targets might become too small to be pointed at if far away [21]. As a consequence, models that fit a small locality directly in front of the user might not generalize to targeting further away. Some work has started to address this issue by modeling angle instead of linear distance [23, 24] as well as by providing models that are more flexible in how they model gain [46]. However, considering targeting in displays that surround the user might introduce additional asymmetries that cannot be captured with the relatively homogeneous current models. For example, we know that different tasks require different muscle groups [12, 40] and therefore pointing left and right may yield different performances [48].

EXPERIMENT

In this section we describe the design of an experiment that we carried out to learn about the characteristics of searching and targeting tasks in full-coverage Display (FCD) environments. More specifically, we were mainly interested in differences in performance between techniques, in how the location of targets affects targeting and search time, in which factors affect performance, and in how subjective workload is affected by different techniques.

Techniques

Of the four types of input typically used for pointing in digital environments (direct touch, radar views, mouse-based and ray-casting [30, 31]) we selected representative techniques for the last two. Direct touch is not represented in the experiment because it is not practical in a room environment (people are not likely to have direct reach to all locations in a room, and physical access will take too much time). Radar views (e.g., [6]) were also excluded because, in most cases, they defeat the purpose of FCDs in the first place. We chose to use mouse control due to its ability to work at a distance, its common usage by the average computer user and its subsequent ability to be used as a baseline for computer-based pointing interactions. From the possible mouse-based options we chose Perspective Cursor [33] (detailed in the next section) because it is the state of the art and its variants (including Ubiquitous Cursor [55]) have been shown multiple times to be faster (although in different kinds of environments). From the ray-casting options we chose an absolute pointer controlled with a solid object. This technique is simple, pervasive and consistent with a large number of previous implementations and evaluations of 'laser pointers' (e.g., [29, 35, 37]). We chose not to use hybrid pointing techniques (e.g., [16]) and techniques that require gaze-tracking, in order to focus on the core differences between mouse-based and pointer-based approaches, which will be the most common techniques used for full-coverage environments.

Apparatus and Technique Implementation

The experiment took place in a purpose-built space with 2.05 meter by 3.25 meter sides and a 2.2 meter-high ceiling. The four walls and the ceiling were off-white projection surfaces. Three walls and the ceiling were projected from a semi-spherical projector with two 4,100 lumens lamps. The remaining wall (the front wall, which is one of the narrow walls) was projected from a separate Sony VPL-FH35, 5,200 lumens projector. The projection mapping and the experimental software were implemented using the ASPECTA Toolkit [39]. A diagram of the layout of the experimental environment can be seen in Figure 1.

Participants sat on a fully rotating chair in the middle of the room, wearing a pair of over-ear headphones with markers tracked by a set of 6 OptiTrack S250:E cameras. OptiTrack markers were also used to track the pointer device for the ray-casting technique. For the mouse-based technique, participants used a lap tray. To avoid a possible confound due to the click action affecting accuracy (the Heisenberg Effect [9]), clicking for the ray-casting technique was done through the mouse button held in the non-dominant hand out of the tray.

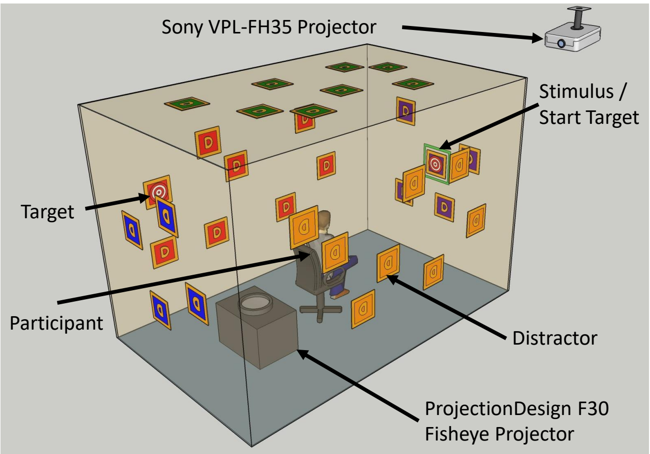


Figure 1. The layout of the experimental environment for an example targeting task. The square objects in the walls are possible locations for distractors, and the initial location for the target (which is also where the initial stimulus appeared) is always in front of the participant. The exact possible locations of targets can be seen in Figures 4 and 5.

The ray-casting technique was implemented by intersecting a virtual line from the pointer object into the walls, which determined the location of the cursor. The Perspective Cursor implementation follows the description provided by Nacenta et al. [33] in its use of head-tracking to enable cursor movement to behave according to the user's field of view[1]. Perspective Cursor helps pointing especially for the ceiling (a surface with no clearly-defined x or y axis orientation) because the cursor moves according to the current frame of reference of the user, rather than a fixed arbitrary X-Y mapping determined by the system.

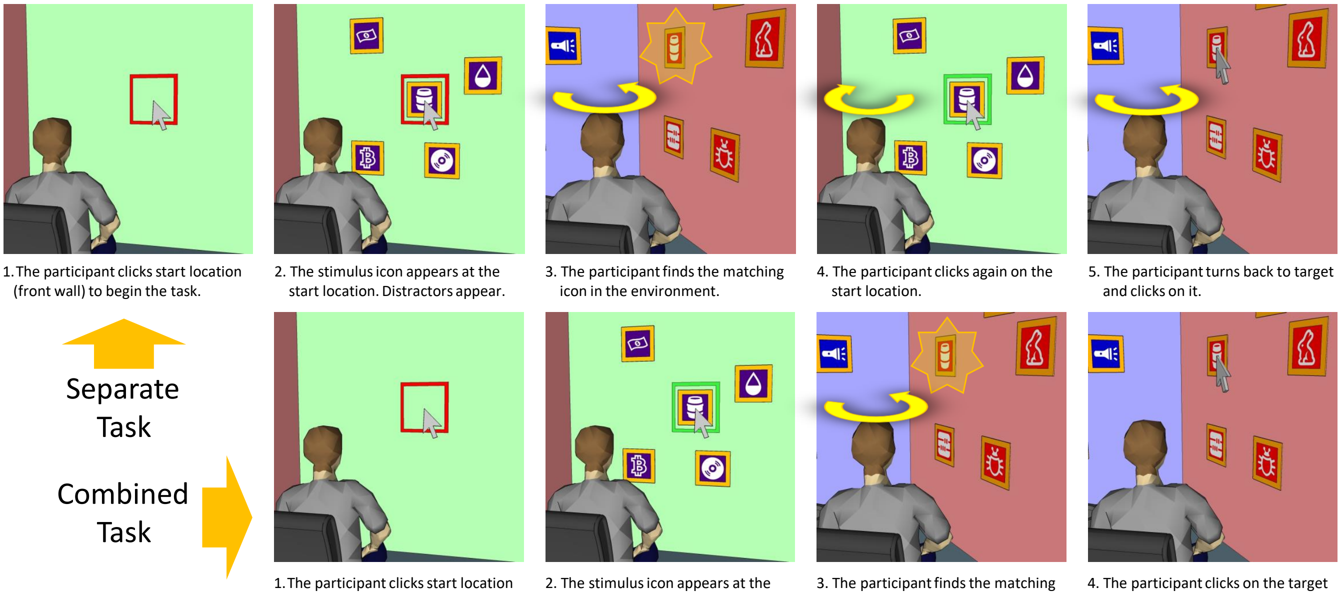
Cursor acceleration was disabled and the VicTsing D-16 mouse was set to 800 CPI. Every pixel of movement translated to 0.05° of cursor movement within the room environment. The lap tray was 38.5cm by 27.5cm in size and moving the mouse directly across it from left to right translated to >360° of cursor movement around the room.

Tasks

The experiment consisted of two phases illustrated in Figure 2. In the first phase participants performed what we call the *separate* task, which consists two separate subtasks, *search+homing* and *targeting*, one after another. A stimulus (a graphical icon from a set of a 102 curated black and white icons) appeared in the middle of the front wall, in front of the participant (the *start location*). After clicking on this stimulus, the participants had to find that same stimulus elsewhere in the room, in one of 20 possible locations distributed over the five projected walls and among a set of 31 additional distractors (8 icons for the long walls and ceiling and 4 for the front

---

[1] For convenience: Perspective Cursor uses the input of the mouse to modify the angle of the cursor with respect to its current position. Vertical movements of the mouse make the cursor move along the meridians of an invisible virtual sphere centered around the user's head, and horizontal movements of the mouse move along the parallels. If the mouse is not moved, the cursor stays in the same position regardless of changes in head pose. The virtual sphere changes position and orientation with the head of the user.



1. The participant clicks start location (front wall) to begin the task.
2. The stimulus icon appears at the start location. Distractors appear.
3. The participant finds the matching icon in the environment.
4. The participant clicks again on the start location.
5. The participant turns back to target and clicks on it.

1. The participant clicks start location (front wall) to begin the task.
2. The stimulus icon appears at the start location. Distractors appear.
3. The participant finds the matching icon in the environment.
4. The participant clicks on the target icon.

Figure 2. Sequence of actions required for the separate task (top row) and combined Task (bottom row). The procedures are identical for both Mouse and Ray-cast interaction.

and back walls with one icon being the target, not a distractor). When the participant found the location of the stimulus, they had to click the initial stimulus at the start location (the "homing" part), which completed the *search+homing* subtask and started the targeting subtask. The final step was then to click on the location of the previously found room location of the stimulus, concluding the targeting subtask. For each trial of each subtask we measured completion time and recorded traces of the cursor movement on the space.

The second phase started after the participant had completed all the repetitions for all target locations of the separate task for a given condition. In the second phase participants performed an identical task except that the two subtasks were performed simultaneously (i.e., in the *combined* task there was no requirement to finish search before starting to point).

The *targeting* subtask of the *separate* task represents situations in which users target to a location known in advance (e.g., because they have done it before). The *combined* subtask represents situations in which users do not know yet where the target is (e.g., because it is the first time that they want to reach that target, or the target might have moved). We added the *search+homing* subtask for experimental design reasons, not because we expected searching to be different across techniques (in principle, search processes are independent of input). Specifically, we wanted to preserve the symmetry between the *separate* and the *combined* tasks in the two phases so that we could observe if parallelism was taking place in the combined task. Moreover, in the second part of the *separate* task the participants had to be informed of the location of the target in advance of each targeting action anyway. Therefore we decided to keep *search+homing* as a subtask for which we collected measures. The sum of the *search+homing* and *targeting* subtask times accounts for the full duration of the *separate* subtask.

The location of targets was different for every trial to preclude memory effects. Clicks which missed the target did not move the experiment forward, forcing the participant to complete a successful task for every target location, under all conditions. We chose to include distractors to help simulate a room environment where it is unlikely that the surrounding environment would be empty except for the object being searched for (a full-coverage display will likely be used for multiple applications simultaneously). Additionally, search-only tasks without distractors are a specific instance of search that might not take place that often in real environments. Therefore we chose a task with a moderate number of distractors as a reasonable initial measuring point. Targets, stimulus and distractors were all squares with 14.3cm sides. We chose this size to balance the need to place many of these objects in different positions in a grid within the room with the needs to make the target icons recognizable and not too small to show non-linear pointing effects due to tremor and other phenomena.

Participants and Procedure

We recruited 24 right-handed participants (12 female, 18 to 34 in age with a mean age of 25.8) from the local university through a newsletter. We dropped the data of an additional three participants; one due to indisposition halfway through the experiment, one because of an experimenter error in setting up the trials, and a last one who did not know how to operate a computer mouse.

The experiment started with the participants providing written consent. In a demonstration phase, the experimenter showed the participants how to complete 20 tasks, five in each of the combinations of task type (Separate, Combined) and technique

(Ray-cast, Mouse). Participants then performed the same tasks on their own as training.

The core of the experiment consisted of 60 trials (3 repetitions on each possible target) of the separate task for each of the two pointing techniques, and then an additional 60 trials of the combined task for each pointing technique, for a total of 240 trials. The order of the techniques was counterbalanced across participants. After each block of 60 trials, participants filled in a NASA TLX questionnaire about perceived workload.

Experimental Questions

The experiment was designed to characterize targeting performance in full-coverage displays. More specifically, we designed the experiment to answer the following research questions:

- *Which interaction technique is fastest for targeting, and where?*
- *Which technique is fastest for search+homing?*
- *Which technique is fastest for the combined task and where?*
- *What are the main spatial asymmetries?*
- *Which factors predict targeting performance in FCDs?*
- *What are the differences between techniques in subjective workload ratings?*

ANALYSIS CONVENTIONS

Analysis of our experimental data is based on ANOVA analyses and regression analyses for quantitative scale data and Wilcoxon's Signed-rank non-parametric test for the ordinal subjective responses of the NASA TLX questionnaires. ANOVA tests that did not comply with the sphericity assumptions were adjusted with a Greenhouse-Geisser correction. This shows up as non-integer degrees of freedom in reports of the F values. Time measurements are log-transformed before each ANOVA analysis to comply with normality assumptions. Multiple repetitions per cell per participant were aggregated using the median. Charts show averages across participants of the median completion times for a given technique-location combination.

In most of the analyses below we use the *target location* factor. This factor provides a unique identity to every possible location of a target (i.e., 20 different identifiers), which therefore includes information about both the target's coordinate within the wall and the wall identity. However, in cases where we want to compare performance across walls this information is separated into a *wall* factor and a *target coordinate* factor.

RESULTS

We organize our findings according to the experimental questions listed above.

Which Technique is Fastest for Targeting, and Where?

To answer this question we carried out a factorial RM-ANOVA on the log-transformed completion time of the targeting subtask, with target position and technique as main factors. We found a main effect of pointing technique ($F_{1,23} = 11.69$, $p < 0.003, \eta^2 = 0.055$) and, as expected, of target location ($F_{6.82,156.78} = 238.33$, $p < 0.001$, $\eta^2 = 0.787$). On average Ray-cast took 13.67% less time ($\mu_{ray} = 2.185s < \mu_{mouse} = 2.531s$). Fig 3.B shows average targeting times by target angle, surface, and technique.

Wall	F(1,23)	p	η^2	Faster Technique
Front	1.45	0.24	0.01	Mouse
Left	0.97	0.34	0.01	Ray-Cast
Right	8.58	<0.01	0.07	Ray-Cast
Back	91.25	<0.001	0.03	Ray Cast
Ceiling	4.41	<0.05	0.03	Ray-Cast

Table 1. Fastest technique per wall for the targeting subtask.

Because the interaction between pointing technique and target location was significant ($F_{19,437} = 10.82$, $p < 0.001$, $\eta^2 = 0.124$), we further explored which targets were more easily reached with the different techniques. Figure 4 suggests that targets in the front wall or at short angles from it show an advantage for Mouse, whereas locations behind the participant are much more favorable for Ray-cast. Further post-hoc analyses per wall reveal that Ray-Casting was significantly faster on the right, back and ceiling surfaces, while the advantage of Mouse in the front wall is non-significant (details of the analysis are in Table 1).

Which Technique is Fastest for Search+Homing?

For this analysis we analyzed the completion time data from the search+homing subtask. We ran a factorial RM-ANOVA with target location and pointing technique as main factors. As expected, there is a main effect of target location on search time ($F_{7.56,173.81} = 172.39$, $p < 0.001$, $\eta^2 = 0.793$), as well as a main effect of technique ($F_{1,23} = 37.37$, $p < 0.001$, $\eta^2 = 0.081$); the interaction was also significant ($F_{8.09,186.01} = 7.32$, $p < 0.001$, $\eta^2 = 0.092$). Figures 3.A shows that search+homing times are generally larger for Ray-cast (Mouse times were on average 19.42% shorter across all targets).

Although it might seem surprising that the interaction technique had an effect on a search task, this can be explained as a consequence of the task design which is, nevertheless, relevant for the design of full-coverage Display interfaces. The search+homing subtask required participants to click on the start point (where the item to find was displayed), find the object visually, and then click back on the original location (homing). Participants were able to look around without moving the cursor when they were using the mouse. This is, however, not possible with the Ray-cast technique because the directional pointer cannot be 'parked' as the mouse can. The pointer in the participant's hand naturally moves around when they move their head and/or body to search for an item around, forcing them to re-target the original position after having found the target, and resulting in longer recorded times.

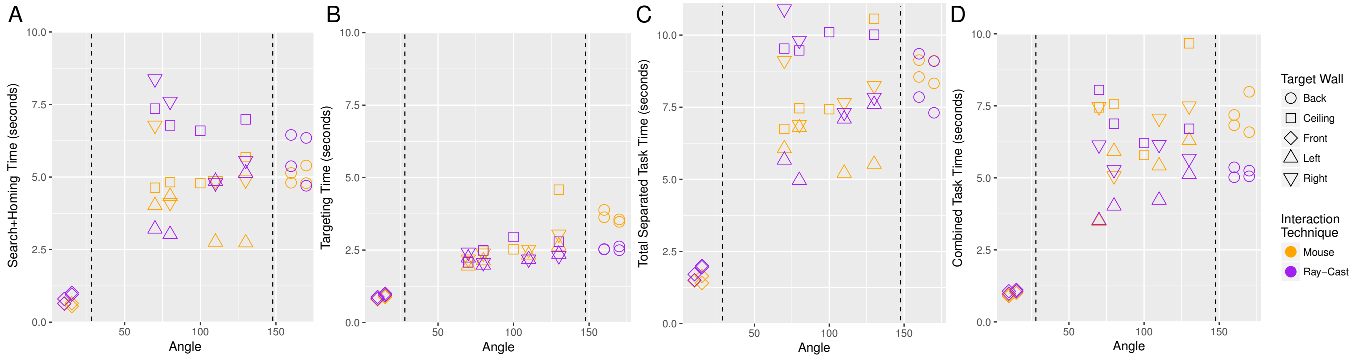


Figure 3. Mean completion times, per surface (shape) and technique (color) of the tasks: A) search+homing subtask, B) targeting subtask, C) arithmetic addition of A and B, D) combined task.

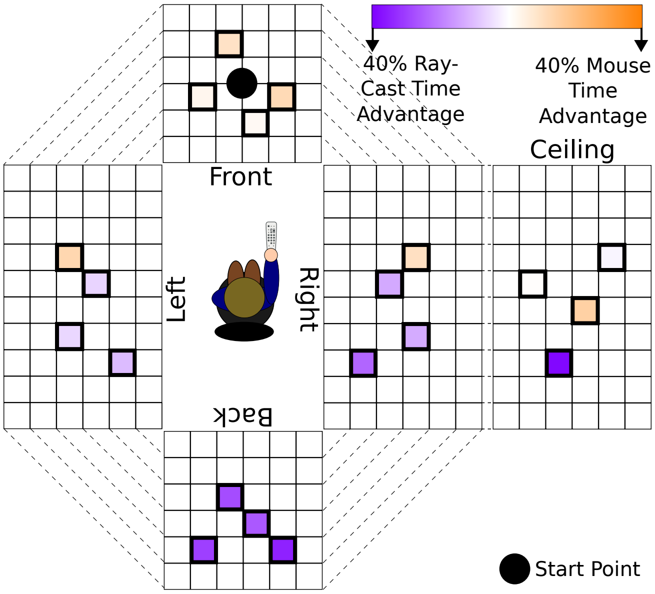


Figure 4. Interaction technique advantage for targeting per target. Targets that appear more orange had shorter average completion times with Mouse, those that appear more purple were shorter with Ray-cast. White targets are approximately equally fast to target with both techniques.

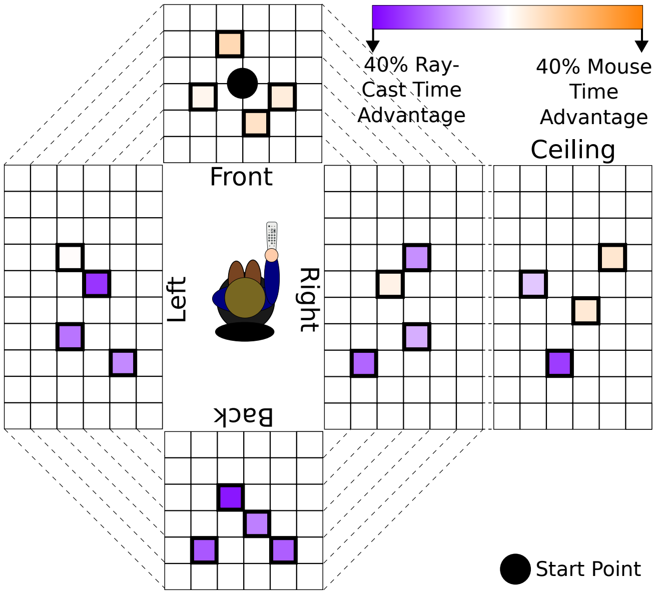


Figure 5. Interaction technique advantage for the combined task per target. See also caption of Figure 2.

Which Technique is Fastest for the Combined Task and Where?

A factorial repeated-measures ANOVA of completion time of the combined task (simultaneous searching and targeting) shows a main effect of target location ($F_{6.08,139.93} = 231.79$, $p < 0.001$, $\eta^2 = 0.837$), and technique ($F_{1,23} = 28.91$, $p < 0.001$, $\eta^2 = 0.044$), as well as a significant interaction between the two ($F_{19,437} = 5.53$, $p < 0.001$, $\eta^2 = 0.07$). In the combined task the Ray-cast technique had a general advantage (16.52% less time on average). Figure 5 shows that Ray-cast is generally faster for angles larger than 50° (locations not on the front wall). Figure 3.D shows that Ray-cast is faster for most locations except those in the front wall. The per wall post-hoc analysis shows statistically significant advantage of Mouse for front wall interaction ($F_{1,23} = 7.07$, $p < 0.05$, $\eta^2 = 0.03$) and for Ray-cast on all other surfaces (Table 2).

Wall	F(1,23)	p	η^2	Faster Technique
Front	7.07	<0.05	0.03	Mouse
Left	15.3	<0.001	0.07	Ray-Cast
Right	7	<0.05	0.03	Ray-Cast
Back	34.72	<0.001	0.22	Ray-Cast
Ceiling	5.24	<0.05	0.03	Ray-Cast

Table 2. Fastest technique per wall for the combined task.

A visual comparison of completion times of the combined task (Fig. 3.D) and the sum of the search+homing and targeting tasks (Fig. 3.C) reveals a key finding. Although overall the Mouse is faster if we consider the simple algebraic addition of search+homing and targeting times ($F_{1,23} =$

13.15, $p < 0.05$, $\eta^2 = 0.03$, $\mu_{mouse} = 6.78$, $\mu_{ray-cast} = 7.46$—Mouse 9.1% faster), the combined task times show the opposite ($F_{1,23} = 28.91$, $p < 0.001$, $\eta^2 = 0.044$, $\mu_{mouse} = 5.96$, $\mu_{ray-cast} = 5.12$—Ray-cast 16.52% faster). The best explanation for this inversion is that the Ray-cast technique supports better simultaneous execution of the searching and targeting subtasks.

What are the Main Spatial Asymmetries?

There are several spatial asymmetries of note in the data shown above, including left-right and ceiling effects.

Left-Right Asymmetries

A close examination of Figure 3.B shows that average Mouse targeting times for targets on the right surface are larger than their counterparts on the left surface. To corroborate this we ran a RM-ANOVA of targeting time with target coordinate, wall and technique as main factors with only the data of the left and right walls. The analysis shows a main effect of wall ($F_{1,23} = 12.65$, $p < 0.005$, $\eta^2 = 0.043$) with targeting times on the left wall 7.39% shorter on average than on the right wall. An even stronger left-right asymmetry effect exists for the search+homing subtask ($F_{1,23} = 51.23$, $p < 0.001$, $\eta^2_p = 0.289$, $\mu_{left} = 3.76$ s, $\mu_{right} = 5.88$ s —the left wall times are 56.38% shorter) and the combined task ($F_{1,23} = 17.11$, $p < 0.001$, $\eta^2 = 0.186$, $\mu_{left} = 4.75$ s, $\mu_{right} = 6.29$ s—left wall times 32.42% shorter). However, we noticed a posteriori that these tasks might have been affected by a confound of the experimental setting, where the brightness (and therefore the saliency) of elements projected on the right wall was noticeably lower. Although this obscures possible findings about the left-right asymmetry of the search+homing task and the combined task, it illustrates the effect that the change of the visual appearance could have on FCD tasks. Nevertheless the effect of laterality and brightness in searching and targeting subtasks will require further study. We believe that it is unlikely that the brightness confound would have had an effect on the targeting time measures because during the targeting subtask participants already knew the target location.

Ceiling Asymmetry

Although targets on the ceiling are at similar angles to their left wall counterparts, they generally were harder to search for ($\mu_{ceiling} = 5.96$ s, $\mu_{left} = 3.76$ s), target ($\mu_{ceiling} = 2.75$ s, $\mu_{left} = 2.23$ s) and search and target combined ($\mu_{ceiling} = 7.29$ s, $\mu_{left} = 4.75$ s). This is confirmed by three RM-ANOVAs, one per task, which compare only the data for ceiling and left wall targets, and all showed a main effect of the target's wall (Search+homing: $F_{1,23} = 81.22$, $p < 0.001$, $\eta^2 = 0.434$, Target: $F_{1,23} = 26.55$, $p < 0.001$, $\eta^2 = 0.17$, Combined: $F_{1,23} = 58.93$, $p < 0.001$, $\eta^2 = 0.448$).

Which Factors Predict Targeting Performance in FCDs?

FCDs are different from large displays in that they have very visible boundaries between projectable surfaces. FCDs are also different from most MDEs in that the displayable surface covers a much wider range of the user's visual field, including the ceiling, and projectable surfaces can be at more pronounced angles. Finally, immersive environments (e.g., VR CAVEs) also span a large portion of the visual field, but they are designed to minimize the visibility of boundaries and provide the illusion that the user is surrounded by a uniform environment. The following analysis addresses the question of which factors (and in which form) to include in models that describe pointing time in FCD environments. This question is also related to previous modeling efforts for large-display pointing that have considered using angles instead of linear dimensions [23, 24].

We use the Shannon-Welford formulation of Fitts's law ($MT = a + b_1 \cdot log_2(A+W) - b_2 \cdot log_2(W)$, where A represents the distance from the start point to the target and W represents the width of the target) [46] because it has been found to provide significantly better fits and, as shown by Shoemaker et al., it subsumes Kopper et al.'s exponentially adjusted formulation. Note that we average all participant's movement times for each possible target and technique, as it is customary in most targeting modeling papers.

We modelled the position of the target in two different ways: one in which distance and width are measured linearly along the surface of the room, and one where we use subtended angles of distance and width, calculated from the point of view of the participant for each target. The model accounts for participant height and shifts in the participant's position, although we asked and checked that the participants stayed roughly in the same position in the middle of the room throughout the experiment. To differentiate between angular and linear (surface) models, we use greek letters for angular models. We also consider whether including information in the model about the wall location would improve the fit of the models, which would also serve as confirmation of the effects of pointing time on wall location (Left-right Asymmetry and Ceiling Asymmetry sections above).

Figure 6 shows the detail of how we calculated A and W and their angular counterparts (α and ω) in our set up, with $A = D - (W/2)$ and $\alpha = \delta - (\omega/2)$. Surface distances across walls follow the shortest path on the surface, as if the walls were unfolded flat.

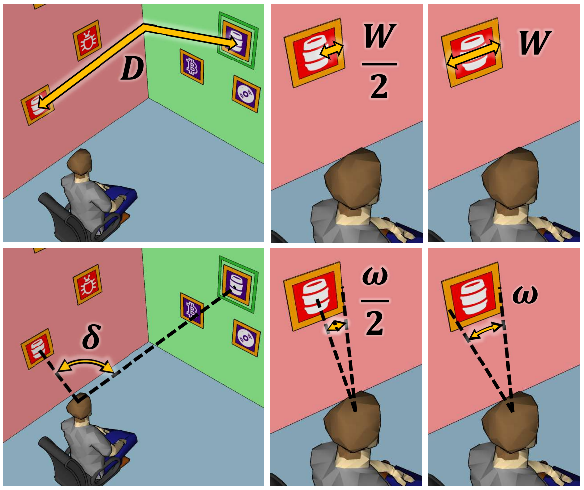


Figure 6. The surface (top row) and angular (bottom row) measures used to calculate the indexes of difficulty. The icon that the user faces is the original starting point, with the target icon on their left.

Table 3. The results of the regression modeling analysis. Slope 2 is N/A for some models because the linear width of the targets did not change.

Pointing Technique	Model Type	Walls Included	Intersect (*a*)	Slope 1 (b_1)	Slope 2 (b_2)	AIC	R^2
Mouse	Surface	No	1.6103	0.8233	N/A	33.67	78.235
Mouse	Surface	Yes	1.1417	1.4670	N/A	27.3	90.713
Mouse	Angle	No	-0.84662	0.79391	0.67180	25.53	86.893
Mouse	Angle	Yes	1.8221	0.6464	1.0981	11.92	95.551
Ray-Cast	Surface	No	1.56184	0.52536	N/A	6.85	84.838
Ray-Cast	Surface	Yes	2.1342	0.3639	N/A	-3.94	94.656
Ray-Cast	Angle	No	-1.79539	0.48435	-0.35255	-4.21	92.1083
Ray-Cast	Angle	Yes	-0.4121	0.3999	-0.1265	-5.67	95.07

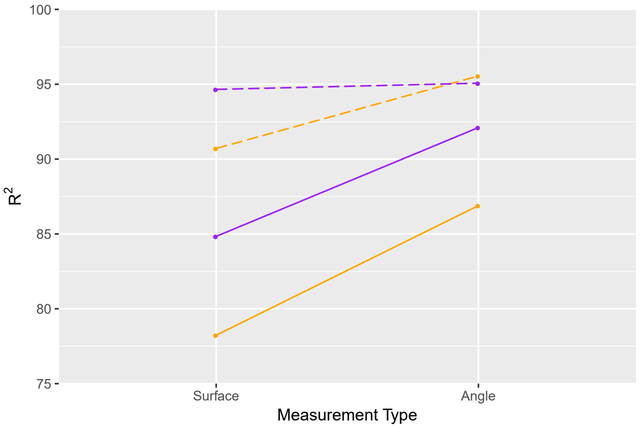


Figure 7. R^2 values for Mouse (orange) vs Ray-cast (purple). Dashed lines correspond to models that include the wall information.

The results are summarized in Figure 7 and in Table 3. The analysis shows several interesting trends. First, the models generally explain more of performance variability for Ray-cast than for Mouse. Second, our models using angular measures always fit better than their linear (surface) counterparts. This corroborates results in previous literature [23, 24]. Finally, information about which wall the target is on always adds information, which further supports the results from the targeting performance prediction section above.

The models that include wall information will naturally accommodate a larger portion of the variance just because they have additional parameters. To compensate for this we calculated Akaike Information Criterion values for each model [1], which are displayed in table 3. The best models for both datasets are always the angular models that include wall information (lowest AICs). Note that AIC absolute values are dependent on the specific data and generally not meaningful, it is the comparison of AICs within the same dataset that provides information about which model to select.

What are the Differences between Techniques in Subjective Workload Ratings?

The results of the 10-point Likert scale NASA TLX ratings are shown in Tables 4 and 5 for the separate and combined tasks respectively. The Wilcoxon Signed-rank tests found statistically significant differences in mental workload (Ray-cast better), physical load (Mouse better), performance (Mouse better), effort (Ray-cast better), and frustration (Ray-cast better) in the separate tasks. The same analysis for the combined task did not yield any significant differences, which suggests that participants were not able to consistently judge differences between techniques when they carried out searching and targeting simultaneously.

Table 4. NASA TLX rating frequencies, averages, medians and statistical tests for the separate task. Low values are better (e.g., lower mental load). M=Mouse, R=Ray-cast.

Separate	0	1	2	3	4	5	6	7	8	9	10	µ	Mdn	p	P<0.05	r
Mental M	0	4	6	3	6	2	1	0	2	0	0	3.38	3	0.04	✓	-0.3
Mental R	0	7	7	3	1	1	2	2	1	0	0	3.04	2			
Physical M	3	1	3	6	2	3	3	1	1	1	0	3.75	3	0.02	✓	-0.34
Physical R	1	2	6	4	3	0	5	2	0	1	0	3.79	3			
Temporal M	4	3	5	4	4	1	3	0	0	0	0	2.67	2.5	0.7	✗	-0.06
Temporal R	2	3	8	2	2	6	1	0	0	0	0	2.88	2			
Perform. M	1	9	5	3	3	1	2	0	0	0	0	2.38	2	0.045	✓	-0.29
Perform. R	2	7	7	1	5	1	0	0	0	1	0	2.42	2			
Effort M	0	2	4	4	3	4	2	4	1	0	0	4.25	4	0.03	✓	-0.32
Effort R	1	4	3	4	5	3	1	3	0	0	0	3.5	3.5			
Frust. M	2	4	7	0	5	3	1	0	2	0	0	3.13	2	0.01	✓	-0.38
Frust. R	4	6	7	2	1	3	1	0	0	0	0	2.13	2			

Table 5. NASA TLX rating frequencies, averages, medians and statistical tests for the combined task. Low values are better (e.g., lower mental load). M=Mouse, R=Ray-cast.

Combined	0	1	2	3	4	5	6	7	8	9	10	µ	Mdn	p	P<0.05	r
Mental M	0	5	7	3	3	3	1	1	0	1	0	3.21	2.5	0.4	✗	-0.12
Mental R	2	8	4	4	2	0	4	0	0	0	0	2.5	2			
Physical M	2	1	5	3	1	3	6	2	1	0	0	4.04	4.5	0.75	✗	-0.05
Physical R	2	5	4	3	3	4	3	0	0	0	0	3	3			
Temporal M	3	6	3	3	2	4	1	1	1	0	0	2.92	2.5	0.38	✗	-0.13
Temporal R	6	5	3	2	1	2	3	1	1	0	0	2.67	2			
Perform. M	4	6	5	0	3	1	3	2	0	0	0	2.71	2	0.77	✗	-0.04
Perform. R	4	9	5	3	2	0	0	0	0	0	1	1.92	1			
Effort M	0	4	4	3	3	3	3	2	2	0	0	4	4	0.11	✗	-0.23
Effort R	0	8	3	4	3	2	3	1	0	0	0	3.04	3			
Frust. M	4	5	1	4	4	2	2	0	1	1	0	3.08	3	0.05	✗	-0.28
Frust. R	5	7	4	4	2	1	1	0	0	0	0	1.92	1.5			

Experiment and Procedure Checks

We did not observe much clutching during the experiments, and an approximate post-hoc analysis of the data showed that there could have been a conservative upper bound of ten clutching gestures for the participant with the most clutches (out of 120 trials). We also ran an ANOVA with *technique order* and *target location* as main factors, which did not show any significant effect of order for Ray-casting ($F_{1,22} = 3.12$, $p > 0.05$, $\eta^2 = 0.05$) or Mouse ($F_{1,22} = 0.43$, $p > 0.05$, $\eta^2 = 0.006$).

DISCUSSION

The study provides five main findings:

- Overall, Ray-casting was significantly faster than the Mouse (13% faster times on average) for pointing tasks;
- There was substantial variation by target location: the Mouse was fastest for front wall tasks (6.5% and 7.3% shorter times on average for the targeting and combined tasks respectively) while Ray-cast was fastest for all other tested surfaces (12.4% and 16% on average across all surfaces for targeting and combined tasks respectively). The

Mouse advantage in the front wall is only significant for the combined task;

- Surprisingly, the Mouse was faster for tasks that required finding and targeting sequentially;
- Performance for targets on the ceiling was different from the other walls, with different specific locations being better or worse for the two techniques;
- Models with angular distances that incorporate the target's wall information fit the data best for both techniques.

As described in earlier sections, these results demonstrate specific ways in which the affordances and capabilities of the two interaction techniques (Ray-cast and Mouse) fit to the constraints of the full-coverage display. There are two overall principles that can summarize these earlier interpretations. First, as the user changes their relative orientation to the room (e.g., by turning around), Ray-cast begins to dominate because it shares the user's reference frame (and so is always in the right place to start a pointing action). Second, when the user must work with both the start location and a target location, the mouse's room-centric reference frame (rather than body-centric) becomes an advantage, as the mouse cursor can be 'parked' at the start point even as the user turns around.

In the following sections, we discuss how these basic differences provide the basis for new information about designing full-coverage environments and new information about targeting in surrounding spaces. We also discuss limitations to our study, and opportunities for further research.

Implications for Design

There are several lessons that designers of multi-display environments can take from our work. This study clearly indicates that there are differences between interaction techniques depending on the coverage of the display. This means that if the environment covers mostly the space in front of the user (e.g., systems that project around one display, such as Baudisch's Focus+Context system [5] or IllumiRoom [22]), then mouse-based interaction is likely to be the fastest and most precise technique. If the whole space around the user is to be used, however, then the large differences in pointing times and perceived effort between ray-casting and mousing (and the decided advantage for ray-casting to targets behind the user) could make a big difference in the usability of the system. In these full-coverage settings, it could be best to provide the user with both types of pointing: for example, the user would work with the mouse when manipulating objects on the front wall, but could pick up a ray-cast device when retrieving items from behind them. Since switching between devices could be a burden, it would be ideal if the environment did not need to include two separate devices. Ray-casting could be accomplished using the mouse itself (assuming it is wireless and can sense 6DoF movement) – mousing when placed on a surface, but ray-casting when held up in the air. If vision-based sensing is used, then it may be possible for the user to simply use their own finger to accomplish ray-casting (assuming an effective trigger mechanism can be implemented).

In addition, the degree of mobility in the environment can influence the choice of input device. If there is no real "front" to the space, and users move around to work with content throughout the room, a ray-cast device may be a better choice both because it does not require a horizontal surface, and because it performs better when the user must change their orientation frequently. Some multi-display environments provide specific pointing devices for use with particular surfaces or displays (e.g., a mouse linked to a high-resolution display), and this could be another way to achieve a hybrid approach.

In the case of combined searching / pointing tasks, the ability of the mouse to be 'parked' at the start location can also be valuable - but these kinds of tasks are likely to make up only a small percentage of the overall manipulations that are carried out, and there are also other ways to complete these kinds of manipulations (e.g., acquiring the start object and 'taking it with you' as the user searches for the destination object).

Finally, the ceiling is a potentially useful place to put elements of a digital interface, especially because in most everyday environments the ceiling is one of the only surfaces that has no real-world objects on it, and therefore offers a large potential work space. There are limitations to this opportunity, however — our study results suggest that the ceiling should be avoided for elements that require frequent interaction, since we found that pointing here is generally slower and more awkward. In addition, users may need to get used to looking for objects on the ceiling; even though our study participants knew that some objects could appear in the ceiling, they tended to look there last, leading to higher search+homing times. The most useful region of the ceiling appears to be the area right above the main focus area (assuming that the room has one); this area was almost as good as the prime real estate just in front of the users for pointing (although not for search+homing).

Implications for Targeting

There are also several findings from the study that add to our understanding of targeting in large everyday spaces. The study corroborated the importance of angle in modeling performance, but also raised several other issues that have not been considered in detail before. First, the difference in performance for different target directions should be considered further — we found differences both between left and right, and between horizontal and upwards movement. Although previous work has also shown directional differences (e.g., [8, 51]), most studies have looked only at small arm and hand movements, rather than targets that also require reorientation of the body. We recognize the potential confound of the difference in brightness between the left and right side of our experimental setting, so these results must be followed up in further studies. The novelty of pointing to the ceiling is also an interesting topic for further study — it may be that once people become used to storing digital objects on the ceiling, pointing performance with this surface becomes more similar to the other walls in the room.

Second, our results suggest that pointing performance can also be considered in light of the previous and subsequent actions that the user carries out. In our study, the performance of the different techniques was affected by the search+homing step in the task, because the mouse's reference frame allowed search to be decoupled from pointing (i.e., by parking the mouse at the start point). Previous work has considered chained interaction tasks, but the larger scale of full-coverage displays and the need to make large changes in bodily orientation presents a new context for these explorations.

Third, we did not explicitly investigate the development of spatial memory for targets to the side or behind the user, and our models of pointing will likely need additional changes to accommodate targeting actions to well-known targets. Once a user develops strong spatial memory for target locations, it will be of interest to study how their initial ballistic impulses may involve pointing to locations outside the field of view, and movements that involve changes in body orientation. Although previous work has shown that arbitrary targets around the user can be remembered and pointed to quickly [42], there is little work to model the details of these pointing actions.

Fourth, our modeling exploration suggests that models based on angular measures of target position are superior for pointing around the user than measures based on surface distance. This had been shown before for large displays [24, 23] but not for pointing in FCDs. Since we found asymmetries in performance between different walls it also makes sense that using information about target wall locations in new models will provide additional predictive power.

Some Ergonomic Considerations

There are several ergonomic issues that need to be considered when designing interaction in FCDs. Although the main focus of our study was not ergonomics, our experience running this experiment provided some useful information. The Mouse was considered less physically demanding in our experimental setup (with a lap tray) for the separate task. Previous research has suggested that laser pointers can lead to fatigue [36, 33]; holding a pointer requires more constant effort than holding a mouse that rests on a horizontal surface. Although holding a lap tray might not be practical in many scenarios, we think this is an unexplored option that could work well in some situations (e.g., a control room where people sit in rolling chairs). A related consideration is that, as we found with one of our participants, we might not be able to assume that everyone knows how to operate a mouse in a future of touch input dominance. Finally, we do not know whether neck strain will limit usage of the ceiling as a display.

Limitations and Directions for Future Study

There are numerous opportunities for further investigations of pointing and targeting in full-coverage environments, some of which arise from limitations in our current study. First, our experimental setting did not project onto the floor, and it will be interesting to see if objects below the user lead to similar performance as objects on the ceiling. Second, our setting had an unavoidable brightness difference between the left and right walls, and it will be important to identify whether our directional results have any interaction with brightness — particularly since brightness is an overall concern for large-scale projected environments. Third, the organization and distribution of targets in our tasks was relatively uniform, and may not match the way that users arrange items in real-world scenarios. In addition, our setting had blank walls, and so did not examine how the presence of physical objects in the room could change targeting — for example, it may be that it is harder to find or point to projected objects that are among physical objects such as bookshelves or cabinets. Fourth, our setup with the mouse involved a rotating chair and a lap board that allowed free movement of the mouse to all regions of the room without clutching; in real-world environments, mouse input may be much more constrained (e.g., to existing horizontal surfaces) which would likely further reduce the performance of the mouse to targets at large angles. Fifth, our experiment only included included right-handed participants. It would be useful to replicate the study with righ-handed and left-handed participants to pinpoint whether handedness is the main source of the asymmetry.

Finally, we designed the experimental procedure so that participants always did the separate tasks before the combined tasks (instead of balancing the order). We believe it is unlikely that this could altered the results, but we cannot completely rule it out. Researchers who might want to replicate this study or consider similar designs should, however, take into account that it may be harder to get participants to do a separate task if they are already used to the combined task. This could introduce measurement artifacts.

CONCLUSION

With the increasing viability and appeal of wide angle and full-coverage display environments it has become more important to evaluate the available interface-design choices, to ensure that these new types of systems are usable and practical. One of the key user actions to support in these new environments is digital object selection and targeting.

We carried out a study that explored the effects of two relevant pointing techniques in targeting tasks *around* the room and the ceiling. We found that a mouse-based technique provides the fastest targeting interaction when the targets do not require the participant to move their body, but a Ray-casting technique was superior for targets at larger angles. Additionally, we discovered that the Mouse technique has the advantage of enabling the cursor to be "parked" while the user looks elsewhere, and that the Ray-casting technique enables better overlap of searching and targeting tasks when the user needs to find an object of interest in the room.

Our findings can help inform designers as they choose interaction techniques that best suit the intended environment, and subsequently support the success of interfaces that take advantage of the full physical environment for digital and augmented information in our future work and home spaces.

ACKNOWLEDGEMENTS

We would like to thank our participants, the reviewers and the AC for their valuable comments, as well as the members of the SACHI research group and the systems group. This research was possible thanks to the generous support of SurfNet (NSERC, Canada), EPSRC (Small Equipment Grant) as well as Pufferfish Displays (`http://www.pufferfishdisplays.com`).

REFERENCES

1. Hirotogu Akaike. 1992. Information theory and an extension of the maximum likelihood principle. In *Breakthroughs in statistics*. Springer, 610–624.
2. Ferran Argelaguet and Carlos Andujar. 2009. Efficient 3D Pointing Selection in Cluttered Virtual Environments. *IEEE Comput. Graph. Appl.* 29, 6 (Nov. 2009), 34–43. DOI: `http://dx.doi.org/10.1109/MCG.2009.117`
3. Ferran Argelaguet and Carlos Andujar. 2013. Special Section on Touching the 3rd Dimension: A Survey of 3D Object Selection Techniques for Virtual Environments. *Comput. Graph.* 37, 3 (May 2013), 121–136. DOI: `http://dx.doi.org/10.1016/j.cag.2012.12.003`
4. Amartya Banerjee, Jesse Burstyn, Audrey Girouard, and Roel Vertegaal. 2012. MultiPoint: Comparing laser and manual pointing as remote input in large display interactions. *International Journal of Human-Computer Studies* 70, 10 (2012), 690 – 702. DOI: `http://dx.doi.org/https://doi.org/10.1016/j.ijhcs.2012.05.009` Special issue on Developing, Evaluating and Deploying Multi-touch Systems.
5. Patrick Baudisch, Nathaniel Good, and Paul Stewart. 2001. Focus Plus Context Screens: Combining Display Technology with Visualization Techniques. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology (UIST '01)*. ACM, New York, NY, USA, 31–40. DOI: `http://dx.doi.org/10.1145/502348.502354`
6. Jacob T. Biehl and Brian P. Bailey. 2004. ARIS: An Interface for Application Relocation in an Interactive Space. In *Proceedings of Graphics Interface 2004 (GI '04)*. Canadian Human-Computer Communications Society, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 107–116. `http://dl.acm.org/citation.cfm?id=1006058.1006072`
7. Richard A. Bolt. 1980. "Put-that-there": Voice and Gesture at the Graphics Interface. In *Proceedings of the 7th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '80)*. ACM, New York, NY, USA, 262–270. DOI: `http://dx.doi.org/10.1145/800250.807503`
8. James Boritz, Kellogg S. Booth, and William B. Cowan. 1991. Fitt's Law Studies of Directional Mouse Movement. In *Proceedings of Graphics Interface '91 (GI '91)*. Canadian Man-Computer Communications Society, Toronto, Ontario, Canada, 216–223. `http://graphicsinterface.org/wp-content/uploads/gi1991-28.pdf`
9. Doug Bowman, Chadwick. Wingrave, Joshua. Campbell, and Vinh Ly. 2001. *Using Pinch Gloves(TM) for both Natural and Abstract Interaction Techniques in Virtual Environments*. Technical Report. Computer Science, Virginia Tech.
10. Michelle A. Brown, Wolfgang Stuerzlinger, and E. J. Mendonça Filho. 2014. The Performance of Un-instrumented In-air Pointing. In *Proceedings of Graphics Interface 2014 (GI '14)*. Canadian Information Processing Society, Toronto, Ont., Canada, Canada, 59–66. `http://dl.acm.org/citation.cfm?id=2619648.2619659`
11. Stuart K. Card, William. K. English, and Betty. J. Burr. 1987. Human-computer Interaction. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, Chapter Evaluation of Mouse, Rate-controlled Isometric Joystick, Step Keys, and Text Keys, for Text Selection on a CRT, 386–392. `http://dl.acm.org/citation.cfm?id=58076.58107`
12. Géry Casiez, Daniel Vogel, Ravin Balakrishnan, and Andy Cockburn. 2008. The Impact of Control-Display Gain on User Performance in Pointing Tasks. *Human-Computer Interaction* 23, 3 (2008), 215–250. DOI: `http://dx.doi.org/10.1080/07370020802278163`
13. Yeonjoo Cha and Rohae Myung. 2010. Extended Fitts' law in Three-Dimensional Pointing Tasks. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 54, 13 (2010), 972–976. DOI: `http://dx.doi.org/10.1177/154193121005401311`
14. Brian W. Epps. 1986. Comparison of Six Cursor Control Devices Based on Fitts' Law Models. *Proceedings of the Human Factors Society Annual Meeting* 30, 4 (1986), 327–331. DOI: `http://dx.doi.org/10.1177/154193128603000403`
15. Paul M Fitts. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology* 47, 6 (1954), 381.
16. Clifton Forlines, Daniel Vogel, and Ravin Balakrishnan. 2006. HybridPointing: Fluid Switching Between Absolute and Relative Pointing with a Direct Input Device. In *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology (UIST '06)*. ACM, New York, NY, USA, 211–220. DOI: `http://dx.doi.org/10.1145/1166253.1166286`
17. Clifton Forlines, Daniel Wigdor, Chia Shen, and Ravin Balakrishnan. 2007. Direct-touch vs. Mouse Input for Tabletop Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)*. ACM, New York, NY, USA, 647–656. DOI: `http://dx.doi.org/10.1145/1240624.1240726`
18. Fabian Göbel, Konstantin Klamka, Andreas Siegel, Stefan Vogt, Sophie Stellmach, and Raimund Dachselt. 2013. Gaze-supported Foot Interaction in Zoomable Information Spaces. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13)*. ACM, New York, NY, USA, 3059–3062. DOI: `http://dx.doi.org/10.1145/2468356.2479610`

19. Tovi Grossman and Ravin Balakrishnan. 2004. Pointing at Trivariate Targets in 3D Environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '04)*. ACM, New York, NY, USA, 447–454. `DOI:` `http://dx.doi.org/10.1145/985692.985749`

20. Carl Gutwin, Andy Cockburn, and Ashley Coveney. 2017. Peripheral Popout: The Influence of Visual Angle and Stimulus Intensity on Popout Effects. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 208–219. `DOI:` `http://dx.doi.org/10.1145/3025453.3025984`

21. Juan Pablo Hourcade and Natasha Bullock-Rest. 2012. How Small Can You Go?: Analyzing the Effect of Visual Angle in Pointing Tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 213–216. `DOI:` `http://dx.doi.org/10.1145/2207676.2207706`

22. Brett R. Jones, Hrvoje Benko, Eyal Ofek, and Andrew D. Wilson. 2013. IllumiRoom: Peripheral Projected Illusions for Interactive Experiences. In *ACM SIGGRAPH 2013 Emerging Technologies (SIGGRAPH '13)*. ACM, New York, NY, USA, Article 7, 1 pages. `DOI:` `http://dx.doi.org/10.1145/2503368.2503375`

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

24. Regis Kopper, Doug A Bowman, Mara G Silva, and Ryan P McMahan. 2010. A human motor behavior model for distal pointing tasks. *International journal of human-computer studies* 68, 10 (2010), 603–615.

25. I. Scott MacKenzie. 1992. Fitts' Law As a Research and Design Tool in Human-computer Interaction. *Hum.-Comput. Interact.* 7, 1 (March 1992), 91–139. `DOI:` `http://dx.doi.org/10.1207/s15327051hci0701_3`

26. I. Scott MacKenzie and William Buxton. 1992. Extending Fitts' Law to Two-dimensional Tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '92)*. ACM, New York, NY, USA, 219–226. `DOI:http://dx.doi.org/10.1145/142750.142794`

27. I. Scott MacKenzie, Abigail Sellen, and William A. S. Buxton. 1991. A Comparison of Input Devices in Element Pointing and Dragging Tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '91)*. ACM, New York, NY, USA, 161–166. `DOI:http://dx.doi.org/10.1145/108844.108868`

28. David E. Meyer, Richard A. Abrams, Sylvan Kornblum, Charles E. Wright, and J. E. Keith Smith. 1988. Optimality in human motor performance: Ideal control of rapid aimed movements. *Psychological Review* 95, 3 (1988), 340–370. `DOI:` `http://dx.doi.org/10.1037/0033-295X.95.3.340`

29. Brad A. Myers, Rishi Bhatnagar, Jeffrey Nichols, Choon Hong Peck, Dave Kong, Robert Miller, and A. Chris Long. 2002. Interacting at a Distance: Measuring the Performance of Laser Pointers and Other Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '02)*. ACM, New York, NY, USA, 33–40. `DOI:` `http://dx.doi.org/10.1145/503376.503383`

30. Miguel A. Nacenta. 2010. *Cross-display object movement in multi-display environments*. Ph.D. Dissertation. University of Saskatchewan. `https://ecommons.usask.ca/handle/10388/etd-01062010-123426`

31. Miguel A. Nacenta, Dzmitry Aliakseyeu, Sriram Subramanian, and Carl Gutwin. 2005. A Comparison of Techniques for Multi-display Reaching. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '05)*. ACM, New York, NY, USA, 371–380. `DOI:http://dx.doi.org/10.1145/1054972.1055024`

32. Miguel A. Nacenta, Satoshi Sakurai, Tokuo Yamaguchi, Yohei Miki, Yuichi Itoh, Yoshifumi Kitamura, Sriram Subramanian, and Carl Gutwin. 2007. E-conic: A Perspective-aware Interface for Multi-display Environments. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology (UIST '07)*. ACM, New York, NY, USA, 279–288. `DOI:` `http://dx.doi.org/10.1145/1294211.1294260`

33. Miguel A. Nacenta, Samer Sallam, Bernard Champoux, Sriram Subramanian, and Carl Gutwin. 2006. Perspective Cursor: Perspective-based Interaction for Multi-display Environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. ACM, New York, NY, USA, 289–298. `DOI:` `http://dx.doi.org/10.1145/1124772.1124817`

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

35. Ji-Young Oh and Wolfgang Stuerzlinger. 2002a. Laser Pointers as Collaborative Pointing Devices. In *Proceedings of the Graphics Interface 2002 Conference (GI '02)*. 141–150. `http://graphicsinterface.org/wp-content/uploads/gi2002-17.pdf`

36. Ji-Young Oh and Wolfgang Stuerzlinger. 2002b. Laser pointers as collaborative pointing devices. In *Proceedings of Graphics Interface*, Vol. 2002. 141–149.

37. Dan R. Olsen, Jr. and Travis Nielsen. 2001. Laser Pointer Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '01)*. ACM, New York, NY, USA, 17–22. `DOI:` `http://dx.doi.org/10.1145/365024.365030`

38. J. Karen Parker, Regan L. Mandryk, and Kori M. Inkpen. 2005. TractorBeam: Seamless Integration of Local and Remote Pointing for Tabletop Displays. In *Proceedings of Graphics Interface 2005 (GI '05)*. Canadian Human-Computer Communications Society, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 33–40. `http://dl.acm.org/citation.cfm?id=1089508.1089515`

39. Julian Petford, Miguel A. Nacenta, Carl Gutwin, Joseph Eremondi, and Cody Ede. 2016. The ASPECTA Toolkit: Affordable Full Coverage Displays. In *Proceedings of the 5th ACM International Symposium on Pervasive Displays (PerDis '16)*. ACM, New York, NY, USA, 87–105. DOI: `http://dx.doi.org/10.1145/2914920.2915006`

40. Ivan Poupyrev, Suzanne Weghorst, Mark Billinghurst, and Tadao Ichikawa. 1997. A Framework and Testbed for Studying Manipulation Techniques for Immersive VR. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology (VRST '97)*. ACM, New York, NY, USA, 21–28. DOI: `http://dx.doi.org/10.1145/261135.261141`

41. Ramesh Raskar, Greg Welch, Matt Cutts, Adam Lake, Lev Stesin, and Henry Fuchs. 1998. The Office of the Future: A Unified Approach to Image-based Modeling and Spatially Immersive Displays. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '98)*. ACM, New York, NY, USA, 179–188. DOI: `http://dx.doi.org/10.1145/280814.280861`

42. Adrian Reetz and Carl Gutwin. 2014. Making Big Gestures: Effects of Gesture Size on Observability and Identification for Co-located Group Awareness. In *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 4087–4096. DOI: `http://dx.doi.org/10.1145/2556288.2557219`

43. Michael Rohs and Antti Oulasvirta. 2008. Target Acquisition with Camera Phones when Used As Magic Lenses. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*. ACM, New York, NY, USA, 1409–1418. DOI: `http://dx.doi.org/10.1145/1357054.1357275`

44. Michael Rohs, Antti Oulasvirta, and Tiia Suomalainen. 2011. Interaction with Magic Lenses: Real-world Validation of a Fitts' Law Model. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 2725–2728. DOI: `http://dx.doi.org/10.1145/1978942.1979343`

45. Alexander Schick, Florian van de Camp, Joris Ijsselmuiden, and Rainer Stiefelhagen. 2009. Extending Touch: Towards Interaction with Large-scale Surfaces. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '09)*. ACM, New York, NY, USA, 117–124. DOI: `http://dx.doi.org/10.1145/1731903.1731927`

46. Garth Shoemaker, Takayuki Tsukitani, Yoshifumi Kitamura, and Kellogg S Booth. 2012. Two-part models capture the impact of gain on pointing performance. *ACM Transactions on Computer-Human Interaction (TOCHI)* 19, 4 (2012), 28.

47. Sophie Stellmach, Sebastian Stober, Andreas Nürnberger, and Raimund Dachselt. 2011. Designing Gaze-supported Multimodal Interactions for the Exploration of Large Image Collections. In *Proceedings of the 1st Conference on Novel Gaze-Controlled Applications (NGCA '11)*. ACM, New York, NY, USA, Article 1, 8 pages. DOI: `http://dx.doi.org/10.1145/1983302.1983303`

48. Helmut Strasser and Karl-Werner MÃijller. 1999. Favorable movements of the hand-arm system in the horizontal plane assessed by electromyographic investigations and subjective rating. *International Journal of Industrial Ergonomics* 23, 4 (1999), 339 – 347. DOI: `http://dx.doi.org/https://doi.org/10.1016/S0169-8141(98)00050-X`

49. Robert J. Teather and Wolfgang Stuerzlinger. 2011. Pointing at 3D targets in a stereo head-tracked virtual environment. In *2011 IEEE Symposium on 3D User Interfaces (3DUI)*. 87–94. DOI: `http://dx.doi.org/10.1109/3DUI.2011.5759222`

50. Kentaro Toyama. 1998. Look, ma-no hands! hands-free cursor control with real-time 3d face tracking. In *Workshop on Perceptual User Interfaces (PUI '98)*. 49–54.

51. Jean-Luc Velay, Virginie Daffaure, Nathalie Raphael, and Simone Benoit-Dubrocard. 2001. Hemispheric asymmetry and interhemispheric transfer in pointing depend on the spatial components of the movement. *Cortex* 37, 1 (2001), 75–90.

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

53. Manuela Waldner and Dieter Schmalstieg. 2010. Experiences with Mouse Control in Multi-display Environments. In *Proceedings of the International Conference on Advanced Visual Interfaces (AVI '10)*. ACM, New York, NY, USA, 411–411. DOI: `http://dx.doi.org/10.1145/1842993.1843089`

54. Andrew D. Wilson and Hrvoje Benko. 2016. Projected Augmented Reality with the RoomAlive Toolkit. In *Proceedings of the 2016 ACM on Interactive Surfaces and Spaces (ISS '16)*. ACM, New York, NY, USA, 517–520. DOI:`http://dx.doi.org/10.1145/2992154.2996362`

55. Robert Xiao, Miguel A. Nacenta, Regan L. Mandryk, Andy Cockburn, and Carl Gutwin. 2011. Ubiquitous Cursor: A Comparison of Direct and Indirect Pointing Feedback in Multi-display Environments. In *Proceedings of Graphics Interface 2011 (GI '11)*. Canadian

Human-Computer Communications Society, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 135–142. http://dl.acm.org/citation.cfm?id=1992917.1992939

56. Hui-Shyong Yeo. 2016. Single-handed Interaction for Mobile and Wearable Computing. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct (MobileHCI '16)*. ACM, New York, NY, USA, 907–908. DOI:http://dx.doi.org/10.1145/2957265.2963110

57. Hui-Shyong Yeo, Xiao-Shen Phang, Steven J. Castellucci, Per Ola Kristensson, and Aaron Quigley. 2017. Investigating Tilt-based Gesture Keyboard Entry for Single-Handed Text Entry on Large Devices. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 4194–4202. DOI: http://dx.doi.org/10.1145/3025453.3025520