Pinch-to-Zoom-Plus: An Enhanced Pinch-to-Zoom That Reduces Clutching and Panning

Jeff Avery, Mark Choi, Daniel Vogel, Edward Lank
Cheriton School of Computer Science
University of Waterloo
Waterloo, ON, Canada N2L 3G1
{j2avery, hichoi, dvogel, lank}@uwaterloo.ca

ABSTRACT
Despite its popularity, the classic pinch-to-zoom gesture used in modern multi-touch interfaces has drawbacks: specifically, the need to support an extended range of scales and the need to keep content within the view window on the display can result in the need to clutch and pan. In two formative studies of unimanual and bimanual pinch-to-zoom, we found patterns: zooming actions follows a predictable ballistic velocity curve, and users tend to pan the point-of-interest towards the center of the screen. We apply these results to design an enhanced zooming technique called Pinch-to-Zoom-Plus (PZP) that reduces clutching and panning operations compared to standard pinch-to-zoom behaviour.

Author Keywords
mobile; multi-touch; interaction.

ACM Classification Keywords
H5.2 Information interfaces and presentation: User Interfaces - Input devices and strategies.

INTRODUCTION
Increasing or decreasing the document zoom level (i.e. scale or magnification) according to the distance between two simultaneous touch points [9,18] is a standard gesture on virtually all multi-touch devices. This "pinch-to-zoom" interaction technique is direct, simple, and intuitive, but not perfect. To reach distant zoom levels, frequent clutching is needed (where both fingers are lifted to reset the distance between touch points to continue zooming). Specifying the optimum zoom origin is difficult and often requires panning to keep the location of interest from moving off the screen. These problems are exacerbated by extreme focus+context tasks which require frequent zooming in-and-out between a close-up view to perform an action, and an overview of the entire document [12,13,15]. Past work has proposed replacing pinch-to-zoom with new multi-touch zoom techniques to eliminate clutching [6,7,14] and switch between an overview and close-up [4]. As of yet, none of these techniques have reduced pinch-to-zoom's dominance.

Rather than replacing pinch-to-zoom, we designed Pinch-to-Zoom-Plus (PZP), and enhancement of the standard pinch-to-zoom technique that reduces clutching and panning. It is fully compatible with current pinch-to-zoom and it works regardless of device size, handedness, or whether under bimanual or unimanual control.

The design parameters in PZP are based on results of two formative studies investigating kinematic and clutching characteristics of standard pinch-to-zoom. We found the velocity of pinch and spread behaviours follows a ballistic pattern, and used this to justify and parameterize pan and zoom acceleration to reduce clutching. We also observed a tendency to pan targets towards the center of the screen, and used this to justify automatic pan-to-centre while zooming. Our results complement recent work on pinch-to-zoom performance and ergonomics [10,17] and provides new insights and practical applications for results.

Our contributions are twofold: a detailed analysis of clutching and panning behaviour with current pinch-to-zoom technique; and the design and evaluation of PZP, an enhanced pinch-to-zoom technique shaped by that analysis. Our PZP technique reduces clutching and panning operations compared to standard pinch-to-zoom and can be implemented on all current multi-touch devices.

RELATED WORK
Buxton [2] traces the history of pinch-to-zoom to the early 1980s based on a demonstration of two-finger pinch and pan interaction in Wellner's Digital Desk video [19]. Hinckley et al. also describe the Pinch gesture [9] which allows users to zoom and pan around the center of two contact points. Pinch-to-zoom has since become the standard zooming gesture on multi-touch devices.

Despite its popularity, there are obvious problems with pinch-to-zoom for a certain subset of interactions: precision-pointing and occlusion problems are well known [16], but there are also specific issues with focus+context tasks [12]. While the pinch gesture allows users to zoom

UIST 2014, October 5–8, 2014, Honolulu, HI, USA.

http://dx.doi.org/10.1145/2642918.2647352

in to see detail, users also need to zoom back out to see high-level context. Interacting at multiple zoom scales requires frequent interaction to continually zoom the document in and out [12,13,15]. These operations often require frequent clutching, which can be slow. It can also be difficult to accurately specify the zoom origin due to finger occlusion. As a result, the location of interest can move unpredictably while zooming, even off the screen forcing users to pan to correct the position. When frequently zooming in and out during focus+context tasks, these problems are exacerbated.

To address these shortcomings, techniques other than pinch-to-zoom have been proposed. Albinsson and Zhai's Zoom-Pointing [1] is a bimanual technique in which the user draws a bounding box to define a persistent zoom area. Zoom-Pointing thus allows a user to specifically delineate the content they wish to see on-screen. This overcomes the need to pan after zooming to accurately position zoomed content. Forlines and Shen's DTLens [6] is similar, but adds controls for minimizing, closing or annotating the enlarged viewport. It also allows users to save and restore zoom levels. However, both techniques require an explicit zoom-out to return to overall context.

Other techniques have a transient zoom mode. Käser et al.'s Fingerglass [12] allows users to specify a zoom-target using a non-dominant hand, then interact with the zoomed-in area using their dominant hand. When the non-dominant hand is released, the view returns to the default zoom level. Fingerglass does not support clutching, so only a limited range of zoom scales is possible. Negulescu et al.'s two handed Offset technique moves the zoom origin out from under one finger into the user's view [15] and is transient by snapping back to the default zoom level when both fingers are lifted. Like Fingerglass, clutching is not supported limiting the range of zoom scales.

These techniques all attempt to address excessive clutching and panning by replacing pinch-to-zoom. However, pinch-to-zoom is already ubiquitous, and a technique that enhances pinch-to-zoom rather than replacing it, may be more approachable. For that reason, researchers have begun looking more closely at pinch-to-zoom. Hoggan et al. [9] examine the ergonomics of pinch-to-zoom, identifying factors that contribute to performance, such as direction, distance, angle and position. Tran et al. [15] build a quantitative model of pinch-to-zoom performance. We extend this work with a complementary analysis of pinch-to-zoom with focus+context tasks. However, we go beyond examining current behaviour, and our results to motivate and provide design parameters for an improved pinch-to-zoom technique.

ANALYZING PINCH-TO-ZOOM

We conducted two formative studies to study pinch-to-zoom scale adjustment with and without panning to alter translation. This is based on pilot studies, where we observed users performing both sequential and separate zoom and pan actions and simultaneous zoom and pan actions. The first study centered all targets requiring zoom actions, but not panning, so that we could examine zoom clutching in isolation. The second study placed targets around a perimeter to encourage panning along with zooming. The pilot studies also revealed people perform pinch-to-zoom on a tablet with one or two hands: using one finger from each hand (bimanual) or two fingers from one hand (unimanual). We control for these two variants in both studies.

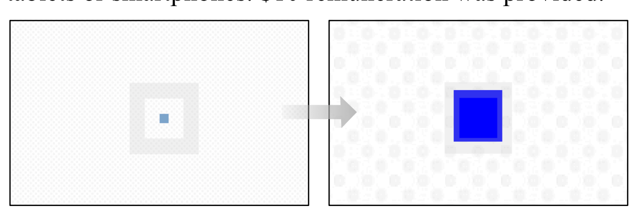
Apparatus

Both studies were performed on a Toshiba AT200 1.2 GHz Dual Core tablet with a 1280 by 800 pixel display (218 by 136 mm, 5.88 px-per-mm). Custom software was implemented in Android 4.03, and tuned to duplicate standard Android pinch-to-zoom behaviour. Observed lag was minimal, and events were logged at a device sample rate of 60Hz. The tablet was placed flat on a standard desk in landscape mode, and anchored to the surface using rubber pads to prevent movement. An overhead video camera recorded the session.

Study 1: Only Zoom Required

The goal of the first study was to examine clutching in pinch-to-zoom gestures using a simple docking task with the target and dock centered on the display (see Figure 1). The target was a blue opaque square and the docking area was a semi-transparent blue rectangle with a border thickness representing the fit tolerance. The background was textured to provide context and simulate a document task, like map navigation. At the start of the task, the dock was placed mid-screen with the target centered over it. Panning could be used to move the target, but the dock did not move from the center. Thus, panning was allowed, but was not required to complete the task. The dock highlighted when the target fit correctly, and the task was completed when both fingers were released in that state. Note that all trials were ultimately successful, but intermediate errors caused by tighter fit tolerance increased the difficulty of the task.

Twenty right-handed participants (3 female), with an average age of 27 (SD 5.44) participated in our first study. All participants had prior experience with multi-touch tablets or smartphones. $10 remuneration was provided.



(i) (ii)

Figure 1: Study 1 docking task. Participants were asked to zoom the target (i) to make it fit within the docking area (ii). Targets turn bright blue when properly scaled to the docking area, indicating that the operation was successful.

The design is within-subjects with independent variables for hand condition (unimanual/bimanual), tolerance of docking region (4 mm, 7 mm, 10 mm tolerances), zoom direction (zoom-in, zoom-out), and zoom level (4, 7, 10, 13, 16, 19x for zoom-in, and multiplicative inverse levels for zoom-out). The zoom-out direction required participants to scale a large target down, and zoom-in required the opposite. The six zoom levels were chosen to cover typical pinch-to-zoom scenarios; for example, the maximum zoom level of 19x is roughly equivalent to zooming from a map of a large city (10 km per cm) down to a city block (525 m per cm). Hand condition (unimanual/bimanual) was counterbalanced between participants, and all tolerances, zoom levels, and zoom directions were fully crossed and presented in a random ordered block of trials, to balance potential learning effects. Five blocks were presented creating five repetitions.

An initial practice block of 12 trials presented a subset of tolerances {7 mm, 10 mm} and zoom levels {4, 7, 15, 0.25, 0.14, 0.06}. Not counting these practice trials, the experiment had 360 trials per participant. The experiment time was approximately 45 minutes.

Pre-Processing

We define an *action* as the sequence of two finger dragging movements occurring between touch down and touch up events. Note that a clutch occurs between actions. We classify each action as either a *zoom* or *pan* using thresholds determined in our pilot study:

- *Pan actions* are defined as actions where fingers spread less than 5 px (0.85 mm). Pans of less than 2 px (0.34 mm) were discarded as unintentional movements.
- *Zoom actions* are defined as actions where fingers spread more than 5 px (0.85 mm). We differentiate between zoom-in and out based on spread direction.

For each trial, we calculated the completion time and number of zoom-in, zoom-out, and pan actions. One participant was an outlier with all completion times more than two standard deviations from the overall mean, and was excluded from analysis. Learning effects of block on elapsed time were observed ($F_{2,18}$ = 7.79, p < .01), so block 1 was excluded. Excluding practice blocks and block 1 tasks, the remaining 19 participants performed a total of 4104 tasks.

Results

Mean task completion time was 5.65 s (SD 2.71). Handedness had no significant effect on elapsed time ($F_{1,18}$ = 0.418, ns). As expected, tolerance had an effect on the elapsed time, with smaller targets taking significantly longer to complete ($F_{2,18}$ = 210.85, p < .01). Zoom level also had a significant impact on time ($F_{2,18}$ = 32.83, p < .01), with the completion time increasing as greater zoom levels were required to achieve the target.

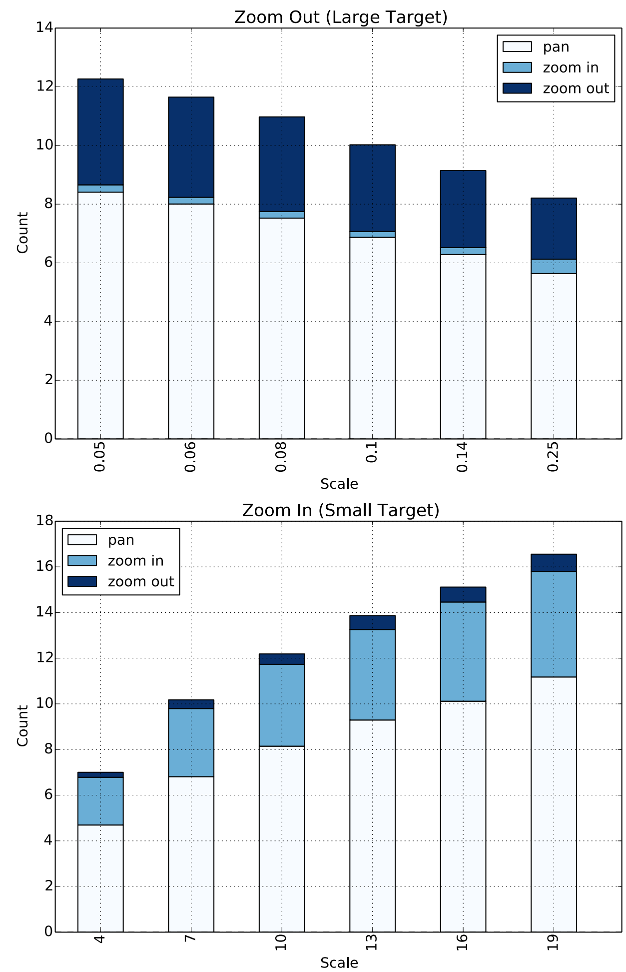


Figure 2: Count of zoom and pan actions by zoom level. More than 1 zoom or pan action requires a clutch (i.e. 1 action has 0 clutches, 2 actions require 1 clutch, etc.)

Mean counts of actions per task were 1.94 zoom-in actions (SD 2.33), 1.75 zoom-out actions (SD 1.86) and 7.76 pan actions (SD 4.66). Zoom-in count was not significantly affected by handedness ($F_{2,18}$ = 2.096, ns), but was affected by tolerance ($F_{2,18}$ = 41.94, p < 0.01) and scale ($F_{2,18}$ = 125.66, p < 0.01). Zoom-out count was not significantly affected by handedness ($F_{2,18}$ = 0.374, ns) but was affected by tolerance ($F_{2,18}$ = 199.792, ns) and zoom-out ($F_{2,18}$ = 125.148, ns). Pan count was not significantly affected by handedness ($F_{2,18}$ = 0, ns) but was affected by tolerance ($F_{2,18}$ = 89.49, p < 0.01) and scale ($F_{2,18}$ = 41.65, p < 0.01). Smaller or larger targets, and low-tolerance targets, require more zoom and pan actions to acquire.

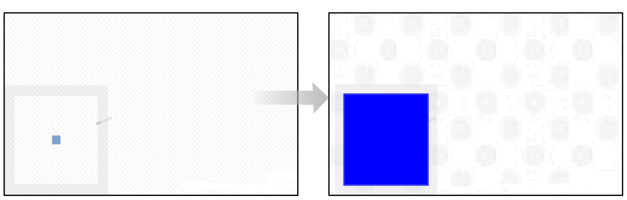
The pattern indicates that even a small change in zoom-level requires clutching: 1.94 clutches for zoom-in (4x had a mean of 2.1 zoom-in actions) and 1.75 clutches for zoom-out (0.25x had a mean of 2.1 clutches). Exceeding 10x or 0.01x zoom level required more than three clutches on average (Figure 2). The mean clutch time (between two successive zoom actions) was 180.5 ms (SD 250.7).

We were somewhat surprised by the number of pan operations. Note that target and dock were located in the center of the display and the dock was initially positioned over the target. Thus, panning was not a required operation during our study. Even if one assumes that zooming in (i.e. enlarging content) would require pan operations to ensure that content did not migrate off the screen, the similarity in frequency of pan actions when zooming out is not explained by this. Even a small change in resolution, 4x for zoom-in and 0.25 for zoom out, was associated with a significant number of pan actions: averages of 4.69 and 5.62 pan actions, respectively. The prevalence of pan actions when a target was optimally positioned in the center of the display raises the question of how and why participants chose to reposition content.

Study 2: Zoom and Pan

This goal of the second study is to expand the scope of the first study by placing the dock and target around the perimeter of the screen. In addition to the docking task from study 1, we add a drawing task that required a simple drawing to be performed after docking.

As in our first study, the docking task is to zoom a target in or out until it fits in a docking area, within a certain tolerance (Figure 3). Unlike our first study, targets are placed at one of 8 locations around the perimeter of the display. Panning caused both the target and dock to move together, so panning was not required to complete the task – successful docking only required zooming. However, based on the voluntary panning of study 1, we believed participants would use pan as part of the zoom operation and/or to optimize the placement of the target.

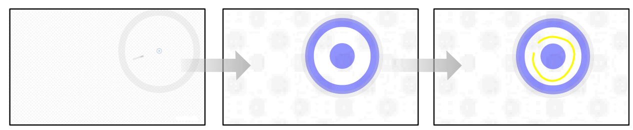


(i) (ii)

Figure 3: Study 2 docking task with targets positioned around the edge of the screen. Participants are asked to zoom the square (i) until it falls within the docking area (ii). Targets turn bright blue when properly scaled within the docking area, indicating that the operation was successful.

The second task simulates editing tasks like annotating maps or editing photographs. The presentation is the same as the docking task, except that the target and dock are circular. After reaching the required zoom level with the target fit inside the dock, participants also had to draw a circle within a confined region of the target (Figure 4). This was done to encourage participants to interact with the target, making it difficult to complete the task unless the target remained visible on the screen. For brevity, we label this second task a *drawing* task. Drawing tasks did not include a zoom-out condition, because the nature of the task required a large target in which to draw.

The screen was divided into a 3 x 3 grid of 9 regions, which were used to determine initial target position (the middle was excluded, so 8 regions were available). At the start of each task, a region was chosen randomly, and the dock and target were centered within that region. All tasks were performed with a 7 mm tolerance, the median tolerance from the first study. We used 2 zoom levels for the docking task, 13x for zoom-in, and the multiplicative inverse for zoom-out, and one zoom level for the drawing task, 13x. The decision to use a single tolerance and zoom level was motivated by the observation that any scale factor required clutch and pan actions, so fully varying across all tolerances and scales was not necessary and allowed us to limit our study to a reasonable length or time for our participants.



(i) (ii) (iii)

Figure 4: Study 2 drawing task. Participants are asked to zoom until the circular target (i) falls within the docking area (ii), and then draw a line along the inside path (iii).

We recruited 17 right-handed participants (4 female), and 3 left-handed participants (all male) with an average age of 27.7 (SD 4.98). All participants had prior experience with multi-touch tablets or smartphones. A $10 voucher was offered as remuneration.

Participants performed docking tasks first, then after a 3 min break, performed all drawing tasks. Each task was repeated for bimanual and unimanual hand conditions, with the order counterbalanced across participants. In the docking task, a practice block of 6 trials was presented at the beginning of each condition (24 trials in total). Drawing tasks included 8 trials per block, representing the 8 positions. Not counting practice trials, the experiment had 384 trials per participant (256 docking tasks plus 128 drawing tasks). The total time for the experiment was 60 minutes.

Pre-Processing

A learning effect existed across four blocks, for both docking tasks ($F_{1,19} = 18.52$, $p < .01$) and drawing tasks ($F_{1,19} = 13.75$, $p < .01$). To minimize this effect, the first block was excluded, resulting in a total of 5780 experimental tasks performed (3860 docking and 1920 drawing tasks). Actions were classified using the same tolerance as study 1 into pan actions (movement without significant zoom), zoom-in actions (scale increasing) and zoom-out actions (scale decreasing).

Results

Two-factor ANOVA and Bonferroni-Dunn correction were used for pairwise comparisons including the effects of target position, and hand condition.

Docking tasks had a mean task completion time of 4.84 s (SD 2.16). Handedness had no significant effect on time ($F_{1,19}$ = 1.005, ns). Starting target position had an effect on task time in docking tasks, both for zoom-in ($F_{1,19}$ = 4.934, $p < .01$) and zoom-out ($F_{1,19}$ = 2.752, $p < .01$), although post hoc analysis did not find which position was significant (likely due to the number of tests required).

Drawing task completion time was defined as the time required to zoom to the desired zoom-level, and excluded drawing time. Mean task completion time was 7.64 s (SD 2.10). Since users had to interact with the content after zooming in the drawing task, users appeared to spend more time positioning the target before proceeding. As with the docking task, handedness had no effect on task time $F_{1,19}$ = 4.11, ns. Starting target position had an effect on task time in drawing tasks as well ($F_{1,19}$ = 4.88, $p < .01$). As with docking tasks, post hoc analysis was unable to indicate which position was significant.

Docking tasks had a mean count of 3.5 zoom-in actions (SD 4.17), 5.67 zoom-out actions (SD 5.87), and 10.94 pan actions (SD 8.98) per task. Drawing tasks had a mean count of 7.03 zoom-in actions (SD 4.72), a mean count of 1.37 zoom-out actions, and a mean count of 14.43 pan actions (SD 9.68). These are higher than study 1.

For both docking and drawing tasks, hand condition (bimanual/unimanual) had no significant effect on zoom-in count ($F_{1,19}$ = 1.33, ns; $F_{1,19}$ = 4.16, ns) or zoom-out count ($F_{1,19}$ = 0.24, ns; $F_{1,19}$ = 4.22, ns). Handedness did have a significant effect on pan count for the docking task ($F_{1,19}$ = 67.39, $p < 0.01$), but not on the drawing task ($F_{1,19}$ = 0.19, ns). Position had no effect on zoom-in count ($F_{1,19}$ = 0.98, ns) or zoom-out count ($F_{1,19}$ = 0.97, ns) but did have an effect on pan count ($F_{1,19}$ = 22.39, $p < .01$) with targets on the left-hand side requiring more pans than targets positioned to the center or right-hand side. We observed participants panning targets towards their dominant hand when the target is further away, which is consistent with this data. For drawing, position affects zoom-in count ($F_{1,19}$ = 5.25, $p < .01$), zoom-out counts ($F_{1,19}$ = 6.00, $p < .01$), and pan counts ($F_{1,19}$ = 3.61, $p < .01$). Targets at the horizontal midpoints of the display (top-center and bottom-center) appear to require fewer zoom actions.

Synthesis and Further Analysis

Multiple clutch and pan actions were observed in both of our studies, regardless of hand condition, location, or zoom direction. In this section, we further analyze zoom and pan gestures to explore ways to reduce the number of clutch and pan actions.

Given our observation of multiple zoom-in/zoom-out and pan operations, a first question to ask is whether there exists some technique akin to cursor acceleration that could be applied to pinch-to-zoom operations to reduce clutching. Figure 5 depicts an analysis of zoom and pan speed data, aggregated across all single-handed trials (blocks 3-4 only, to reduce learning effects). Zoom and pan gestures both exhibit a ballistic pattern, with an initial high-speed movement to resize or reposition the target. For zoom actions, an initial ballistic movement peaking at 20% of the total duration, and is followed by a longer, low-speed tail beginning at about 40% of the duration. Pan movements exhibit this same ballistic profile, but without the longer, low-speed tail.

Considering screen location data for both of our studies, panning was not required, but was a high-frequency operation. This suggests that participants were deliberately repositioning content on the display. To understand why pan operations were occurring, we look at the final location of target and dock to determine why participants reposition content. At task completion, the mean distance between the target/dock and the center of the screen is 248.9 px (SD 193.7). Histogram data for distance from the center of the screen is shown in Figure 7. In a scatter plot of the final location of dock and target shown in Figure 7, we can see that target locations are grouped around the center of the display and that locations near the edges are avoided. The slight bias to the right-of-center in Figure 7 may be due to handedness of participants, but more data is needed to determine if there is any significant bias.

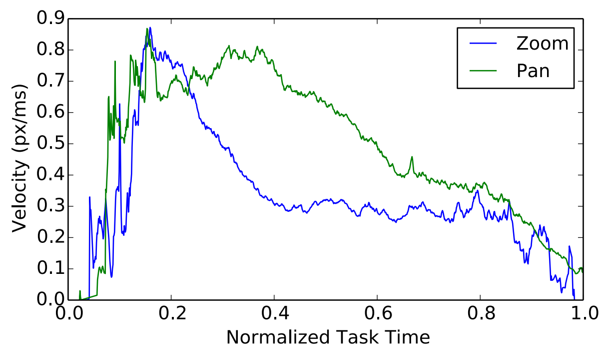


Figure 5: Mean pan and zoom velocity profiles for docking tasks with duration between 2681 ms to 7005 ms (mean time +/- 1 SD, normalized time for zoom-in, blocks 3 and 4 only).

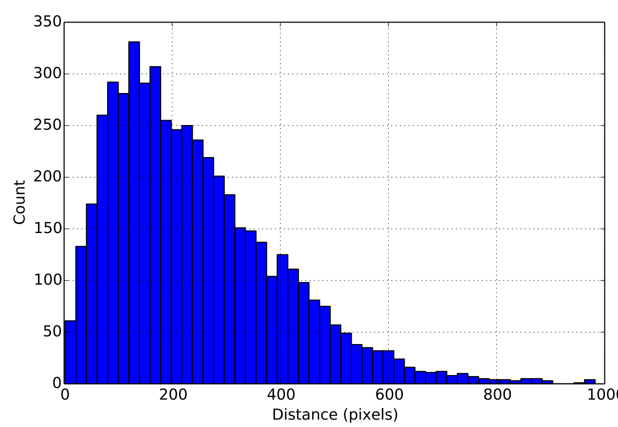


Figure 6: Distances from the final target to the center of the screen, measured in pixels.

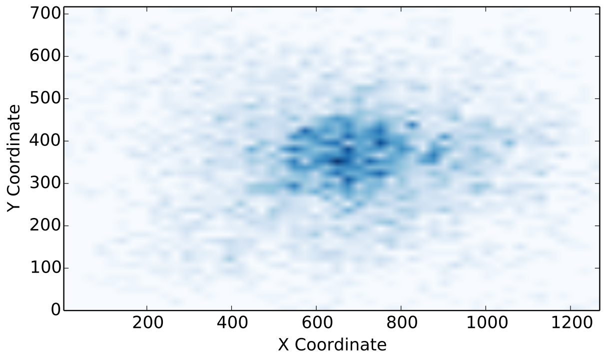


Figure 7: Heat map showing screen coordinates of completed experiment 2 docking and drawing tasks, demonstrating that users tend to pan the target to the center of the display.

We leverage these study insights regarding balistic velocity profiles when clutching, and the tendancy to pan content to the centre when zooming, to design an enhanced pinch-to-zoom technique.

PINCH-TO-ZOOM-PLUS (PZP)

PZP is designed to reduce the need to clutch and pan while zooming. To accomplish this, we incorporate two distinct augmentations to pinch-to-zoom. Zoom acceleration is a zoom transfer function that increases the relative zoom factor during rapid spreading or pinching movements. Pan-to-center is an automatic translation that moves the zoom target toward the center of the display.

Zoom Acceleration

With standard pinch-to-zoom, two contact points are positioned on the screen near the area of interest, and spread apart to zoom-in, or pinched together to zoom-out. This results in a linear zoom scale change proportional to the change in distance between the two contact points.

However, velocity data from Study 2 shows the gesture is not performed at constant speed: pinch-to-zoom behaviour exhibits an early ballistic pattern, where the user uses an initial high-speed movement to perform larger scale manipulations, followed by a corrective phase for final positioning (Figure 5). This speed profile is similar to other aimed mouse movement [3].

To reinforce this behaviour, we designed a pinch-to-zoom acceleration function to dynamically adjust a zoom scale multiplier based on the velocity of pinch and spread movement. With high velocity, the zoom scale multiplier increases the rate of zoom beyond the standard pinch-to-zoom rate. With slow velocity, the standard pinch-to-zoom zoom is used. Note that Jellinek and Card [11] found that mouse acceleration functions do not provide performance benefits, although recent work has been more encouraging [3,8]. More relevant to us, Casiez et al. [3] show that mouse acceleration functions can reduce clutching. An acceleration function will let people adjust the zoom scale more quickly, and we expect that this will decrease clutching.

Our acceleration function is not only motivated by the results of the formative studies, but uses study results to tune the function parameters. Figure 9 provides pseudo code for the acceleration function. Two thresholds, min and max, are calculated using the average pinch and spread velocity and standard deviation (SD) from study 2. Min is equal to mean velocity minus 1 SD, and max is set equal to mean velocity plus 1 SD. Study 2 yields a mean velocity of 1260 px/ms (SD 1190), creating a min threshold of 70 px/ms and a max threshold of 2450 px/ms. Using these thresholds and the current velocity of pinch or spread movement, we calculate a scale multiplier to increase the standard linear zoom scale change (or decrease it when zooming out):

- If current velocity is below min, scale multiplier = 1.0.
- If velocity is between min and max, the scale multiplier is in [1.0, 2.0], linearly interpolated by current velocity.
- If velocity is above max, scale multiplier = 2.0.

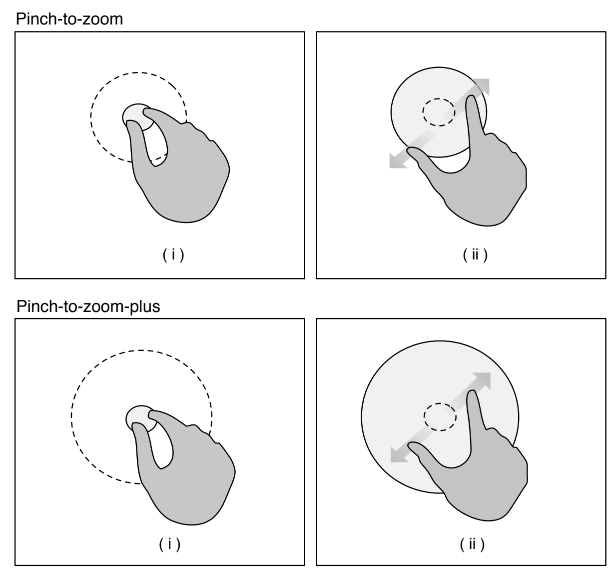


Figure 8: Zoom acceleration. Standard pinch-to-zoom (i) pinch and (ii) spread. PZP (iii) pinch and (iv) spread magnify the effects of these movements.

```
// lower and upper bounds based on mean from experiment 2
min = MEAN_VELOCITY - 1 SD
max = MEAN_VELOCITY + 1 SD

// return a multiplier from 1.0 to 2.0
if (current_velocity > min)
    return max( 1.0 + (current_velocity - min / max - min), 2.0 )
else
    return 1.0
```

Figure 9: Function to return the scale multiplier, which returns a value of 1.0 to 2.0 based on the current velocity

This change to the control-gain function results in acceleration at the start of the pinch-to-zoom gesture, which smoothly changes to standard linear movement during the low-speed tail of movement. This behaviour is similar to existing mouse pointer acceleration techniques [3].

Pan-to-Centre

Zooming requires that the user place their fingers directly over the area of interest, which can cause target occlusion [16]. We observed in our initial studies, particularly in study 2, that users typically panned this area of interest to the center of the screen before or during the zoom operation. Our goal is to partially automate this pan operation during a pinch-to-zoom gesture by moving the content of interest to the central area of the display (Figure 11).

Looking at the pan velocity in Figure 6, we note that pan follows a ballistic profile, but without the longer, low-speed tail found in zoom operations. This ballistic profile is more similar to un-aimed movements of the kind characterized by Flash and Hogan [5]. This indicates that participants panned toward the center of the display without a precise goal for the final content position. They simply wished to avoid the edge of the display. The scatterplot in Figure 7 supports this characterization of behaviour.

Pan-to-center is expressed as a translation applied directly to the pinch-to-zoom gesture. In study 2, we found that the mean distance from the target to the center of the screen at task end was 247.8 px, SD 193.7 (Figure 6). We use a similar threshold for pan-to-center, specifically mean plus 1 SD, or 441.6 px, which represents 65% of the distance to the center of the screen. In particular, if we examine Figure 7, we see that only a band around the edge of the display is avoided. This band that participants avoid represents approximately 35% of the distance from the edge of the display to the center. Our pan-to-center approximates movement into this center-region.

The pan-to-center algorithm is shown in (Figure 10). It performs as follows:

- A user places their fingers on the display, yielding a point of interest calculated as the mid-point between the user's fingers [14]. This point's coordinates are (center_x, center_y).
- The algorithm calculates a circular region in the center of the display, as described in the previous paragraph. This is the target region for the pan.
- Next, the algorithm calculates a distance remaining between the midpoint and target (lines 7 and 8).
- The algorithm gradually pans to the center of the display. The translation is attenuated such that the pan to center is gradual (less than 1/6th of the remaining distance) and stops once the target point has moved into the center region of the display (lines 11 – 16 in Figure 10). The 1/6th distance attenuation in our algorithm was an empirically determined value found to smooth the movement. During a pilot of this technique, we found that users preferred a gradual movement over a quick "snap" to the final target.

The goal of our design was to allow a 13x scale using a single clutch, but in general users still need to clutch. To support multiple clutches, the starting midpoint (center_x, center_y) is saved and re-used for each subsequent clutch operation. In this way, the initial target location, defined as the initial midpoint between finger contact points [14], is panned toward the center of the display (Figure 11). If no contact is made for a period of time (682 ms, the mean clutch time from study 1 plus 2 SD), the midpoint resets.

```
1  // mean target distance derived from experiment 2
2  // defines a target region around center of screen
3  // auto-pan to this region, then stop panning
4  center_radius = MEAN_TARGET_DISTANCE - 1 SD
5
6  // remaining distance from midpoint to display center
7  remaining_x = (center_x - current_x)
8  remaining_y = (center_y - curent_y)
9
10 // move distance of a single clutch
11 dx = min(remaining_x / 6, remaining_x * center_radius)
12 dy = min(remaining_y / 6, remaining_x * center_radius)
13
14 // adjust signs so that (dx, dy) properly translate
15 if current_x > center_x then dx = -dx
16 if current_y > center_y then dy = -dy
17
18 return dx, dy
```

Figure 10. Pan distance calculation.

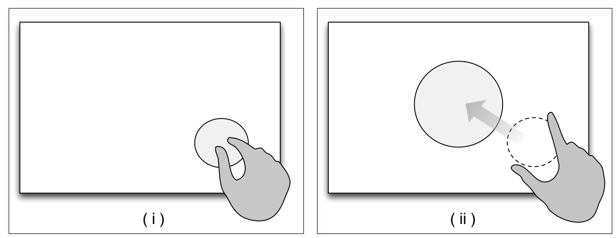


Figure 11: Pan-to-center. Zooming on a target near the edge (i) causes it to be scaled and translated to the center (ii).

EVALUATION

We performed a controlled experiment to evaluate the performance of PZP compared to standard pinch-to-zoom. Our hypothesis is that PZP will reduce task effort, measured as the number of clutches and pans required to complete the experimental task.

Task and Apparatus

We use the drawing task from study 2 (Figure 4) with the initial circular target placed at one of eight locations around the perimeter of the screen. The task is complete after the user draws a circle within the indicated region, and the zoom-level and pan are automatically reset for the next task. The same tablet from the formative studies was also used for this experiment. As in the formative studies, an overhead video camera recorded the experiment to capture additional user behaviours.

Participants

We recruited 24 participants, 16 male (15 right-handed, 1 left-handed) and 8 female (6 right-handed and 2 left-handed) with an average age of 26.0 (SD 4.7). Participants were asked about their smartphone or tablet usage, and self-reported a mean usage of 15.3 hours per week (SD 13.8). When recruiting participants, we asked for people who were “regularly used” a smartphone or tablet. Based on this filter, and the self-reported usage, we consider these users to be proficient with pinch-to-zoom. A $10 voucher was offered as remuneration.

Design

The experiment was a mixed between-subjects and within-subjects design. Within-subject factors were 2 technique conditions (Pinch-to-Zoom-Plus PZP and standard pinch-to-zoom PZ), 3 zoom levels (7x, 13x, 19x), one zoom direction (zoom-in), one tolerance (7 mm), and 8 positions along the perimeter of the screen (top-left, top-center, top-right, middle-left, middle-right, bottom-left, bottom-center, bottom-right). Hand condition (unimanual versus bimanual) was treated as a between-subject factor, with participants being randomly assigned to either the unimanual or bimanual condition. Hand condition ensures that PZP works for both unimanual and bimanual pinch-to-zoom gestures.

Technique conditions were fully crossed and presented in a randomly ordered block of trials, to balance potential learning effects. Four repetitions of each task were presented, to provide adequate time to learn the technique. Practice blocks of 12 tasks were presented at the start of each block. Not counting practice tasks, the experiment had 192 tasks per participant.

At the end of the experiment, we conducted a short interview to record participants' subjective impressions of the two techniques. The experiment took 45 minutes.

Pre-Processing

Excluding practice blocks, a total of 4632 tasks were performed across 24 participants. 140 tasks with an elapsed time of more than two standard deviations from the mean were excluded, which reduced the total tasks to 4492. Excessive time per trial was chosen for filtering data, as it presented the best indication of abnormal performance when completing a task (e.g. a participant stopping to ask a question mid-task, answering their mobile phone, etc.).

Results

Two-factor ANOVA and Bonferroni-Dunn correction were used for pairwise comparisons including the effects of handedness, acceleration and zoom level. Dependent measures were zoom-count, pan-count and elapsed time for each task (to determine the potential effects of the changes on task duration).

Zoom and Pan Actions

The mean number of zoom actions was 3.01 (i.e. 2.01 clutches). PZP reduced the number of zoom actions by 21% (a 43% reduction in clutches). Tasks completed without acceleration required a mean of 3.36 zoom actions (SD 2.26), but tasks completed with the accelerated technique only required a mean of 2.65 zoom actions (SD 1.45). This was a significant effect ($F_{1,23} = 11.63$, $p < 0.01$), indicating a reduction in zoom clutch actions. There was no observed learning effect for zoom actions on block ($F_{1,23} = 0.89$, ns).

PZP also reduced the amount of panning required by 14%: tasks completed without acceleration required a mean of 7.1 pans (SD 2.64), while tasks completed with acceleration required a mean 6.1 pans (SD 2.42). This was a significant effect ($F_{1,23} = 13.80$, $p < 0.01$). There was no panning learning effect on block ($F_{1,23} = 0.28$, ns).

Other variables (hand condition, zoom level, position) had no significant effect on zoom or pan actions.

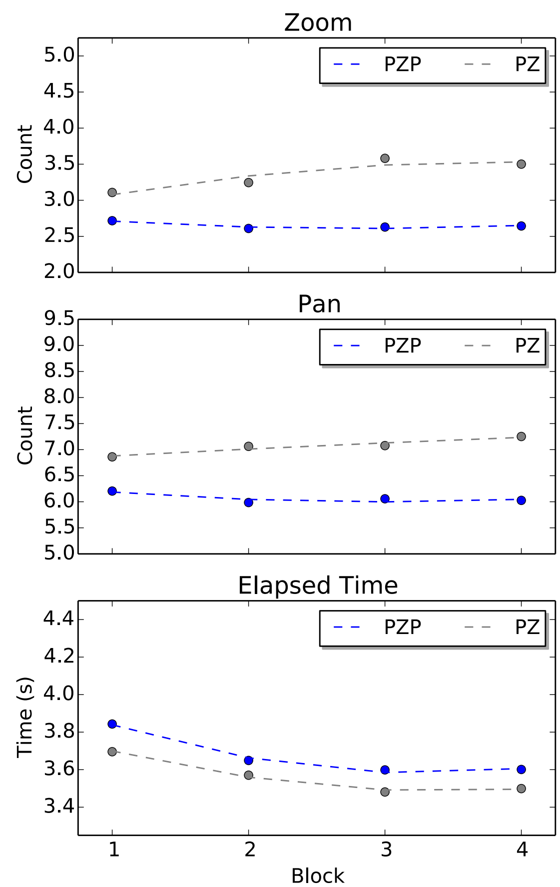


Figure 12: Measures by Block. PZP required far fewer zoom actions and pans than standard pinch-to-zoom, with comparable times required to complete the task.

Task Time

Mean task completion time was 3.62s (SD 1.25). Zoom-level, as expected, directly correlated to completion time: 7x scale tasks took 3.06s, 13x took 3.65s and 19x took 4.16s to complete, a significant effect ($F_{1,23} = 118.45$, $p < 0.01$). Position of the target had no significant impact on task completion time ($F_{1,23} = 4.95$, ns). Technique (PZP vs. PZ) had no statistically significant effect on time ($F_{23} = 1.56$, ns). Presentation order of PZP vs. PZ conditions was not significant ($F_{1,23} = 1.25$, ns) suggesting no technique carry-over effect.

There was a significant learning effect for elapsed time, where time improved across blocks. Mean times for all tasks (PZ and PZP) were 3.77s, 3.61s, 3.54s and 3.55s for blocks 1 through 4 ($F_{1,23} = 8.76$, $p < 0.01$).

User Feedback

Participants seemed quick to grasp the PZP technique. Several participants indicated that it was a fairly seamless change from current pinch-to-zoom behaviour. P2 "couldn't see a difference" between PZP and PZ behaviour. P6 also indicated that she "couldn't really tell the difference between [PZP] and [PZ] behaviour".

Those that noticed the difference seemed to like PZP, but did not always find it easy to adjust to using it. P1 said "For single-handed operation, the [PZP] technique was much better… because the automatic movement was taken care of." P4 said "I think the [PZP] case is better when I need to move the target to the center, like Google Maps." However, P5 commented, "I feel like I'm fighting [PZP], but I'm getting used to it".

Discussion

PZP had a significant impact on the number of clutch and pan actions. The number of clutches was reduced by 21% and the number of pans reduced by 14%. This demonstrates that PZP achieved its primary goals. We did not see a significant time saving associated with the reduced clutch and pan behaviours. However, we did see a persistent learning effect with decreasing time, and when observing our participants, we noted that their ingrained behaviour with standard pinch-to-zoom may have reduced the benefit from PZP. Many of our participants used multi-touch smartphones and tablets extensively prior to this study, and had high familiarity with the expected behavior of standard pinch-to-zoom. The "fighting" comment from P5 above is a fair description of participants who struggled to adapt to PZP. For these reasons, we conducted a follow-up longitudinal study.

LONGITUDINAL STUDY

To assess whether additional practice is needed to acclimatize to the behaviour of the PZP technique, we conducted a small longitudinal study. Using the same hardware and task as the controlled experiment, participants performed zooming tasks for five consecutive days using PZP. We recruited 5 participants, 2 male (both right-handed) and 3 female (1 right and 2 left-handed) with an average age of 23.0 (SD 5.1). The tasks included 3 zoom levels (7x, 13x, 19x), one zoom direction (zoom-in), one tolerance (7 mm), and 8 positions around the perimeter of the display (top-left, top-center, top-right, middle-left, middle-right, bottom-left, bottom-center, bottom-right). All tasks were performed unimanually using the participants preferred hand. Practice blocks of 12 tasks were presented at the start of each block. Six task repetitions were presented each day, providing time to learn the technique. Task and zoom/pan actions were recorded.

Results

Excluding practice blocks, a total of 720 tasks were performed across 5 participants. Zoom action count across days had little variance: mean values were 2.48 on day 1 and 2.50 on day 5 (Figure 13 top). Pan action count across days also had little variance: mean values were 5.97 on day 1 and 5.89 on day 5 (Figure 13 middle). This suggests PZP's efficacy for consistent reduction of clutch and pan actions. In addition, task time decreased significantly ($F_{1,4} = 6.4$, $p < 0.01$). Over the course of 5 days, time improved by 27.1% (Figure 13 bottom). Mean task time fell from 3.51s on day 1 to 2.56s on day 5.

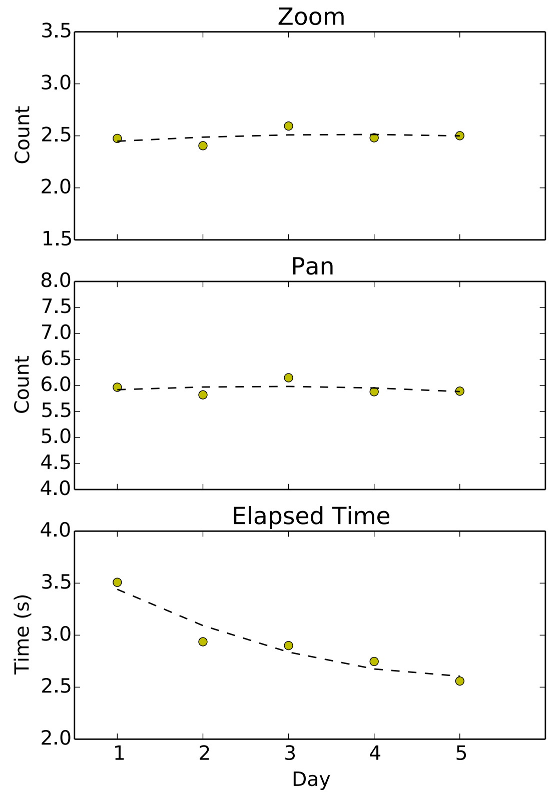


Figure 13: PZP mean action counts and task times over 5 days. Zoom and pan actions remain constant, but task time is significantly reduced over the 5 days.

DISCUSSION

We are optimistic about the results that we have achieved, but there are caveats: zoom acceleration was designed to reduce effort - specifically the number of pans and clutches - and not necessarily task time. As Casiez et al. and others point out [3,11], accelerated CD gain functions may not always reduce task time. With that in mind, our evaluation results are encouraging.

The longitudinal study was introduced to determine the "upper-bounds" of our technique, and see if task-time improvements may be possible for advanced pinch-to-zoom users. The results of the longitudinal study suggest that PZP performance may improve with additional training. Performance continued to improve over a 5-day period, so lengthy training may be necessary to see the effect. Overall, we were impressed that expert users were able to overcome their ingrained knowledge of standard pinch-to-zoom to harness the benefits of PZP enhancements.

In general, the PZP automatic pan assumes user always wish to keep a single point of interest visible by moving it near the center of the screen. However, this may not be a universally applicable strategy, since not all tasks have a well-defined single point of interest. Some map navigation tasks, such as route finding, require zooming two or more points. Users can initiate PZP at the centroid of multiple points of interest and still gain a benefit, but this may not be intuitive. A technique to selectively disable automatic panning may be a useful future refinement for PZP.

CONCLUSIONS AND FUTURE WORK

Since its inception pinch-to-zoom has become an effective, usable technique for enlarging content on multi-touch displays. However, as content increases in size and displays increase in resolution, the need to support ever-larger scale factors has resulted in inefficiencies.

In this paper, we analyze two of those inefficiencies: the performance of multiple sequential clutch operations while zooming to attain a desired scaling of content; and the performance of multiple panning translations to ensure content remains within the field of view on the display. First, we show that zoom and pan actions are quite common, even with relatively modest scale factors. Next, leveraging the kinematics and positional data associated with zoom and pan actions, we design PZP, a modified form of pinch-to-zoom that seeks to reduce these operations. Finally, through a controlled study and a longitudinal study, we show that PZP can significantly reduce the number of zoom and pan operations.

Future work will include a longitudinal study comparing the performance of PZP with standard PZ, to determine if the long-term performance gains translate into improved temporal performance. Regardless of a significant reduction in time, PZP has the potential to make zooming easier for everyone. PZP does not require specialized hardware, does not have complex algorithms, and remains backward compatible with standard pinch-to-zoom and all other multi-touch gestures. This enhanced pinch-to-zoom can be added to all multi-touch operating systems.

ACKNOWLEDGEMENTS

Research funding provided by the Natural Science and Engineering Council (NSERC), GRAND Networks of Centres of Excellence, and the Ontario Research Fund for Research Excellence.

REFERENCES

1. Albinsson, P.A. and Zhai, S. High precision touch screen interaction. *CHI '03* (2003), 105.
2. Buxton, B. Multi-touch systems that i have known and loved. *Microsoft Research*, (2007).
3. Casiez, G., Vogel, D., Balakrishnan, R., and Cockburn, A. The Impact of Control-Display Gain on User Performance in Pointing Tasks. *Human-Computer Interaction 23*, 3 (2008), 215.
4. Cockburn, A., Karlson, A., and Bederson, B.B. A review of overview+ detail, zooming, and focus+ context interfaces. *ACM Computing Surveys 41*, 1 (2008), 2.
5. Flash, T. and Hogan, N. The coordination of arm movements: an experimentally confirmed mathematical model. *The journal of Neuroscience*, (1985).
6. Forlines, C. and Shen, C. DTLens: multi-user tabletop spatial data exploration. *UIST'05*, (2005), 119.
7. Hinckley, K. and Song, H. Sensor synaesthesia. *CHI '11* (2011), 801.
8. Hinckley, K., Cutrell, E., Bathiche, S., and Muss, T. Quantitative analysis of scrolling techniques. *CHI '02* (2002).
9. Hinckley, K., Czerwinski, M., and Sinclair, M. Interaction and modeling techniques for desktop two-handed input. *UIST '98*, (1998), 49.
10. Hoggan, E., Nacenta, M., Kristensson, P.O., Williamson, J., Oulasvirta, A., Lehtiö, A. Multi-touch pinch gestures: performance and ergonomics. *ITS '13* (2013), 219.
11. Jellinek, H.D. and Card, S.K. Powermice and user performance. *CHI '90* (1990).
12. Käser, D.P., Agrawala, M., and Pauly, M. FingerGlass: Efficient multiscale interaction on multitouch screens. *CHI '11* (2011), 1601.
13. Lank, E. and Phan, S. *Focus+Context sketching on a pocket PC*. *CHI EA '04* (2004).
14. Malacria, S., Lecolinet, E., and Guiard, Y. Clutch-free panning and integrated pan-zoom control on touch-sensitive surfaces: the cyclostar approach. *CHI '10* (2010).
15. Negulescu, M., Ruiz, J., and Lank, E. ZoomPointing revisited: supporting mixed-resolution gesturing on interactive surfaces. *ITS '11* (2011), 150.
16. Sears, A. and Shneiderman, B. High precision touchscreens: design strategies and comparisons with a mouse. *International Journal of Man-Machine Studies 34*, 4 (1991), 593.
17. Tran, J.J., Trewin, S., Swart, C., John, B.E., and Thomas, J.C. Exploring pinch and spread gestures on mobile devices. *MobileHCI '13* (2013), 151.
18. Wellner, P. The DigitalDesk calculator: tangible manipulation on a desktop display. *UIST '91*, (1991), 27.
19. Wellner, P. Digital Desk. (1991).