Pointable: An In-Air Pointing Technique to Manipulate Out-of-Reach Targets on Tabletops

Amartya Banerjee[1]*, Jesse Burstyn*[1]*, Audrey Girouard*[1,2]*, Roel Vertegaal*[1]

[1] Human Media Lab
School of Computing, Queen's University
Kingston, Ontario, K7L 3N6 Canada

[2] School of Information Technology
Carleton University
Ottawa, Ontario, K1S 5B6 Canada

{banerjee, jesse, roel}@cs.queensu.ca, audrey_girouard@carleton.ca

ABSTRACT

Selecting and moving digital content on interactive tabletops often involves accessing the workspace beyond arm's reach. We present Pointable, an in-air, bimanual perspective-based interaction technique that augments touch input on a tabletop for distant content. With Pointable, the dominant hand selects remote targets, while the non-dominant hand can scale and rotate targets with a dynamic C/D gain. We conducted 3 experiments; the first showed that pointing at a distance using Pointable has a Fitts' law throughput comparable to that of a mouse. In the second experiment, we found that Pointable had the same performance as multi-touch input in a resize, rotate and drag task. In a third study, we observed that when given the choice, over 75% of participants preferred to use Pointable over multi-touch for target manipulation. In general, Pointable allows users to manipulate out-of-reach targets, without loss of performance, while minimizing the need to lean, stand up, or involve collocated collaborators.

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces: Input Devices and Strategies, Interaction Styles.

General terms: Design, Experimentation, Performance

Keywords: multi-touch, remote interaction, tabletop, input device, interaction technique

INTRODUCTION

Selecting and moving digital content on interactive tabletops often involves gaining access to workspace beyond arm's reach. When a tabletop only supports direct-touch as an input modality, users must compromise and use one of two strategies to acquire out-of-reach documents:

- *Move, stand up, or lean over the table to reach the document*. In a single-user setting, this is an inconvenience. For a multi-user collaborative setting, each of these movements can obstruct the view of other users, or disturb their physical territory [33].
- *Ask another user to pass the document* [30]. This typically disrupts the workflow of the called upon user, even more so when this document is also out of their reach.

Toney and Thomas [36] reported that, for a single user, over 90% of direct-touch interactions were confined to 28% of the total length of the table. Thus, several techniques have been proposed to improve the efficiency of reaching distant digital content on large displays. These include remote pointing [26] and indirect pointing techniques for distant targets [2,3,30]. While these techniques provide access to out-of-reach areas, they involve frequent change of input modalities, i.e. the transition between using direct-touch and picking up a device (mouse, pen or laser pointers).

With this in mind, we present the design and evaluation of *Pointable*, an interaction technique that combines precise reachability with in-place manipulation of remote digital content. This technique has been created to satisfy the following design goals:

1. *Augment Touch*: Pointable should serve as an addition to direct-touch, not replace or impede it.
2. *Minimize Modality Switches*: Pointable should have a low invocation and dismissal overhead.
3. *In-Place Manipulation*: Pointable should allow users to perform in-place manipulation for remote targets.
4. *Low Fatigue*: Pointable should minimize physical movement and fatigue where possible.
5. *Unobtrusive*: In multi-user settings, Pointable should minimize intrusion into the personal space of others.

Pointable is an in-air, asymmetric bimanual manipulation technique, which augments touch input on a tabletop to more easily interact with distant content. The dominant hand points and acquires remote targets (Figure 1), while the non-dominant hand scales and rotates the target without the need to drag the target closer, i.e. Pointable allows users to perform *in-place manipulation*. However, if users prefer direct-touch for scaling and rotation transforms, they can use Pointable just as a tool to move content to and from a distant area of the tabletop. Switching from using Pointable to using direct-touch is simply a matter of placing a fingertip of the dominant hand on the tabletop.

ITS 2011, November 13-16, Kobe, Japan.

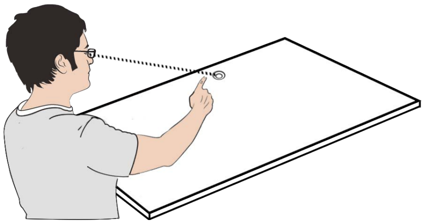


Figure 1. Perspective-based pointing technique. The cursor position is determined through two points: the nose bridge, and the index finger of the dominant hand.

The pointing technique for the dominant hand employs image-plane or perspective-based [13,27] pointing (Figure 1) that follows the user's line of sight. As seen from the user's perspective, finger positions are mapped onto the display when they are within its boundary box. Importantly, the non-dominant hand does not have to point at the remote target, or the surface itself, to invoke manipulations. After the dominant hand has acquired the target, the user can then perform a selection gesture with their non-dominant hand to enable scaling and rotation. Varying the distance between both hands results in an affine transformation that controls the target's size and orientation.

In this paper, we report on three experiments designed to investigate Pointable's potential when used in isolation or in conjunction with multi-touch on a tabletop. The first experiment measures performance of Pointable in a Fitts' law analysis. The second compares manipulation performance of Pointable versus multi-touch. Finally, the third experiment observes user behavior when Pointable is used in tandem with touch.

RELATED WORK

Pointable builds upon the following areas of previous research: (1) sensing direct-touch and in-air gestures for tabletops; (2) accessing out-of-reach areas on a large display; (3) bimanual input and the use of the non-dominant hand to switch between input modalities.

Sensing Direct-Touch and In-Air Gestures for Tabletops

DiamondTouch [4] and *SmartSkin* [31] are early sensing technologies measuring direct-touch on tabletops. *DiamondTouch* presented a technique allowing multiple, simultaneous users to interact with a tabletop. Its primary feature is the ability to associate each touch on a common workspace with a specific user. Using capacitive sensing, *SmartSkin* recognizes multiple hand positions and shapes, and calculates the distance between a hand and the surface within 5-10cm.

DViT by SMART Technologies [35] uses computer vision to sense touch. This technology detects a hovering finger more precisely than either DiamondTouch or SmartSkin. *Barehands* [32] and *Touchlight* [41] also use computer vision to track uninstrumented hands pressing against a vertical surface. *Barehands* transforms ordinary displays into touch-sensitive surfaces with infrared (IR) cameras, while *Touchlight* detects hand gestures over a semi-transparent upright surface with cameras. All these techniques can be implemented on tabletops, with a key ability to extract hover information. More recently, the Kinect depth camera [16] was used in *LightSpace* [40] as a sensor to detect both in-air gestural input and touch on a surface.

The initial version of the Microsoft *Surface* [19] used a bottom-projected display that could sense objects placed on top using integrated cameras and computer vision. The Surface 2 uses a new display technology where each pixel is a combination of RGB and IR elements, thus being able to detect hand shadows close to the surface.

To augment touch with Pointable, we drew on this body of prior research to explore the affordances associated with rich sensor data, including but not limited to, touch input, arm or hand hover information, and in-air gestural data.

Accessing Out-of-Reach Areas on a Large Display

We categorize techniques for accessing and positioning out-of-reach digital content into widgets, cursors, and pen-based interactions, and remote interactions.

Widgets, Cursors and Pen-based Interactions. Widget or cursor based interaction techniques [2,3,15] can be used to access distant digital content on tabletops, while shuffling or flicking [30,42] facilitate moving objects on large displays. *I-Grabber* [1] is a multi-touch based visualization that acts as a virtual hand extension for reaching distant items on an interactive tabletop.

Remote Interaction Techniques - Device-based. The following device-based techniques could potentially be applied to tabletop interactions.

A laser pointer is a common device for remote interactions with large displays [20]. Nacenta et al. [22] evaluated an array of methods for interacting with remote content on tabletops in collaborative settings. These techniques included direct-touch with passing, radar-based views, and laser pointers, among others. Users found it difficult to acquire smaller and more distant targets with laser pointers. They observed that when using laser pointers, collaboration was reduced, as the lack of embodiment in the technique did not communicate where a user was pointing.

TractorBeam [30] allows users to select objects directly, using a stylus as touch input, and remotely, with the stylus serving as a laser pointer. Parker et al. found it to be a fast technique for accessing remote content on a tabletop, though users faced issues with smaller, distant targets [22]. Building on the initial system, Parker et al. compared three selection aids to improve target acquisition with ray-casting: expanding the cursor, expanding the target, and snapping to the target; the last was found to be the fastest technique [25]. With support for only a single contact point, *TractorBeam* focused on target selection and not manipulation.

Remote Interaction Techniques - Device-less. Vogel and Balakrishnan [37] explored single hand pointing and clicking interactions with large displays from a distance. They proposed *AirTap* and *ThumbTrigger* as clicking techniques, and found that ray-casting was a fast, yet inaccurate pointing method. Jota et al. [13] compared four pointing techniques: laser, arrow, image-plane and fixed-origin. They demonstrated that taking the user's line of sight (i.e. perspective) into account improves performance for tasks requiring more accuracy. Their work was restricted to single, unimanual interactions. Similarly, *Shadow Reaching* [34] applied a perspective projection to a shadow representation of the user to enable manipulation of distant objects on a large display.

The *g-speak* [24] spatial operating environment offers users remote bimanual input. The user points at a target by making a trigger gesture, previously demonstrated by Grossman et al. [7].

Most device-based remote interactions, including many of the widget or cursor-based techniques, involve picking up an intermediary object to interact with the tabletop. Thus users are prevented from transitioning to direct touch-based input seamlessly. In addition, most of these techniques cannot be used for in-place manipulation of distant objects. These key issues must be addressed, and motivated our design goals of *minimizing modality switches* and providing *in-place manipulation*, with Pointable.

Bimanual Input & Non-Dominant Hand as a Modifier

Myers and Buxton [21] found that, given appropriate context, users were capable of providing continuous data from two hands simultaneously without significant overhead. The speed of performing a task was directly proportional to the degree of parallelism employed. In another example, Latulipe et al. [17] compared the performance of single mouse input to symmetric and asymmetric dual mouse input in an image alignment task that involved minor amounts of translation, scaling and rotation. They found that the symmetrical technique recorded the highest performance followed by asymmetrical.

Contextualizing the actions of the dominant hand is commonly achieved by using the non-dominant hand as a modifier. Nancel et al. [23] used bimanual interaction techniques to pan-and-zoom content on a large display. Since pan-zoom operations inherently have a high level of parallelism, it is well afforded by the use of bimanual input techniques [8]. In Rock-and-Rails [39], the shape of the non-dominant hand was used to switch between different modes, such as isolating resize or rotate transforms. Hinckley et al. [11] changed the input mode of a pen held in the dominant hand via multi-touch gestures performed by the non-dominant hand.

The use of bimanual interactions, including those where the non-dominant hand can be used to switch contexts, to increase the level of parallelism was also central to the development of Pointable.

DESIGN RATIONALE & POINTABLE DESCRIPTION

When reaching for distant content on an interactive tabletop, it is desirable if a user does so without operating multiple input devices. However, including essential input actions, such as selection, rotation and translation, can quickly overload the mappings of just one input device and reduce its usability. To alleviate this design tension, Pointable supports multi-modal gesturing using bimanual asymmetric input. In line with Guiard's Kinematic Chain [8], the dominant hand points, while the non-dominant hand scales and rotates. Compared to using devices such as laser pointers or mice, in-air pointing offers a minimal number of modality switches. Hence, the transition between touch and pointing can be fluid, where touch contacts have priority over in-air gestures.

Even though this type of freehand pointing has been proposed as an input solution for large wall displays, it can be imprecise for pointing tasks [16,23,29,40] and causes arm-fatigue, particularly for up-down arm movements [30]. However, for tabletop displays, analogous movements have more favorable ergonomic properties; users can steady their arm and reduce fatigue by resting it on the tabletop.

In-air pointing also helps to lessen an input device's impact on proxemics by minimizing intrusion into the personal space of other users. Disruptions are also less taxing to the called upon user; instead of physically passing a document, the requesting user can move a remote document, after negotiating approval for its transfer.

We designed Pointable with the following core characteristics as a first step to understand how to support this proxemic fluidity while gesturing at distant content.

Single Cursor for In-Air Pointing

Pointable features one cursor that is positioned using *perspective-based pointing*, i.e. the cursor is placed at the intersection of the display plane and the nose-index vector (Figure 1). The nose-index vector is determined through two points in space: the location of the nose bridge, and the location of the index finger of the dominant hand. We added a dynamic offset to the cursor based on the nose-index vector to alleviate pointer occlusion by the hand; from perfect overlap, the offset increases proportionally with increased distance to the display plane. Perspective-based cursor positioning provides the user, as well as collaborators, a more accurate mental model of the mapping between hand location and click location [22]. This is in line with Kendon's work in social anthropology [14], which classified pointing gestures in the context of what is being pointed at.

In addition, while ray-casting and perspective-based pointing both devolve into a touch at a surface, perspective-based pointing transitions more smoothly [9], which is in accordance with our design goal to *augment touch*.

SideTrigger Gesture

Pointable interactions can only be activated when using the *SideTrigger* gesture. To acquire targets, a user points with the dominant hand's index finger while the middle, ring and little fingers are curled towards the palm (Figure 2). Bringing the thumb close to the second knuckle of the middle finger results in a click-down event. Moving it away generates a click-up event. Throughout, the palm faces and

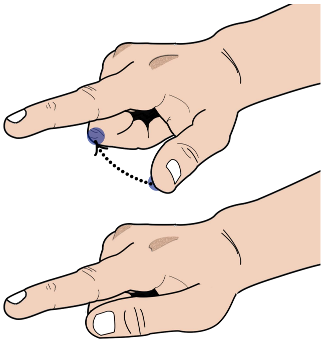


Figure 2. SideTrigger gesture.

stays parallel to the tabletop, avoiding occlusion of the targeted content, and closely mimics real-world pointing. *SideTrigger* is similar to the trigger gesture proposed by Grossman et al. [7] and *ThumbTrigger* [37], except the thumb strikes the side of the middle finger instead of on top of the index finger. Placing the thumb on the curled middle finger, rather than on the index, minimizes cursor jitter during clicking, while offering haptic feedback.

Dominant Hand to Select and Translate

On a horizontal tabletop, accessing out-of-reach content calls for precision, especially since the target appears to be smaller due to perspective distortion. Hence, the dominant hand was deemed more suited to this task. Simply moving the cursor over a target and clicking allows for translation.

Use of the Non-Dominant Hand to Scale and Rotate

Performing the *SideTrigger* gesture with the non-dominant hand, *in any location*, invokes manipulation, enabling in-place scaling and rotation of the acquired target. The center of manipulation is determined by the cursor position on the target. The relative motion between the index finger of each hand scales and rotates the target correspondingly.

Pointable alleviates some potential issues with in-air manipulation, as the user is only required to point at the target with a single hand. This reduces the probability of occlusion resulting from both hands pointing at the target and lowers overall muscular fatigue; the user may choose to rest the non-dominant arm on the tabletop surface. This is similar to the findings of Pierce et al. [28] who showed perspective-based pointing produced less fatigue than ray-casting when combined with waist level secondary manipulations.

Dynamic C/D Gain

Drawing on the concept of above the surface interactions [10], we decided to use the height above the table to vary the C/D gain. Increasing the vertical distance between the non-dominant hand and the tabletop surface increases the C/D gain of scaling and rotation transformations. At tabletop level, the C/D gain is 1. Following pilot studies, we limited the maximal C/D gain to 1.5 to avoid exaggerated transformations.

Thus, Pointable is an in-air interaction technique for tabletops with the following core characteristics: (1) single cursor, positioned by perspective-based pointing of the dominant hand; (2) *SideTrigger* gesture to click; (3) target acquisition and translation based on the cursor position; (4) scaling and rotation transforms based on the non-dominant hand's XY position; and (5) dynamic C/D gain through the non-dominant hand's Z position.

POINTABLE IMPLEMENTATION

We implemented Pointable with the Vicon motion capture system. We selected this technology over other systems that might be less obtrusive (e.g. the gloveless Kinect) because the Vicon offers higher 3D accuracy, a requirement for the performance measures of our three experiments.

Our system uses 8 Vicon T40 cameras to track passive IR retroreflective markers. Each marker is tracked at 100Hz, with an accuracy of 3mm in a room-sized 3D volume. The accuracy afforded by the Vicon system allows Pointable to recognize subtle gestures. Our interactive display is a 47" LED television mounted horizontally, running at a resolution of 1280x720. The experimental software was written in C# with WPF4.0.

To track motion and perspective with Pointable, we affixed marker arrangements on gloves and an eyeglass frame. The glasses are used to track the position and orientation of the head and the nose bridge. We also placed markers on each corner of the display to calculate the surface plane. This plane is raised to the height of the centroid of a marker on the tip of each user's index finger, allowing the system to determine whether a user has their finger within 3mm of the tabletop (a touch).

The perspective-based cursor is visualized as a circular icon with 30% transparency. The cursor diameter is approximately 7mm (17 px) at 1280x720 resolution, similar to the average touch-area recorded on a touchscreen [12]. Similarly, we calculate a 7mm circular area around the centroid of the finger marker and project it onto the display. The touch point is resolved to the center of the projected area.

EXPERIMENT 1: RECIPROCAL TAPPING TASK

We designed three experiments to evaluate Pointable. In our first, we evaluated the performance of participants in a Fitts' law tapping task [5]. Our primary objective was to compare the throughput of perspective-based pointing to touch. Additionally, we report on movement time and errors analyzed independently. Although the goal of Pointable is to *augment touch*, the performance of perspective-based pointing should establish it as a highly usable selection technique, while following a Fitts model tightly.

Task

Participants performed a variant of a Fitts' law tapping task [5] while sitting at the center of the long side of the table. Two bars, spanning the height of the table, appeared on the display. Participants were asked to tap or point between the two bars "as quickly and as accurately as possible". When the participant successfully selected the bar, it changed color from blue to green. For touch and perspective-based pointing within-reach, participants were seated as close to the table as comfortable. For out-of-reach perspective-based pointing, participants were seated such that their fin-

Interaction Technique	Model	R^2
Touch	-0.06 + 0.13 * ID	0.92
Pointing (Within-Reach)	-0.08 + 0.19 * ID	0.95
Pointing (Out-of-Reach)	-0.12 + 0.22 * ID	0.97
Mouse [6]	0.28 + 0.23 * ID	0.97
Touch [6]	0.46 + 0.12 * ID	0.93

Table 1. Fitts model and linear fit for each interaction technique.

gertips reached the edge of the table with a fully extended arm.

Two measures were recorded: *movement time* and *selection errors*. Movement time reports the time between two successful ‘taps’ within a target. Selection errors specify when the participant failed to successfully tap on the target. Movement times for trials with selection errors were excluded from the Fitts analysis.

Design

We used a 3x3x5 factorial repeated-measures within-subject design. The factors were: *interaction technique* (touch, perspective-based pointing within-reach and perspective-based pointing out-of-reach), *target width* (64, 92 and 128 pixels), and *target distance* (300, 500, 700, 900 and 1100 pixels). The target widths and distances correspond to Fitts’ law index of difficulties ranging between 1.7 and 4.2. Each participant performed 20 trials for each combination of factors, for a total of 900 trials (3 interaction techniques x 5 target widths x 3 target distances x 20 trials). We counter-balanced the interaction techniques first, and then counter-balanced among target widths and target distances. The experimental sessions lasted about 40 minutes. Participants trained with each interaction technique until they achieved less than 10% improvement between trials.

User Feedback. Participants were asked to rate perspective-based pointing and clicking based on whether it was *easy to use*. The questions were structured using a 5-point Likert scale (1=strongly disagree to 5=strongly agree). Additionally, participants were asked to rate whether touch was preferable to perspective-based pointing for within-reach conditions.

Participants. 12 participants between the ages of 21 to 30 took part in the study, as well as the following two studies. Each participant had some familiarity with multi-touch gestures, e.g., on a smartphone or a laptop. They were paid $20 for their participation in all three studies.

Hypotheses

We hypothesized that touch would have the highest throughput, followed by perspective-based pointing within-reach, and perspective-based pointing out-of-reach. This hypothesis was based on previous work that demonstrates that touch is faster than using a laser pointer from a distance in a Fitts’ law tapping task [20]. Perspective-based pointing is more accurate, though slower, than laser pointers [13], and therefore would not have as high a throughput as touch. We expected that within-reach perspective-based pointing would have a higher throughput than perspective-based pointing out-of-reach due to the greater accuracy afforded for identically sized targets.

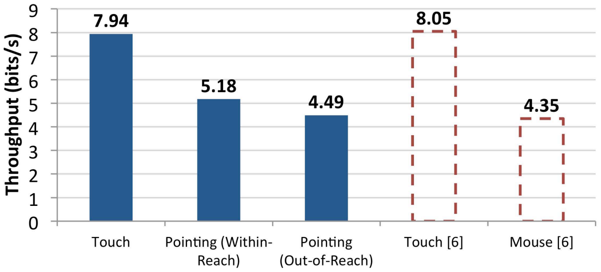


Figure 3. Throughput results for Experiment 1 (solid), compared to previous evaluation [6] (dashed).

Results

Fitts’ Law Analysis. We modeled the performance of each interaction technique using the Shannon formulation of Fitts’ law. In this form, the index of difficulty (ID) is a function of target distance (D) and target width (W). Movement time (MT) can be predicted as:

$$MT = a + bID, where\ ID = \log_2(\frac{D}{W} + 1)$$

where *a* and *b* are specific to a particular technique and are found using linear regression. Table 1 summarizes the fit for each interaction technique, as well as results by Forlines et al. [6] that set the baseline for touch and mouse performance on tabletops. Higher R^2 values indicate a close fit with the linear model. The index of performance (*IP)*, calculated as the reciprocal of *b,* is a measure of a technique’s throughput. Throughput, measured in bits per second, is independent of target width and distance. Figure 3 shows a comparison of the three measured interaction techniques as well as the previous results reported in Table 1.

Selection Time and Error Analysis. Independent analysis of width and distance in a Fitts’ law tapping task should be done cautiously, since width and distance are not independent factors — which is an assumption of an ANOVA. However, an analysis of interaction techniques and IDs does provide some insight. We analyzed the measures collected by performing a repeated measures factorial analysis of variance using *interaction technique* (5) x *ID* (15) on movement time and errors.

For movement time, the analysis showed a significant main effect for both *interaction technique* (F(2, 22)=73.33, p<0.001) and *ID* (F(14, 154)=140.83, p<0.001). Pairwise post-hoc tests with Bonferroni corrected comparisons between interaction techniques reveal that touch was significantly faster than both the perspective-based pointing conditions. For errors, the analysis showed a significant main effect for both *interaction technique* (F(2, 22)=30.96, p<0.001) and *ID* (F(14, 154)=11.22, p<0.001). Pairwise post-hoc tests with Bonferroni corrected comparisons between interaction techniques showed that touch had significantly fewer errors than both the perspective-based pointing conditions.

User Feedback. For the tapping task, 92% of participants found perspective-based pointing easy to use. 58% of participants agreed that touch was easier than perspective-based pointing within-reach.

Discussion

As hypothesized, touch is the fastest technique due to the nature of hitting a surface as a selection mechanism. The Fitts model of hand movement is divided into the 'distance-covering phase' and the 'homing-in phase' [38]. We believe the homing phase is primarily responsible for the difference between techniques. In the touch condition, the user is required to move their finger towards the surface, in addition to moving between the two targets. When using perspective-based pointing, the participant is not required to do so, and must home in on the target mid-air while synchronizing the invocation of the selection gesture. Having to strike this balance may have caused participants to slow down to ensure the cursor was on target before beginning the selection gesture. A benefit of direct touch is that the selection action is an integral part of the homing-phase and participants do not have to perform a deliberate selection action.

Within the two perspective-based pointing conditions, our prediction that sitting further back from the table would reduce throughput was correct. When pointing, the angle of motion between fixed distances was reduced when the participant sat out of reach. It seems that this should decrease movement times. At the same time, however, the perceptual width of the target was reduced, requiring the participant to be more accurate in placing the cursor on the target. In this comparison, we surmise that the decreased movement time during the distance covering-phase was not sufficient to overcome the increase within the homing-phase.

It is interesting to note that throughput measures for perspective-based pointing (4.49 bits/s and 5.18 bits/s) is similar to previously reported values for mice (4.35 bits/s [6], ~5.7 bits/s [18]). In addition to the benefits of perspective-based pointing previously outlined, it is encouraging to note that in single point scenarios, it can also serve as an alternative to a mouse for selecting distant targets, without sacrificing performance.

From the ratings and comments, we observed that participants found perspective-based pointing easy to use (92%). However, a few noted that the cursor had a slight delay. We believe this perception was triggered by the mapping of the cursor in close proximity to the participant's finger, in conjunction with the high-speed nature of the task. During normal use, this lag would be imperceptible.

EXPERIMENT 2: TARGET MANIPULATION

With a performance baseline set for perspective-based remote pointing, we wanted to compare the performance of multi-touch to Pointable in a standard translate/resize task defined by Forlines et al. [6]. We added a 45° rotation to the target to provide a more challenging and realistic abstraction of classic multi-touch photo sorting actions. Pointable was designed not to replace, but *augment touch* in situations where a user cannot access out-of-reach locations. Therefore, in this experiment, each interaction technique was evaluated on the part of the surface that highlighted its greatest strengths, the reachable half for touch, and the unreachable half for Pointable. The outcome of this experiment should support Pointable as a viable interaction technique for situations where touch cannot be applied.

Task

Participants were asked to point at or touch a start location, select the target, and then scale, rotate and drag it to a dock location "as quickly and as accurately" as possible. The distance between the start location and the target was equal to the distance between the target and the dock.

To prevent participants from anticipating the trial, only the start and dock locations initially appeared on the left side of the display. In half the trials, the dock was located *away* from the user with respect to the start location, and in the other half, the dock was located *towards* the user. To start the trial, the participant either touched the start location or pointed at it and performed a selection gesture, thereby causing the target to appear. The target was initially 1.5 times the size of the dock and rotated counter-clockwise at a 45° angle. To successfully dock, each participant was required to scale, rotate and drag the target inside the dock. Docking was considered successful if the target was of the correct size (within 5% of the dock size), correct orientation (within 2.5°), and if at least 63% of the target was placed inside the dock. The dock flashed orange when the target was within the acceptable margin of error for docking.

A car illustration was placed on the target to indicate the correct target orientation. To help participants assess target size, two arrows appeared on the target, pointing in the required direction of scaling (inwards if the target was too large, outwards if too small). The arrows disappeared if the target was the correct size. The color of the target changed from blue to green if the target was both the correct size and correct orientation. These features were implemented because we were primarily concerned with evaluating the motor, not perceptual, skills of our participants with respect to the two interaction techniques on each half of the table.

Three measures were collected: *selection time, manipulation time* and *docking errors*. Selection time represents the time it took to acquire the target after it appeared. If the participant did not successfully select the target on his or her first attempt, the trial was not recorded and was repeated. Manipulation time reports the time from selection to the time of successful docking, including the time spent scaling and rotating the target. The docking errors report the number of unsuccessful attempts at placing the properly scaled and rotated target into the dock.

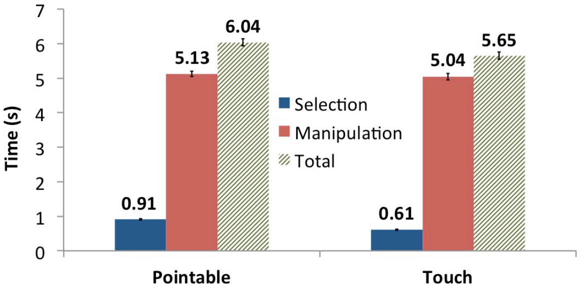


Figure 4. Selection, manipulation, and total times for Experiment 2.

Design

We used a 2x3x3x2 factorial repeated-measures within-subject design. Our variables were: *interaction technique* (multi-touch, Pointable), *target size* (64, 92 and 128 pixels), *target distance* (250, 400, and 550 pixels*)* and *docking direction* (towards or away). Each participant performed 3 trials per combination of factors, for a total of 108 trials (2 interaction techniques x 3 target sizes x 3 target distances x 2 docking directions x 3 trials). Participants were seated such that their maximum reach was the midpoint of the table length. We counter-balanced the interaction techniques first, then counter-balanced among digital variables (target size, target distance, docking direction). The experimental sessions lasted about 40 minutes. Participants trained until they achieved less than 10% improvement between trials.

User Feedback. Participants were asked to rate the two interaction techniques on whether target manipulation felt *easy to use*. In addition, we asked participants whether they found the ability to vary the rate of scaling and rotation (dynamic C/D gain for Pointable) compelling. Finally, to account for effects of depth perception, participants were asked if they felt the targets appeared to be the same size on both the reachable and unreachable halves of the table. The questions were structured using a 5-point Likert scale.

Hypotheses

Based on our predictions for throughput in Experiment 1, we hypothesized that multi-touch interaction would have faster selection times (H1). With respect to manipulation times, we expected touch to be faster overall (H2), although we predicted each technique would be faster in particular scenarios, producing interaction effects. We hypothesized that there would be an interaction between interaction technique and size, as Pointable would allow for more precise scaling and rotation (due to the dynamic C/D gain), providing faster manipulation times for the smallest targets (H3). We predicted that the direction of docking would affect both techniques, where docking away from the body would be slower (H4), and we hypothesized that docking away would result in more docking errors (H5). Finally, we predicted that both target size and target distance would have significant differences, with smaller targets and larger distances increasing manipulation time (H6). With respect to user feedback, we expected that almost all participants would report a disparity in target sizes for each half of the display (H7).

Results

Performance Analysis. We analyzed the measures collected by performing a repeated measures factorial analysis of variance (ANOVA) using *interaction technique* (2) x *target distance* (3) x *target size* (3) x *docking direction* (2) on selection time, docking time, and docking errors.

For selection time (Figure 4), the analysis showed that *interaction technique* was a significant factor ($F(1, 9)=15.60$, $p<0.05$). *Target size* ($F(2, 18)=22.37$, $p<0.001$) and *target distance* ($F(2, 18)=23.66$, $p<0.001$) were found to be significant factors. In addition, we found a significant interaction between *interaction technique* and *target size* ($F(2, 18)=9.62$, $p<0.05$) as well as *interaction technique* and *target distance* $F(2, 18)=9.11$, $p<0.05$).

For manipulation times, the analysis of variance showed that *docking direction* was a significant factor ($F(1, 9)=15.41$, $p<0.05$), with docking towards the participant's body having faster times. *Target size* ($F(2, 18)=17.53$, $p<0.001$) and *target distance* ($F(2, 18)=13.26$, $p<0.05$) were also found to be significant factors.

On docking errors, the analysis revealed *docking direction* as a significant factor ($F(1, 9)=19.67$, $p<0.05$) with docking away from the participant's body resulting in more errors.

User Feedback. We observed that 92% of participants found both multi-touch and Pointable easy to use for scale, rotate and drag operations. 82% of participants found the ability to dynamically change the C/D gain compelling. When asked if their perceptions of the target sizes were identical on both halves of the table, 58% of participants agreed with the statement.

Discussion

Results demonstrate that Pointable can serve as a substitute in situations where touch cannot be used at all, or without discomfort, without sacrificing performance.

The observed selection times both reinforced our results from Experiment 1 and confirmed that touch would be faster than pointing (H1). Although we expected touch to be faster overall with respect to manipulation times (H2), we did not observe a main effect of interaction technique in the statistical analysis, meaning performance did not differ significantly between the touch and Pointable conditions.

Our hypothesis that docking direction would significantly impact manipulation times was confirmed (H4). Although this result affected both techniques, we believe it was for different reasons. When docking away, the effort required to reach out with the hands increased manipulation times for touch. For Pointable, the heightened perspective distortion made the task more difficult when docking away. It is clear that docking away from the body requires more physical effort, causing a significantly different number of errors in both cases (H5), and increasing manipulation times. This was confirmed by our user feedback with several comments stating that participants found it easier to dock towards them.

We confirmed our prediction that both target size and docking direction would have a significant effect on manipulation (H6). However, we did not find an interaction effect with respect to interaction technique and target size (H3). Smaller targets were, overall, more difficult to manipulate.

Our questionnaires indicated 92% of participants found scaling, rotating and dragging using either touch or Pointable easy to use. However, participants felt somewhat more strongly about touch (average rating of 4.5 vs 4.1). Comments suggested that breaking the requirement of needing to point at the target with the non-dominant hand made Pointable less intuitive compared to direct touch, but allowed for greater precision and reduced occlusion of the target. 83% of participants found the dynamic C/D gain to be compelling and useful for completing the task.

Contrary to our expectations (H7), 58% of participants reported that the targets appeared to be of identical size on both halves of the display. This may be because participants adapted to the Pointable condition reducing apparent effects of distortion on perception.

Pointable was designed to *augment touch*. The results indicate that, in isolation, Pointable can perform the same task as touch in a distant location, yet achieve similar performance.

EXPERIMENT 3: BEHAVIORAL EVALUATION

The primary design goals of Pointable were to *augment touch* interaction on tabletops, to allow users to *manipulate content in-place,* while *minimizing modality switches*. Given these motivations, we wanted to observe the behavior of participants when they were free to choose their interaction technique at any given moment during each trial of a scale, rotate and drag task spanning the full length of the table. We presented participants with a range of scenarios, where the target and dock could each appear in locations that were within-reach or out-of-reach. For each scenario, participants could use touch or Pointable, or both. The only restriction we imposed was that all participants had to stay seated, and were positioned such that their maximum reach was at the midpoint of the table's length. In some conditions, the target appeared at this midpoint location - inconvenient to reach, yet possible when leaning.

Task

As in Experiment 2, participants were asked to point or touch a start location, select the target, scale, rotate and drag it to a dock location, with the same docking tolerance. The start and dock locations appeared on the top-left and bottom-left of the entire surface, and would again swap positions. We recorded the loci where participants manipulated the target and with which interaction technique.

Design

We used a 3x3x2 factorial repeated-measures within-subject design. Our variables were: *target size* (64, 92 and 128 pixels), *target position* (easily reachable, reachable with leaning, and unreachable) and *docking direction* (towards and away). Each participant performed 3 trials per combination of factors, for a total of 54 trials (3 target sizes x 3 target positions x 2 docking directions x 3 trials). Randomization and training was performed as in Experiment 2. The experimental sessions lasted about 30 minutes.

User Feedback. Participants were asked to report the technique (multi-touch or Pointable) they preferred for scale and rotate operations when the target appeared in the midpoint of the table (reachable with leaning). In addition, we asked participants to rate whether they preferred to acquire targets using remote pointing rather than reaching or walking, and if they found the remote target manipulation a compelling extension of touch interaction for distant targets. The questions were structured using a 5-point Likert scale.

Hypotheses

Our predictions for choice of interaction technique depended on the scenario the participant was presented with.

Dock and Target Appeared on Same Half. We hypothesized that participants would exclusively use the technique optimized for the relevant side of the table: using touch up close, and Pointable at a distance (H1).

Dock and Target Appeared on Opposite Halves. We expected that participants would resize and rotate the target using multi-touch and use Pointable to translate (H2).

Target Appeared at the Mid-Point of Table. At this distance, participants would have to lean over or stretch to touch the target. Therefore, we predicted that participants would use Pointable to acquire the target, but then scale and rotate based on the dock location (Similar to H1, dock towards – touch, dock away – Pointable) (H3).

Results

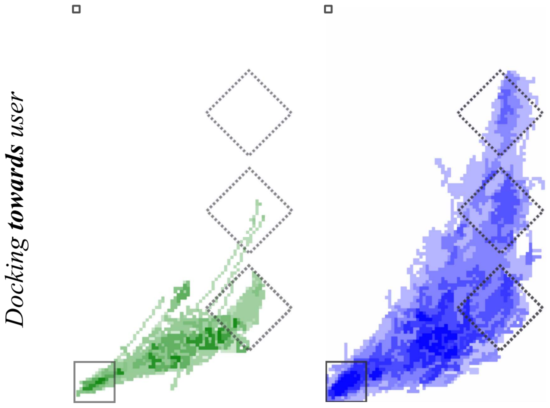
Behavioral Analysis. Figure 5 presents a map of the locations where participants manipulated (dragged, scaled and rotated) the target. We separated the maps based on two variables: interaction technique, and direction of docking.

User Feedback. For the cases where the target appeared reachable when leaning, 92% of participants reported that they preferred using Pointable when the dock was on the far edge of the table. When the dock was on the close edge of the table, 75% reported that they preferred Pointable. 83% of participants found that Pointable was a compelling addition to multi-touch interaction.

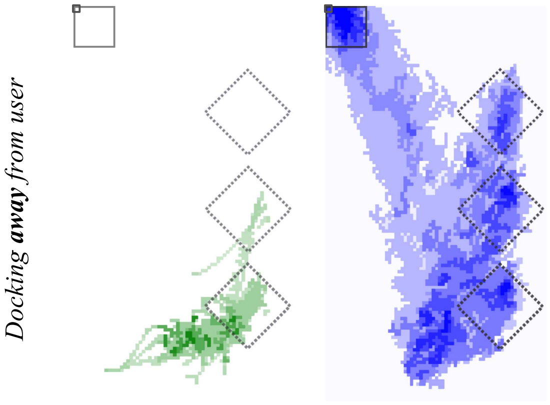
Discussion

The results indicate that Pointable can be used in conjunction with direct-touch, not only in situations where touch cannot be used without inconvenience to the user, but also in cases where less occlusion and finer control with Pointable make it preferable.

The interaction maps on Figure 5 (a) and (c) confirm results from Toney and Thomas [36] who reported that over 90% of direct-touch interaction was performed within a 34 cm range in front of the participant, which corresponded to



(a) Touch (b) Pointable



(c) Touch (d) Pointable

Figure 5. Interaction maps for each technique. Darker shades represent more manipulations in that location. Solid square shows dock location. Dashed diamonds show initial target configurations for the largest target size. All three target sizes had common centers. Participants were seated at the bottom edge.

28% percent of the total length of their table. Our interaction maps show that most of the touch interaction was limited to less than 33% of the length of the table, with a 'hot spot' (dark area in Figure 5) centered in front of the user. Notably, these dark spots also appear in similar locations for Pointable (Figure 5 (b) and (d)). This area remains a 'personal area' [33] for manipulation, regardless of interaction technique.

For the conditions when the dock and target appeared on the same half of the table, our prediction that the participants would use the technique appropriate for that half was mostly correct (H1). Participants used multi-touch to manipulate in the closer half (Figure 5 (a)) and Pointable in the further half (d). However, several participants also chose to use Pointable when both the dock and target appeared close to them, causing a less discrete divide in strategies.

For the cases when the dock and target appeared on opposite halves of the table, we did not observe the pattern of behavior we expected (H2). Strategies varied widely. We observed that 33% of participants completed the task in the manner hypothesized, i.e. using touch for scaling and rotation (Figure 5 (c), dark green patches), while another 33% chose to use Pointable almost exclusively, opting to avoid all modality switches. The rest mixed the two techniques. This strategy can be more easily seen in Figure 5 (d) where, despite the availability of multi-touch, participants used Pointable in their 'personal area' to scale and rotate.

The sparse number of touch points for the middle targets in Figure 5 (a) and (c) indicates that participants chose to acquire middle targets predominantly using Pointable (H3). However, technique choice was split with respect to scaling and rotation. We believe that Pointable makes the acquisition of targets less demanding, even those in the vicinity of the user reachable by touch.

One emergent theme within Figure 5 is that participants used Pointable more than touch interaction. It is important to note that at some point during every trial, the participant was required to use perspective-based pointing, although not necessarily to interact with the target. This either involved clicking on the start location to begin the trial, or to dock the target, in both cases on the far side of the display. Some of the imbalance may be attributed to this design.

However, several comments acquired after this experiment referred back to the high degree of precision afforded by Pointable (also shown in Experiment 2). Some noted that with Pointable, occlusion was reduced significantly when scaling and rotating the smallest targets, thus participants chose to continue using Pointable in situations where they could have used multi-touch. The user feedback indicating that only 25% of participants preferred to use multi-touch when the dock was close reflects these situations.

Fatigue issues normally associated with in-air pointing did not deter participants from opting to use Pointable. We believe this can be attributed to three aspects of Pointable: pointing with only a single hand, even during scaling and rotation; pointing without raising the arm above the shoulder; and the option to rest the non-dominant hand on the tabletop itself. However, as Experiment 3 only lasted 30 minutes, extended sessions may reveal a different trend in the ratio of Pointable interactions to touch input.

CONCLUSION

In this paper, we introduced Pointable, an in-air, asymmetrical bimanual object manipulation technique that augments touch input on a tabletop for distant content. Pointable has a single cursor, determined by perspective-based pointing of the dominant hand, and uses the *SideTrigger* gesture to click. Pointable allows for target acquisition and translation based on the cursor position, while scaling and rotation transforms are based on the non-dominant hand's XY position, and offers a dynamic C/D gain through the non-dominant hand's Z position. Pointable was designed to realize the following goals: to *augment touch*, *minimize modality switches*, *in-place manipulation*, *low fatigue*, and be *unobtrusive*.

We evaluated Pointable in three experiments designed to test these goals. The first experiment demonstrated that perspective-based pointing has throughput measures within the previously reported range of mouse performance and therefore can serve as a highly performing technique for distant target selection. The second experiment showed that Pointable fulfilled the design goal of *in-place manipulation* by establishing that Pointable can perform as well as multi-touch in a scale, rotate and drag task on the unreachable section of the table. The third experiment established that Pointable can be used in conjunction with multi-touch, fulfilling the design goals of *augmenting touch, low fatigue* and *minimizing modality switches*.

We designed Pointable keeping collaborative settings in mind, with the design goal of being *unobtrusive*. However, this paper did not evaluate Pointable within a collaborative scenario and therefore needs further exploration and a thorough collaborative evaluation to verify that this design goal was met. In addition to making the system multi-user, Pointable could gain from an accurate, uninstrumented (gloveless) system implementation, which might encourage casual use.

REFERENCES

1. Abednego, M., Lee, J., Moon, W., and Park, J. I-Grabber : Expanding Physical Reach in a Large-Display Tabletop Environment Through the Use of a Virtual Grabber. *Proc. ITS*, (2009), 61-64.
2. Baudisch, P., Cutrell, E., Robbins, D., et al. Drag-and-pop and drag-and-pick: Techniques for accessing remote screen content on touch-and pen-operated systems. *Proc. INTERACT*, (2003), 57-64.
3. Bezerianos, A. and Balakrishnan, R. The vacuum: facilitating the manipulation of distant objects. *Proc. CHI*, (2005), 2–7.
4. Dietz, P. and Leigh, D. DiamondTouch: a multi-user touch technology. *Proc. UIST*, (2001), 219–226.
5. Fitts, P.M. The information capacity of the human motor system in controlling amplitude of movement. *Journal of Experimental Psychology 47*, (1954), 381-391.
6. Forlines, C., Wigdor, D., Shen, C., and Balakrishnan, R. Direct-touch vs. mouse input for tabletop displays. *Proc. CHI*, (2007), 647-656.
7. Grossman, T., Wigdor, D., and Balakrishnan, R. Multi-finger gestural interaction with 3d volumetric displays. *Proc. UIST*, (2004), 61–70.
8. Guiard, Y. Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. Journal of Motor Behavior. (1987), 486-517.
9. Hill, A. and Johnson, A. Withindows: A Framework for Transitional Desktop and Immersive User Interfaces. *IEEE SI3D*, (2008), 3-10.
10. Hilliges, O., Izadi, S., Wilson, A., Hodges, S., Garcia-Mendoza, A., and Butz, A. Interactions in the Air : Adding Further Depth to Interactive Tabletops. *Proc. UIST*, (2009), 139–148.
11. Hinckley, K., Yatani, K., Pahud, M., et al. Pen + Touch = New Tools. *Proc. UIST*, (2010), 27-36.
12. Holz, C. and Baudisch, P. The generalized perceived input point model and how to double touch accuracy by extracting fingerprints. *Proc. CHI*, (2010), 581-590.
13. Jota, R., Nacenta, M.A., Jorge, J.A., Carpendale, S., and Greenberg, S. A comparison of ray pointing techniques for very large displays. *Proc. GI*, (2010), 269–276.
14. Kendon, A. *Gesture: visible action as utterance.* Cambridge University Press, 2004.
15. Khan, A., Fitzmaurice, G., Almeida, D., Burtnyk, N., and Kurtenbach, G. A remote control interface for large displays. *Proc. UIST*, (2004), 127-136.
16. Microsoft Kinect. http://www.xbox.com/en-US/kinect.
17. Latulipe, C., Kaplan, C.S., and Clarke, C.L.A. Bimanual and unimanual image alignment: an evaluation of mouse-based techniques. *Proc. UIST*, (2005), 123–131.
18. MacKenzie, S. and Isokoski, P. Fitts' throughput and the speed-accuracy tradeoff. *Proc. CHI*, (2008), 1633-1636.
19. Microsoft Surface. http://www.microsoft.com/surface/.
20. Myers, B., Bhatnagar, R., Nichols, J., et al. Interacting at a distance: measuring the performance of laser pointers and other devices. *Proc. CHI*, (2002), 33-40.
21. Myers, B.A. and Buxton, W. A Study in Two-Handed Input. *Proc. CHI*, (1986), 321-326.
22. Nacenta, M., Pinelle, D., Stuckel, D., and Gutwin, C. The effects of interaction technique on coordination in tabletop groupware. *Proc. GI*, (2007), 191-198.
23. Nancel, M., Wagner, J., Pietriga, E., Chapuis, O., and Mackay, W. Mid-air pan-and-zoom on wall-sized displays. *Proc. CHI*, (2011), 177-186.
24. Oblong Industries. http://www.oblong.com/.
25. Parker, J.K., Mandryk, R.L., Nunes, M.N., and Inkpen, K.M. TractorBeam selection aids: Improving target acquisition for pointing input on tabletop displays. *Proc. INTERACT*, (2005), 80–93.
26. Parker, J.K., Mandryk, R.L., and Inkpen, K.M. TractorBeam: seamless integration of local and remote pointing for tabletop displays. *Proc. GI*, (2005), 33–40.
27. Pierce, J., Forsberg, A., Conway, M., Hong, S., Zeleznik, R.C., and Mine, M.R. Image plane interaction techniques in 3D immersive environments. *Proc. I3DG*, (1997), 39-43.
28. Pierce, J. and Pausch, R. Comparing voodoo dolls and HOMER: exploring the importance of feedback in virtual environments. *Proc. SIGCHI*, (2002), 105–112.
29. Pinelle, D., Barjawi, M., Nacenta, M., and Mandryk, R. An evaluation of coordination techniques for protecting objects and territories in tabletop groupware. *Proc. CHI*, (2009), 2129-2138.
30. Reetz, A., Gutwin, C., Stach, T., Nacenta, M., and Subramanian, S. Superflick: a natural and efficient technique for long-distance object placement on digital tables. *Proc. GI*, (2006), 163–170.
31. Rekimoto, J. SmartSkin: an infrastructure for freehand manipulation on interactive surfaces. *Proc. CHI*, (2002), 113–120.
32. Ringel, M., Berg, H., Jin, Y., and Winograd, T. Barehands: implement-free interaction with a wall-mounted display. *Proc. CHI EA*, (2001), 368-374.
33. Scott, S.D., Carpendale, S., and Inkpen, K.M. Territoriality in collaborative tabletop workspaces. *Proc. CSCW*, (2004), 294-393.
34. Shoemaker, G., Tang, A., and Booth, K.S. Shadow Reaching: A New Perspective on Interaction for Large Wall Displays. *Proc. UIST*, (2007), 53-56.
35. SMART Technologies. http://www.smarttech.com/.
36. Toney, A. and Thomas, B.H. Applying reach in direct manipulation user interfaces. *Proc. OZCHI*, (2006), 393-396.
37. Vogel, D. and Balakrishnan, R. Distant freehand pointing and clicking on very large, high resolution displays. *Proc. UIST*, (2005), 33-42.
38. Welford, A.T. *Fundamentals of Skill*. Methuen, London, 1968.
39. Wigdor, D., Benko, H., Pella, J., Lombardo, J., and Williams, S. Rock & rails: extending multi-touch interactions with shape gestures to enable precise spatial manipulations. *Proc. CHI*, (2011), 1581–1590.
40. Wilson, A.D. and Benko, H. Combining multiple depth cameras and projectors for interactions on, above and between surfaces. *Proc. UIST*, (2010), 273–282.
41. Wilson, A.D. TouchLight: an imaging touch screen and display for gesture-based interaction. *Proc. Multimodal Interfaces*, (2004), 69–76.
42. Wu, M. and Balakrishnan, R. Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. *Proc. UIST*, (2003), 193-202.