# Characterizing User Performance with Assisted Direct Off-Screen Pointing

**Barrett Ens**
University of Manitoba
Winnipeg, Canada
bens@cs.umanitoba.ca

**David Ahlström**
University of Klagenfurt
Klagenfurt, Austria
david@isys.uni-klu.ac.at

**Andy Cockburn**
University of Canterbury
Christchurch, New Zealand
andy@cosc.canterbury.ac.nz

**Pourang Irani**
University of Manitoba
Winnipeg, Canada
irani@cs.umanitoba.ca

## ABSTRACT

The limited viewport size of mobile devices requires that users continuously acquire information that lies beyond the edge of the screen. Recent hardware solutions are capable of continually tracking a user's finger around the device. This has created new opportunities for interactive solutions, such as *direct off-screen pointing:* the ability to directly point at objects that are outside the viewport. We empirically characterize user performance with direct off-screen pointing when assisted by target cues. We predict time and accuracy outcomes for direct off-screen pointing with existing and derived models. We validate the models with good results ($R^2 \geq 0.9$) and reveal that direct off-screen pointing takes up to four times longer than pointing at visible targets, depending on the desired accuracy tradeoff. Pointing accuracy degrades logarithmically with target distance. We discuss design implications in the context of several real-world applications.

### Author Keywords

Direct off-screen pointing, off-screen target visualizations, performance models, Fitts' law, steering law.

### ACM Classification Keywords

H5.2 [Information interfaces and presentation]: User Interfaces – Input devices and strategies

### General Terms

Performance, Experimentation, Human factors.

## INTRODUCTION

Most workspaces extend beyond users' available viewport into *off-screen space*, particularly on small screen mobile devices. Scrolling, panning and zooming are commonly used to manipulate the content of the viewport, but hardware innovations are enabling new forms of interaction. In particular, range sensors and similar technology have recently been added to devices to support interaction in their surrounding space. For example SideSight [5] and similar devices [15, 18] allow users to interact with on-screen content by extending the region of input into space around the device. Little attention has been given to direct interaction in the surrounding off-screen space, despite empirical evidence supporting the use of direct kinesthetic manipulations for improving spatial memory [19, 26, 27].

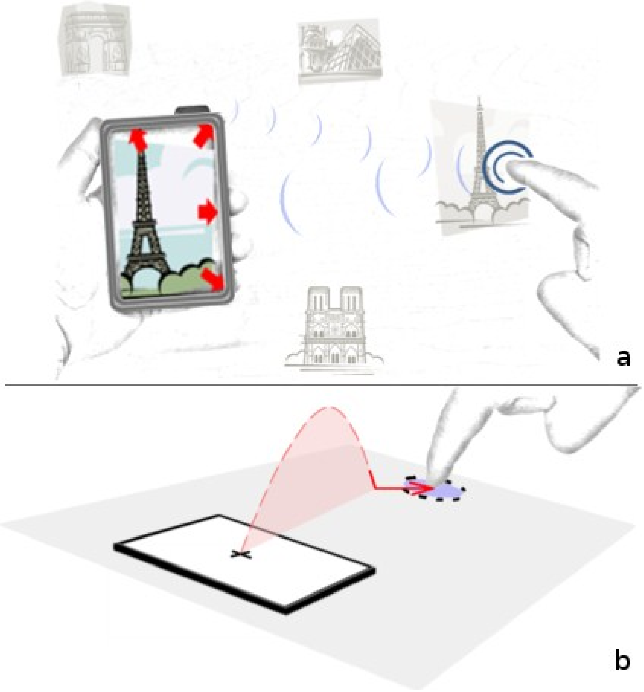

**Figure 1. (a) A conceptual depiction of assisted direct off-screen pointing; arrows indicate the direction of targets beyond the viewport. (b) The launch-adjust interaction, central to our study's focus, enacted on a simulated device.**

Direct input techniques involve a one to one correspondence between the location of the input device (such as a finger) and the location of the virtual object. We use the term *direct off-screen pointing* to describe the ability to directly select objects that lie outside the viewport. Direct off-screen pointing can allow familiar item locations to be specified without visual feedback (e.g., [13]), but on-screen targeting assistance, via visual cues for off-screen targets, or *off-screen target cues*, is required for items whose whereabouts are previously unknown. This paper concerns the latter, which we call *assisted direct off-screen pointing* (Figure 1a).

*MobileHCI 2011,* Aug 30–Sept 2, 2011, Stockholm, Sweden.

Due to the novelty of these interactions, there is a lack of understanding of the human factors that influence their design and efficiency. Relevant research questions include

- How quickly and accurately can users point at off-screen items?
- How do known properties of different visualization cues affect performance?
- What theoretical models describe performance?
- Is performance influenced by the location of the target around the viewport?
- How does performance vary with target distance and width?

To examine these questions, we conducted an empirical study using a touch-sensitive table to emulate the interaction space around a space-sensing device (Figures 1b, 3). Results show that assisted direct off-screen pointing can be as much as four times slower than on-screen pointing, although with an expected speed-accuracy tradeoff. Spatial granularity of selectable regions around the device ranges from dense (≤ 4cm) with visual assistance to a minimum of three distinguishable targets within arm's reach without the assistance of cues. Time and accuracy are strongly influenced by the nature of the visual targeting assistance. Fitts' law [9] and the steering model [1] delineate the lower and upper bounds of direct off-screen pointing times.

Our contributions in this paper are: 1) a performance model for assisted direct-off screen pointing; 2) a characterization of pointing accuracy and pointing time with assisted direct off-screen pointing; 3) a quantification of targeting error, including target size requirements for accurate pointing; 4) design implications.

## RELATED WORK

Our work builds on recent results in pointing situations where the target is not immediately visible. We discuss off-screen target visualizations that can serve to guide direct off-screen pointing and we also review hardware-based prototypes that enable direct off-screen pointing.

### Acquiring Targets That Are Not Visible

Because portions of documents often lie outside the viewing range, there are numerous techniques that assist with off-screen target acquisition. Three general strategies for acquiring off-screen targets are to shift the workspace so that the target lies within the viewport (e.g., scrolling and panning); to scale the workspace so that the viewport encloses the target region (e.g., zooming); to move the viewport to the target (e.g., Peephole pointing [20, 25, 29]).

Scrolling performance follows Fitts' law if the user knows where the target resides ahead of time, but when the user is dependent on visual search, scrolling time is linearly dependent on the distance to the target [16]. Peephole pointing performance is more complex, with two distinct components: one for moving the viewport to make the target visible, and a second part for moving the cursor to select the target [6]. This two part model can be captured with a variation on Fitts' law. The index of difficulty (*ID*) has one term defined by $\log_2(A/W)$ and another term defined by $\log_2(A/S)$, where $A$ and $W$ are the target amplitude and width, and $S$ is the viewport size.

### Visual Cues for Off-screen Objects

Several techniques can direct users' attention to the location of objects that reside outside of the viewport.

Contextual views such as arrows, CityLights [30], or EdgeRadar [14] represent off-screen object distances by means of abstract shapes or proxies that appear on the edges of the viewport. These are common in GPS devices and games. They clearly provide target direction but require either a legend or an appended numerical value to communicate the target's distance.

Halo [2] and Wedge [12] make off-screen target distances perceptually immediate and improve on the limited distance awareness of contextual views. Their designs rely on users' ability to complete the cue when part of it is outside the viewport and point at the exact location of the object. By interpreting the cue, users can identify the location of the off-screen target.

Overviews [28] take up a portion of on-screen space to show a geometrically scaled down version of the entire workspace. This allows users to assess not only the location of various off-screen objects but also their interrelations. Overviews are effective [22], but consume precious screen real estate on small devices.

Studies have shown relative benefits for each technique category with different tasks. For instance, for tasks that order targets from closest to farthest, scaled arrows work better than Halos [4]. For accurate positioning Halos or Wedges are preferred over arrows [2]. Not surprising, when inferences about relationships between targets are needed, overviews are the technique of choice [17].

### Prototype Sensor-Equipped Devices

The feasibility of direct off-screen pointing is demonstrated by several recent prototypes that use sensors to expand the interactive space around a device.

SideSight [5] consists of a touch screen mobile device with side-facing infrared sensors. The device has a row of ten sensors on the left and right sides and is intended for use when placed on a flat surface. The sensors can detect a finger up to a range of about eight centimeters. Hoverflow [18] similarly uses a coarse grained infrared sensor input. This prototype uses six distance sensors aligned with the face of the device to respond to gestures above the screen.

Abracadabra [15] uses a magnetic sensor to allow high resolution input on a miniature device, although a small magnet must be worn on the user's fingertip. A usability study with Abracadabra conducted on a radial input task, showed significant improvement over direct manipulation of the touch screen.

Such innovations could soon become predominant on mobile devices. To support direct off-screen pointing, designers need a characterization of this interaction space. We use object targeting, a fundamental building block of direct manipulation, to begin this investigation.

Although current prototypes have limited granularity, we imagine a hypothetical device whose sensing fidelity will not constrain our design. For various reasons (see Study Apparatus, below) we emulate the space-sensing device on a touch-sensitive table. A tangible surface provides an abstract setting where the plane of interaction is precisely defined, and eliminates confounding factors that arise from untested prototype technology.

## ASSISTED DIRECT OFF-SCREEN POINTING

We focus our study on a two-phase interaction with a planar-sensing device (i.e. the device can sense interaction only within an extended plane defined by its display screen) emulating technology similar to that demonstrated by SideSight [5]. In the first phase, the pointing finger *launches* above and onto the surface where the device rests. 'Touching' the device's interactive plane activates a dynamic visualization cue, which can be used to *adjust* the pointing finger's position to the target. We call this two-phase interaction *launch-adjust* (Figure 1b).

Given this setup, we define a second pointing technique, the *glide* interaction. In this case the user maintains contact with the plane of interaction (i.e. the touch-table) throughout the entire pointing motion with the benefit of continuous dynamic feedback. We introduce the glide interaction for two reasons: 1) We are primarily interested in characterizing performance with dynamic cues, thus the glide represents a 'magnified' and controlled instance of the adjustment phase of the launch-adjust interaction. 2) The continuous feedback of the glide motion may provide some insight into the case of a fully 3D-capable sensing device.

Interacting in unfamiliar space requires visual assistance, which can be provided with off-screen target cues. In our study, we choose the well-studied techniques Wedge and overview+detail (Figure 2) to provide interactive feedback to the user about the positions of a target and the sensed pointing finger (Figure 3). These cues can be either static or dynamic, depending on the situation. A static cue provides information about the position of a stationary object. In our case, the static cue indicates the target's location, and is available to the user prior to initiating a selection. A dynamic cue adjusts according to the motion of an object (or the relative motion of a device and workspace). We use dynamic cues to indicate the user's tracked pointing finger, initiated by the touch down following a launch, or provided continuously in the glide interaction. We borrow the term *information scent,* or just *scent,* from information foraging theory [24] to refer to the amount of information given by a combination of static and dynamic cues. Some cues, like Wedge, contain a distance component, while overview+detail cues contain additional scent information, such as relative position.

Due to Wedge's poor performance at long range and with corner selections (from [11, 12] and a pilot), we annotated the static cue with the target's distance from the screen edge, (Figure 3). We adjusted the Wedge aperture and intrusion parameters from the original values (Figure 2) used by [12] to accommodate greater distances, and to make the leg length roughly constant regardless of rotation.

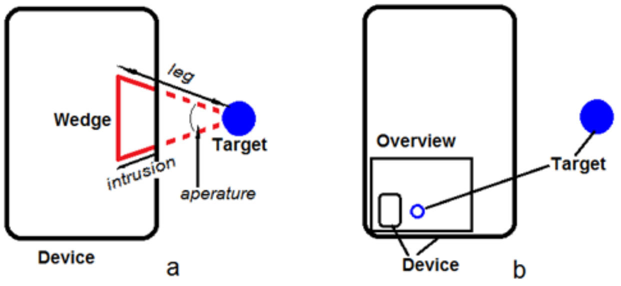

**Figure 2. Off-screen target cues a) Wedge and b) overview+detail. The overview is a miniature representation of the workspace [28]. Wedge's leg length, aperture and intrusion distance are a function of target distance [12]:**
**leg ≈ ln (dist)**
**aperture ≈ dist / leg**

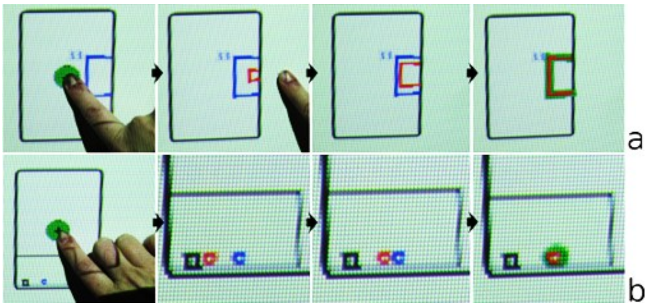

**Figure 3. A user selects an off-screen target using the glide interaction. Assisted by visualization cues (annotated Wedge, top and overview+detail, bottom), the user adjusts their finger position until the target is highlighted. Static target cues are blue and dynamic cues are red (best viewed in color).**

## MODELS FOR DIRECT OFF-SCREEN POINTING STYLES

### Model for the Glide Interaction

Target acquisition tasks are traditionally modeled with two parameters, *A* and *W*. *A* is the Euclidian distance between the starting position of the pointing finger and the position of the target center. *W* is the target width.

Fitts' law has successfully been applied to pointing tasks, including those involving sustained motions, such as drag-and-drop [10]. The movement time is a function of the target distance and width, given in its widely used form as

$$MT = a + b\log_2\left(\frac{A}{W} + 1\right) \tag{1}$$

where *a* and *b* are empirically determined constants. The application of Fitts' law is based on the assumption that a user relies on visual stimulus received from the location of both the pointing limb and the target [7]. This assumption holds if the target is visible, but not when the user must rely solely on visual cues. In the latter case, we expect time vs. distance performance to degrade with loss of scent. Our pilot studies show a liner response when the target cues are arrows with no distance information. This linear performance can be modeled using the steering law:

$$MT = a + b\left(\frac{A}{W}\right) \tag{2}$$

Whereas a pointing task modeled by Fitts' law begins with a large ballistic impulse, the steering law is applicable to tasks where users continually modify their motion path in response to changing stimulus throughout the entire motion [1]. The latter model is better suited to the task of assisted direct off-screen pointing, as the user is continuously inspecting the dynamic cue in relation to the static cue. For example, with Wedge, users need to match the pointing finger's dynamic Wedge to the static one. In practice, the user steers the inner dynamic Wedge within a virtual tunnel defined by the legs of the outer static Wedge (Figure 3).

Given a cue with reasonable direction and distance information, we can expect the performance to be better than the worst-case example of the arrow, but less than the ideal situation of a visible target. Therefore, we can view the Fitts' law (Eq. 1) and the steering law (Eq. 2) as upper and lower bounds, respectively, of an assisted pointing task using the glide interaction.

### Model for the Launch-adjust Interaction

For target acquisition using the launch-adjust pointing style, we split the movement time *MT* into two phases (Figure 4): a launch phase where users *launch* their pointing finger above and into a point in the interactive plane (traversing distance $A_L$), and a correction phase, where users *adjust* their pointing position to find the target (distance $A_C$). We call these durations of these phases the launch time, $T_L$ and the adjustment or correction time, $T_C$:

$$MT = T_L + T_C \tag{3}$$

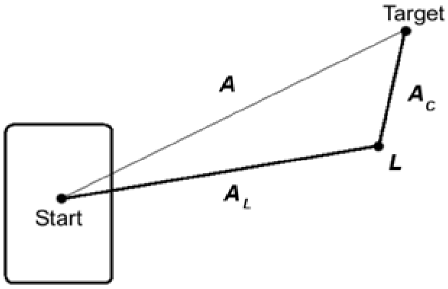

**Figure 4. Launch-adjust: the launch takes the pointing finger from the device (lower left) a distance AL into the interaction plane. AC is the pointing error. The adjustment phase completes the traversal to the target.**

We examine each phase in detail. Fitts' law holds for tasks where a target is not visible, but whose whereabouts are known [16]. Prior to the launch motion, the user is aware, with reasonable accuracy, of the target's location through the static off-screen target cue. If a cue contains sufficient scent, the movement time of the launch motion should follow Fitts' law:

$$T_L = a_L + b_L \log_2\left(\frac{A}{W} + 1\right) \tag{4}$$

In the adjustment phase, users are guided with dynamic feedback, as in the glide task. The initial landing point, *L*, and the remaining target distance (or *launch error*), $A_C$, both depend on the accuracy of the initial launch motion (i.e. a more accurate launch will result in a smaller error). Adjustment time, $T_C$, is thus a function of the launch error, which is proportional to the actual target distance. Thus,

$$T_C = f(A_C) \tag{5}$$

and

$$A_C = g(A) \tag{6}$$

As in the glide model, the adjustment time is bounded by logarithmic (Fitts') and linear (steering) functions, resulting in the following derivations: For Fitts' law, substituting (1) for $A_C$ in (5) gives

$$T_C = a_C + b_C \log_2\left(\frac{A_C}{W} + 1\right) \tag{7}$$

We then substitute (6) into (7) to get a new formula

$$T_C = a_C + b_C \log_2\left(\frac{g(A)}{W} + 1\right) \tag{8}$$

A similar substitution of (2) provides an alternative for the steering model:

$$T_C = a_C + b_C\left(\frac{g(A)}{W}\right) \tag{9}$$

Finally, to get a complete picture of the two-phase launch-adjust motion, we substitute (4) and (8) into (3):

$$\begin{aligned} MT &= a_L + b_L \log_2\left(\frac{A}{W} + 1\right) + a_C + b_C \log_2\left(\frac{g(A)}{W} + 1\right) \\ &= a + b_L \log_2\left(\frac{A}{W} + 1\right) + b_C \log_2\left(\frac{g(A)}{W} + 1\right) \qquad (10) \end{aligned}$$

where $a = a_L + a_C$. Likewise, (4) and (9) into (3) produces

$$MT = a + b_L \log_2\left(\frac{A}{W} + 1\right) + b_C\left(\frac{g(A)}{W}\right) \tag{11}$$

The rate of growth in function *g* is unknown although we can determine its form empirically. Pointing error tests in [11] show that the growth function has a logarithmic component, so we estimate *g*(*A*) with equation (12) where $a_e$ and $b_e$ are constants that describe error growth:

$$A_C = a_e + b_e \log_2 A \tag{12}$$

## STUDY

The purpose of our empirical evaluation was twofold: (1) to validate the models described above; and (2) to observe the characteristics of direct off-screen pointing when assisted by static and dynamic target cues. The latter goal can be split into three components: (a) what target size is required to achieve a high percentage of direct hits (i.e. without the adjustment motion) in response to static cues; (b) how does this required size vary across space?; and (c) how is performance affected by dynamic visualizations for off-screen pointing with varying types of information scent?

### Apparatus

We used a SMART board with a top projected display measuring approximately 130×104cm with a resolution of 1280×1024 pixels. The display was inclined toward the user at an angle of about 23° (Figure 5).

Our simplified setup reduces experimental noise in a manner similar to [6] which replaced a spatially aware device with a Wacom tablet. This ensures reliable data, which is critical in assessing the models we derived. Additionally, the use of a constrained surface avoids latency, external wires/markers, selection delimiters and noise caused by unintentional over-extension into or egress from the interaction plane, while providing strict control of the device position and the ability to show and hide off-screen targets.

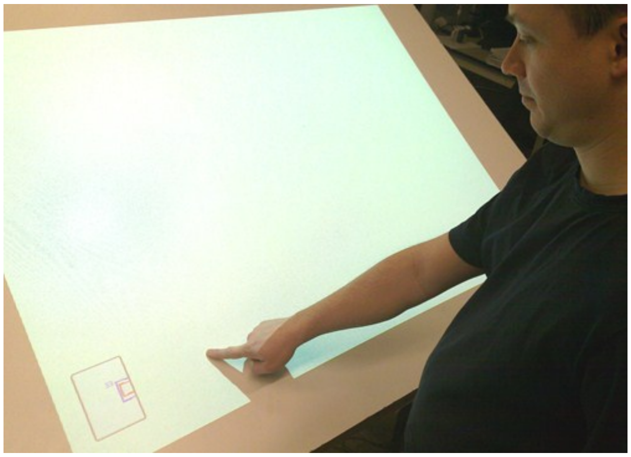

**Figure 5. Emulated device projected onto a touch surface.**

### Task

The emulated mobile device appeared in the lower left corner of the interactive display such that interactions could occur either above or to the right of the device. These regions are within reach (~60cm) for right-handed use [21]. In each trial, a circular target (4cm diameter) was placed at one of several predetermined target distances and angles. We asked participants to point to the target as quickly and accurately as possible.

### Procedure

To begin a trial, the participant presses and holds a 'start' button. After a random delay of 500-1000ms, introduced to prevent rote selection, either a 4cm target or its static visualization cue becomes visible. The trial time begins when the participants drag (in glide) or lift (in launch-adjust) their finger from the button. For the launch-adjust interaction, a transition from the former to the latter phase occurs when contact is made with the touch surface. Once contact is made (or when the glide action leaves the start button), a dynamic cue indicates the finger location. Targets and static cues are blue and are highlighted in green when the pointing finger is aligned with the target. Dynamic cues are shown in red (Figure 3).

The trial ends successfully when the user finds and selects the target. An unsuccessful trial results if the participant makes a selection outside of the target or if the launch motion lands beyond a predefined boundary (3 SD of launch error across all conditions determined by our pilot test, a total of 18cm from the target center). At the end of each trial an auditory cue is played to indicate either success or failure. Failed trials are re-queued unless a condition was failed five times within a block. Before each trial the frame of the emulated mobile device is randomly offset by 4 to 8cm to prevent memorization of target positions based on external kinesthetic or visual references.

### Design

The study uses a 3×6×2 within-subjects design:

- Visualization, *Viz*: Wedge, overview+detail (O+D) and Visible Target (VT);
- Target distance, *A:* 12, 19, 27, 37, 48 and 60cm;
- Target angle, $\theta$: 0° and 45°;

To counter learning and fatigue effects, the visualization conditions are spread across seven blocks, each containing a balanced rotation of visualizations (e.g. Block 1: VT, O+D, Wedge; Block 2: O+D, VT, Wedge; …). Within each *Viz,* all combinations of $A$ and $\theta$ are presented in random order. The orders of the interaction methods and visualization blocks are balanced between participants. The complete set of trials is completed by each participant for both the launch-adjust and glide interaction methods, in balanced order.

Because screen corners are known to be problematic for some off-screen visualization cues [12], we place targets along a line that makes either a 0° or 45° angle from the x-axis defined by the center of the emulated display.

We did not include target width as a factor, and instead chose a single target size of 4cm. Results of a pilot study with a crossing task showed target width to have little or no effect on performance. Because visualization cues do not encode target width, any effect that does exist cannot be interpreted in the sense typically associated with Fitts' law. Instead we directed our focus to effects of visual cue and target angle over distance.

### Participants

We recruited eighteen paid volunteers for the study, twelve male and six female. None were familiar with the concept of direct off-screen pointing nor had participated in the pilot. All participants completed tasks with their right hand.

Two participants, one ambidextrous and one left-handed, both completed the tasks without difficulty using only their right hand.

## Results

We analyzed each interaction method independently. We used a 3×6×2 repeated measures ANOVA and Bonferroni adjustment for post-hoc comparisons.

### *Time Analyses*

Our time data analyses included only successful trials, with the first block removed to reduce learning effects. We discarded a further 189 trials (2% of the total 9072 successful trials) that were more than 3 SD from the mean for any combination of $A$, $\theta$ and *Viz*.

### *Effect of Off-Screen Target Cues*

Off-screen target cues add a considerable overhead to pointing times for both the glide and launch-adjust interactions, with trials roughly three to four times as long as VT (Figure 6a). We are surprised to find times for the adjustment phase of the launch-adjust interaction (Figure 6b) approaching those of the glide trials, despite the comparatively short distance of the adjust motion. The adjustment times are far greater than the launch times, consuming almost three quarters of the entire launch-adjust movement time. We can attribute the slower motions for Wedge and O+D mainly to the cognitive load of interpreting the visual cues, which occurs concurrently with pointing motion. VT is significantly faster than O+D ($p<0.001$) and Wedge ($p<0.0001$) in the launch phase of the launch-adjust, accounting for main effects of *Viz* ($F_{2,34}$=26.31, $p<0.0001$) on launch time. Trials with visual cues are slower despite our encouragement of participants to study the visual cue and picture an imaginary target before beginning the pointing motion. Thus, although the difference is less pronounced in the launch, the cognitive overhead is not entirely eliminated.

Overall, O+D appears to allow faster target selection than Wedge, however the difference is significant only in the case of the glide ($p<0.01$) motion (main effect of *Viz* on glide: $F_{2,34}$=146.30, $p<0.0001$). The difference nonetheless confirms our assumption that not all visualization cues perform equally. The inequality of O+D and Wedge is further highlighted by a crossover in trial times, visible in Figures 7 and 8. Wedge is faster for closer targets, but is overtaken by O+D as $A$ increases.

### *Effect of Target Angle*

Effects of $\theta$ in the glide and launch-adjust interactions are visible in figures 7 and 8. We see longer movement times for distant targets at 45°, particularly for Wedge. Interactions between $\theta$ and *Viz* are significant for both glide ($F_{2,34}$=11.04, $p<0.001$) and launch-adjust ($F_{2,34}$=5.45, $p<0.01$) but not for sub-components of the launch-adjust. $\theta$ × $A$ interactions are significant in all cases (glide: $F_{5,85}$=15.74, $p<0.0001$; launch-adjust: $F_{5,85}$=13.22, $p<0.0001$; launch phase: $F_{5,85}$=5.98, $p<0.0001$; adjust phase: $F_{5,85}$=12.69, $p<0.0001$) and can be seen by the crossover between the 0° and 45° conditions in both figures.

The effects of target angle indicate differences in the comprehensibility of visualization cues between corner and edge conditions. Human factors, however, including kinesthetic and perceptual constraints, also contribute to performance differences between target angles.

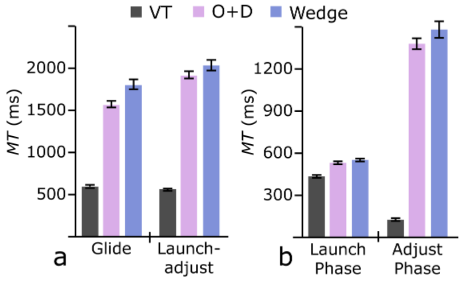

**Figure 6. a) Mean trial time across A and θ for the glide and launch-adjust interaction. b) Mean time across A and θ for sub-components of the launch-adjust (Error bars: ±2 SE).**

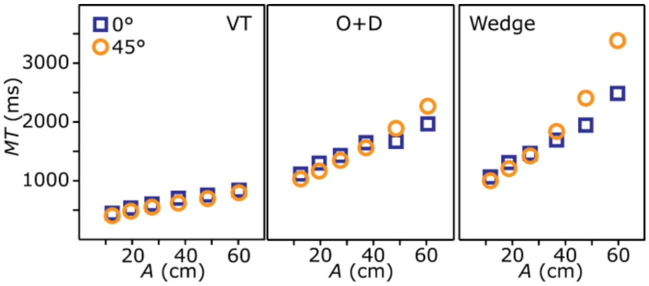

**Figure 7. Mean glide movement time across distances for each target angle.**

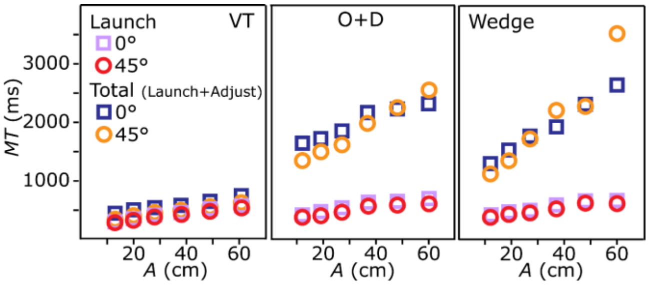

**Figure 8. Mean launch and total movement time (launch phase + adjust phase) across distance for each target angle.**

### *Launch Error Analyses*

For analyses of pointing error in the launch motion, we combined completed and uncompleted launch trials, including those out of the minimum tolerance range.

The mean launch error for each visualization condition is shown in Figure 9b (next page). While the small mean launch error for VT is unsurprising, we see a much larger mean error for both visualization cue types, a clear result of a participant's reliance on interpretation of the static off-screen target cue. Participants tended to undershoot (38% of trials for O+D and 49% for Wedge), despite our requests for participants to predict the target location as accurately as possible.

A speed-accuracy tradeoff between O+D and Wedge is apparent (Figure 9b). Whereas O+D is faster, we see significantly higher (p<0.01) overall accuracy with Wedge (main effect of *Viz* on launch error: $F_{2,34}$=285.62, p<0.0001). On closer examination (Figure 10, next page), it appears that Wedge is particularly accurate at close range but degrades with *A* much more rapidly than O+D does.

We used principal components analysis (PCA) [23] to find a 95% [13] confidence ellipse surrounding the spread of launch landing points produced in the launch phase. PCA allows us to determine the two-dimensional vector that contains the greatest variance in the distribution of launch points. The eigenvalues and eigenvectors of the covariance matrix respectively correspond to the dimensions and orientations of our variance vectors.

The resulting ellipses (Figure 9a indicate the target size required to achieve a 95% accuracy rate for direct off-screen pointing without visual cue assistance. Examination reveals further implications for designers. First, the pronounced tendency for participants to undershoot the target is clearly visible. Also, there are apparent differences across cue types. The Wedge produces shorter and much narrower ellipses, reflecting differences in information scent between cue types. We can also see apparent differences in accuracy between the 0° and 45° target angles. This difference is much less pronounced for the O+D, which is expected, given the Wedge's inherent weakness in corners.

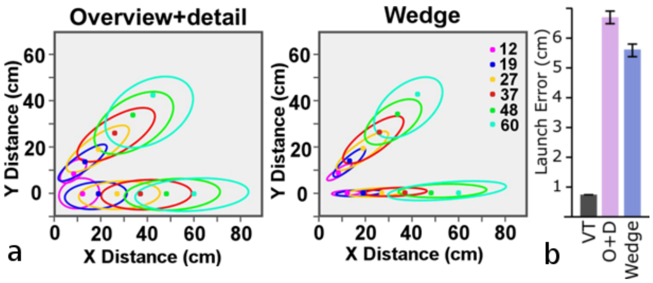

**Figure 9. a) Principal component analysis reveals potential target sizes along different angles. Ellipses are narrower for the Wedge than for the overview+detail (best viewed in color). b) Launch error across A and θ (Error bars: ±2 SE).**

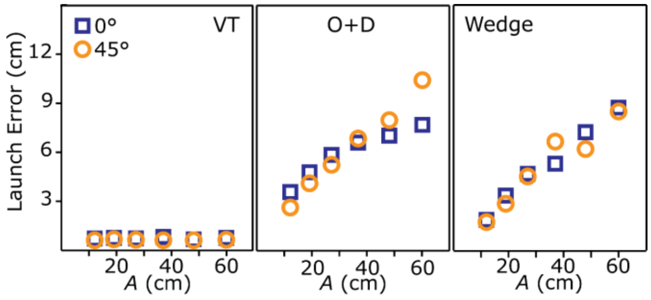

**Figure 10. Mean launch error across distance for each target angle.**

## Model Verification

### *Glide Interaction*

We tested data from trials using the glide interaction against both Fitts' and steering (Eq. 1 and 2) models using linear regression of *MT* over task *ID*. Results are summarized in Table 1.

As expected, the visible target baseline condition fits very well to Fitts' law. Both assisted pointing conditions fit either model, however, we find the negative *y*-intercept (constant *a*) value produced by fitting Wedge to Fitts' model to be counter-intuitive.

We use Hotelling's t-test to compare the fit for the Fitts' and steering models. The results indicate that the steering model is a significantly better fit (p < 0.05) for both O+D (t(3) = 2.518) and Wedge (t(3) = 3.299).

| Model | Visualization | *a* (ms) |  | $R^2$ |
|---|---|---|---|---|
|  |  | Est. | Std. Err. |  |
| Fitts' Law | VT | 12 | 26 | 0.992 |
|  | O+D | 51 | 208.1 | 0.933 |
|  | Wedge | -911 | 456.6 | 0.902 |
| Steering Law | VT | 329 | 17.2 | 0.986 |
|  | O+D | 860 | 43.0 | 0.988 |
|  | Wedge | 519 | 104.2 | 0.979 |

**Table 1. Summary of model fitting for the glide interaction.**

### *Launch-Adjust Interaction*

For the launch-adjust interaction, we verify its components $T_L$ and $T_C$ (Eq. 3), then the model as a whole. First, we fit the launch time $T_L$ against the *ID* from Fitts' law (Eq. 4). A surprisingly good fit ($R^2 \geq 0.964$) results from all conditions (Table 2), regardless of target visibility, confirming Fitts' law as a predictor for unseen targets.

| Model | Visualization | *a* (ms) |  | $R^2$ |
|---|---|---|---|---|
|  |  | Estimate | SE |  |
| Fitts' Law | VT | -25 | 34.1 | 0.979 |
|  | O+D | 87 | 34.3 | 0.978 |
|  | Wedge | 101 | 44.8 | 0.964 |

**Table 2. Summary of model fitting for the launch segment of the launch-adjust interaction.**

$T_C$ in the baseline case (VT) is simply the time it takes for participants to remove their finger from the target. However, with target cues, the average adjust time is dependent on the function *g*(*A*), which describes the amount of 'pointing error' in the launch phase. The logarithmic function in Eq. 12 provides a good estimate of this error function ($R^2 \geq 0.936$ for O+D and Wedge).

Table 3 (next page) shows a summary of how the *adjust* time $T_C$ fits a linear regression test against the Fitts' and steering models (Eq. 8 and 9) with a substitution of eq. 12 for *g*(*A*). The *ID* is estimated using the parameters found by the least-squares fit of the logarithmic error function. As with the glide interaction, all cases produce a good fit ($R^2 \geq$ 0.873). Hotelling's t-test shows a significant difference between the models for Wedge (t(3) = 9.619, p < 0.01) and approaches significance for O+D (t(3) = 3.110, p = 0.053).

| Model | Visualization | *a* (ms) | | $R^2$ |
|---|---|---|---|---|
| | | Estimate | SE | |
| Fitts' Law | O+D | 269 | 180.0 | 0.908 |
| | Wedge | -252 | 341.4 | 0.873 |
| Steering Law | O+D | 584 | 103.1 | 0.941 |
| | Wedge | 144 | 210.2 | 0.919 |

**Table 3. Summary of model fitting for the adjust stage of the launch-adjust interaction.**

Based on the above analyses, we choose the steering law (Eq. 9) for the adjust phase in our validation of the complete launch-adjust model. Since our study used a single target width of 4cm, we can reduce the model to a function with three constants:

$$MT = a + b\log_2(A + 4cm) + c\log_2(A) \qquad (13)$$

Eq. 13 provides a very good fit for both visualizations (O+D: $R^2$ = 0.995, Wedge: $R^2$ = 0.980). Since no previous model exists for the launch-adjust interaction, we choose Fitts' law as a baseline for comparison (VT: $R^2$ = 0.976, O+D: $R^2$ = 0.964, Wedge: $R^2$ = 0.942). Because we are comparing two models with different numbers of parameters, we can use the Akaike information criterion to assist in our evaluation. The resulting values (O+D: AIC = -28.69 for Eq. 1, AIC = -38.54 for Eq. 13. Wedge: AIC = -25.83 for Eq. 1, AIC = -30.22 for Eq. 13) indicate that our improved model is superior (because it has the lower scores). Regardless, our two-step model provides a superior explanation than Fitts' of the mechanisms behind the launch-adjust interaction.

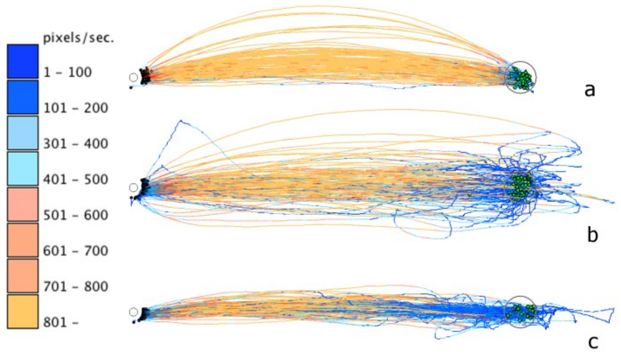

**Figure 11. Paths of all glide trials for (a) VT, (b) O+D and (c) Wedge with a target at 0° and distance of 48cm, colored according to instantaneous velocity. Motion begins on the left and finishes at the target location, right (best viewed in color).**

Examination of traversal paths for the glide motion reveals differences between visualization conditions. Response to a visible target (Figure 11a) is what we expect by Fitts' law, with an initial ballistic impulse and decreasing velocity as the target is approached. As information about the target location is reduced (Figure 11b and c), the ballistic impulse becomes shorter and less powerful, presumably resulting in a degradation towards linear movement time. Notice the path breadth for the Wedge is narrower than for the other two conditions. These findings are consistent with our regression fitting results.

## DISCUSSION

### Summary of Findings

Speed-accuracy tradeoffs inherent to pointing tasks are heightened in the off-screen setting. Fast performance, comparable to a visual task, is possible with static cues. Degrading accuracy may limit granularity to as few as three targets in depth from the screen edge. Conversely, dynamic cues allow high precision target acquisition, but with average times up to three or four times as long.

Our results show trade-offs in different aspects of information scent for off-screen visualization cues. The annotated Wedge used in our study shows a disadvantage in the resulting movement time for off-screen pointing, likely because of the superior spatial information portrayed by the overview+detail cue. A notable exception to the rule occurs at close range; Wedge results in faster pointing times with better accuracy. In general, pointing errors with Wedge are considerably lower than with overview+detail. This is particularly true regarding directional accuracy.

In modeling the movement time, we have produced satisfactory explanations for the glide and launch-adjust interaction styles. The results show interesting patterns. The Steering law can be viewed as a predictor of the upper bound of movement time for assisted direct off-screen pointing. Cues with better scent can improve on this upper limit.

We define a two-step interaction with distinct segments for the launch-adjust interaction. The launch is clearly a good fit to Fitts' law, despite reduced visual target feedback. The adjust motion can be modeled as a miniature glide, in which the parameters are defined by the accuracy of the launch.

Some interesting effects consistently occur as a result of target angle across interaction styles. Differences due to problematic corner use are not unexpected for Wedge. More intriguing are the similar but lesser effects for the overview+detail and the drastically different accuracy ellipses for the side and corner conditions that result with both cue types.

### Guidelines to Designers

Our results offer the following guidelines to designers:

- coarse-grained selection (only a few items per axis) is possible with only static cues;
- dynamic cues should be used for fine-grained selection and are effective for targets as small as 4cm;
- cues for off-screen target should be carefully selected as their scent capacity can vary with distance; a cue well-suited for shorter distances could impair performance at larger distances;
- weigh the costs of fine vs. coarse-grained selection; the latter can significantly improve performance times;

- take advantage of directional precision; better accuracy is achievable along the axis of directional cues.

## Applications

We describe a few applications that can benefit from our results and general guidelines.

### *Coarse-grained storage bins*

Binning content in off-screen space [27] requires storing a small number of items and has few constraints on targeting accuracy. For example, to assist with multitasking on a mobile device, instead of pinning applications on the screen (e.g. Microsoft Zune), application windows can be placed in 2D bins around the viewport (Figure 12). With a direct pointing gesture into off-screen space the user brings a binned application into view. Based on our results (a) larger sized bins can be targeted without resorting to dynamic cues; (b) off-screen space on the sides of the device would be used before space in the corners; (c) space closest to edges should be used before space further away; and (d) bin widths should grow logarithmically with distances away from the device's edge.

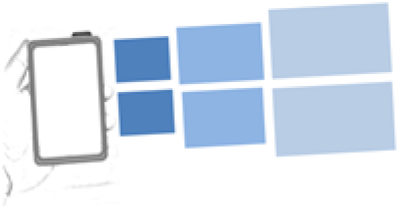

**Figure 12. Off-screen storage bins. The user can save application windows in these positions and later recall them.**

### *Fine-grained direct navigation*

Finer grained off-screen selection is also possible, according to our results. However, these can be more time-consuming as they necessitate dynamic cues for tracking the finger. For example, techniques such as space-filling thumbnails [8] cannot be easily adapted to small viewport sizes. Instead, systems such as Flip Zooming [3] that break up a document into smaller segments have been proposed. When a page is in focus it is displayed at its normal, readable size; but otherwise it is reduced in size and displayed as a thumbnail. With direct off-screen pointing, the document can be partitioned into thumbnails that reside in the off-screen workspace. As the device tracks the user's finger's motion among the thumbnails, it displays an on-screen preview. With a lift-off action the user can bring the current selection into focus or, conversely, cancel their action with a gesture (e.g. circular motion).

### *Continuous direct off-screen exploration*

Whereas fine- and coarse-grained selections (minimally) require dynamic and static cues, respectively, certain applications do not necessitate either. For example, when the system cannot discretize the workspace (e.g. a map), direct exploration is still possible by gliding the finger in off-screen regions. A small on-screen window could display the context surrounding the finger, and with a gesture the user can 'call-in' that part of the workspace. Unlike flicking which continuously refreshes the user's viewport, this form of interaction would facilitate an extra 'peep' state before calling content onto the screen.

## Limitations and Future Work

*Experimental platform.* While our experimental system characterizes direct off-screen pointing performance, we need to verify its robustness in a more realistic mobile setting. In the short-term we will design prototypes that sense distances up to 60cm around the device and use these to verify our results. Additional work is required to verify and adapt our models to pointing in free air.

*Targeting regions.* We verified our hypotheses in two general regions (right side and top-right corner) around a mobile device. Additional experiments are needed to derive the limits of off-screen pointing in less accessible regions. For example, right-handed users have limited reach and can occlude the screen by targeting the left side.

*Target width.* With the exception of overview+detail, current off-screen target visualizations do not encode target size. This partly constrained our evaluation to only one width. In the future we will verify our base models with representations that encode varying degrees of target size.

*Scalability.* This first investigation restricted pointing at one target only. In most applications, cues for multiple off-screen targets will clutter the screen. While visualizations such as Wedge can minimize the effect of clutter, additional studies are needed to evaluate the scalability of our results.
Conclusion

Direct off-screen pointing extends interactions on a device to input around the device's edges. This form of input is becoming accessible with novel sensing technologies. In this paper we present a study to characterize the performance of direct off-screen pointing, when assisted with target cues. We show that pointing time and accuracy are dependent on the type of feedback given to users during this task. We present two styles of interaction, glide and launch-adjust. Our results show that (a) regardless of interaction style, the type of dynamic off-screen target cues dictate pointing performance; (b) the correction of pointing error can take as much as three quarters of the overall selection time, thus improving this phase can lead to significant performance benefits; (c) sophisticated cues can be highly accurate but with up to three times the cost in throughput; (d) our derived model provides a good explanation and prediction for pointing performance; (e) performance changes with target position around the device. These results can guide the design of future applications for direct off-screen pointing.

## ACKNOWLEDGMENTS

We thank NSERC for financially supporting this project, our reviewers for invaluable feedback, and our participants.

## REFERENCES

1. Accot, J. and Zhai, S. Beyond Fitts' law: models for trajectory-based HCI tasks. In *Proc. CHI 1997*, ACM Press (1997), 295-302.
2. Baudisch, P. and Rosenholtz, R. Halo: a technique for visualizing off-screen objects. In *Proc. CHI 2003*, ACM Press (2009), 481-488.
3. Björk, S. Hierarchical flip zooming: enabling parallel exploration of hierarchical visualizations. In *Proc. AVI 2000*, ACM Press (2000), 232-237.
4. Burigat, S., Chittaro, L., and Gabrielli S. Visualizing locations of off-screen objects on mobile devices: A comparative evaluation of three approaches. In *Proc. MobileHCI 2006*, ACM Press (2006), 239-246.
5. Butler, A., Izadi, S., and Hodges, S. SideSight: multi-"touch" interaction around small devices. In *Proc. UIST 2008*, ACM Press (2008), 201-204.
6. Cao, X., Li, J. J., and Balakrishnan, R. Peephole pointing: modeling acquisition of dynamically revealed targets. In *Proc. CHI 2008*, ACM Press (2008), 1699-1708.
7. Carlton, L. Visual information: The control of aiming movements. *Quar. J. Exp. Psy. 33*, 1 (1981), 87-93.
8. Cockburn, A., Gutwin, C., and Alexander, J. Faster document navigation with space-filling thumb-nails. In *Proc. CHI 2006*, ACM Press (2006), 1-10.
9. Fitts, P.M. The information capacity of the human motor system in controlling the amplitude of movement. *J. Exp. Psychology* (1954) 47, 381-391.
10. Gillan, D. J., Holden, K., Adam, S., Rudisill, M., and Magee, L. How does Fitts' law fit pointing and dragging? In *Proc. CHI 1990*, ACM Press (1990), 227-234.
11. Gustafson, S. Visualizing off-screen locations on small mobile displays. Master's Thesis, Department of Computer Science, University of Manitoba (2008).
12. Gustafson, S., Baudisch, P., Gutwin, C., and Irani, P. Wedge: clutter-free visualization of off-screen locations. In *Proc. CHI 2008*, ACM Press (2008), 787-796.
13. Gustafson, S., Bierwirth, D., Baudisch, P. Imaginary interfaces: Spatial interaction with empty hands and without visual feedback. In *Proc. UIST 2010*, ACM Press (2010), 3-12.
14. Gustafson, S. and Irani, P. Comparing visualizations for tracking off-screen moving targets. In *Proc. CHI 2007,* ACM Press (2007), 2399–2404.
15. Harrison, C. and Hudson, S. E. Abracadabra: wireless, high-precision, and unpowered finger input for very small mobile devices. In *Proc. UIST 2009*, ACM Press (2009), 121-124.
16. Hinckley, K., Cutrell, E., Bathiche, S., and Muss, T. Quantitative analysis of scrolling techniques. In *Proc. CHI 2002*, ACM Press (2002), 65-72.
17. Irani, P., Gutwin, C., Partridge, G., and Nezhadasl, M. Techniques for interacting with off-screen content. In *Proc. INTERACT 2007*, Springer (2007), 234-249.
18. Kratz, S. and Rohs, M. Hoverflow: exploring around-device interaction with IR distance sensors. In *Proc. MobileHCI 2009*, ACM Press (2009), 1-4.
19. Li, F. C., Dearman, D., and Truong, K. N. Virtual shelves: interactions with orientation aware devices. In *Proc. UIST 2009*, ACM Press (2009), 125-128.
20. Mehra, S., Werkhoven, P., and Worring, M. Navigating on handheld displays: Dynamic versus peephole navigation. *TOCHI 2006*, ACM Press (2006), 448-457.
21. NASA-STD-3000 Man-Systems Integration Standards. Rev B, 1995.
22. Nekrasovski, D., Bodnar, A., McGrenere, J., Guimbretière, F., and Munzner, T. An evaluation of pan & zoom and rubber sheet navigation with and without an overview. In *Proc. CHI 2006*, ACM Press (2006), 11-20.
23. Owen, J. and Chmielewski, M. On canonical variates analysis and the construction of confidence ellipses in systematic studies. *Systematic Zoology 34*, 3 (1985), 366-374
24. Pirolli, P. Computational models of information scent-following in a very large browsable text collection. In *Proc. CHI 1997*, ACM Press (1997), 3-10.
25. Rohs, M., Schöning, J., Raubal, M., Essl, G., and Krüger, A. Map navigation with mobile devices: virtual versus physical movement with and without visual context. In *Proc. ICMI 2007*, ACM Press (2007), 146-153.
26. Tan, D. S., Pausch, R., Stefanucci, J. K., and Proffitt, D. R. (2002). Kinesthetic cues aid spatial memory. *Ext. Abstracts CHI 2002*, ACM Press (2002), 806-807.
27. Wang, Q., Hsieh, T., and Paepcke, A. Piles across space: Breaking the real-estate barrier on small-display devices. *Int. J. Hum.-Comput. Stud. 67*, 4 (2009), 349-365.
28. Ware, C. and Lewis, M. The DragMag image magnifier. In *Proc. CHI 1995*, ACM Press (1995), 407-408.
29. Yee, K. Peephole Displays: Pen interaction on spatially aware handheld computers. In *Proc. CHI 2003*, ACM Press (2003), 1-8.
30. Zellweger, P. T., Mackinlay, J. D., Good, L., Stefik, M. and Baudisch, P. City lights: contextual views in minimal space. *Ext. Abstracts CHI 2003*, ACM Press (2003), 838-839.