Target Acquisition with Camera Phones when used as Magic Lenses

Michael Rohs
Deutsche Telekom Laboratories, TU Berlin
Berlin, Germany
michael.rohs@telekom.de

Antti Oulasvirta
Deutsche Telekom Laboratories, TU Berlin
Berlin, Germany
Helsinki Institute for Information Technology HIIT
Helsinki, Finland
antti.oulasvirta@hiit.fi

ABSTRACT
When camera phones are used as magic lenses in handheld augmented reality applications involving wall maps or posters, pointing can be divided into two phases: (1) an initial coarse physical pointing phase, in which the target can be directly observed on the background surface, and (2) a fine-control virtual pointing phase, in which the target can only be observed through the device display. In two studies, we show that performance cannot be adequately modeled with standard Fitts' law, but can be adequately modeled with a two-component modification. We chart the performance space and analyze users' target acquisition strategies in varying conditions. Moreover, we show that the standard Fitts' law model does hold for dynamic peephole pointing where there is no guiding background surface and hence the physical pointing component of the extended model is not needed. Finally, implications for the design of magic lens interfaces are considered.

Author Keywords
Target acquisition, magic lens pointing, Fitts' law, human-performance modeling, camera phone, augmented reality.

ACM Classification Keywords
H.5.2 [Information Interfaces and Presentation]: User Interfaces – input devices and strategies, interaction styles, theory and methods.

INTRODUCTION
This paper examines one-handed target acquisition in a situation in which a camera-equipped device acts as a movable window, or *magic lens* [3], on a large surface and overlays virtual information on the camera view (see Figure 1). We examine the selection of targets under varying sizes and distances in two experiments. To anticipate the main result, we found that standard Fitts' law [6] does not adequately model performance with magic lens interfaces, because the conditions of the visual feedback loop change during the movement, whereas it does adequately model the case in which no visual context is given outside the device display, i.e., when the handheld device acts as a *dynamic peephole* [24] or *spatially-aware display* [7].

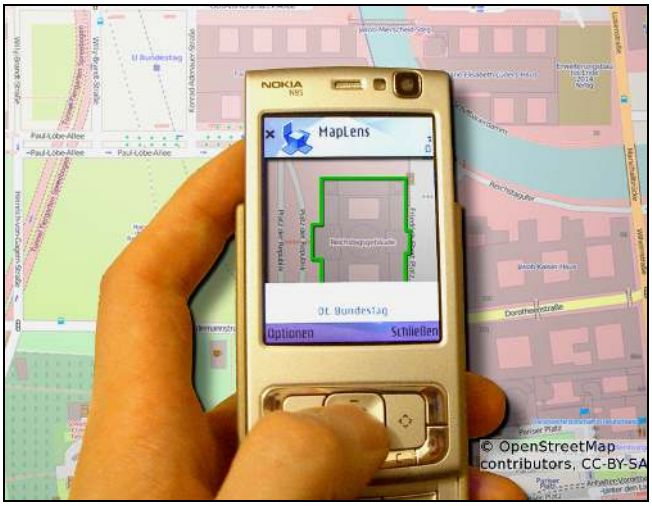

Figure 1. Magic lens pointing over printed map (constructed). Additional information is overlaid on recognized objects, and these objects can be selected for more information.

In order to explain the observed difference between these two types of selection tasks with camera phones, we present a two-part modification of Fitts' law that improves prediction of cameraphone-based selection performance in the magic lens pointing case. A key idea of the model is to split interaction in two parts, one for initial targeting by direct observation and the second one for targeting through the magic lens. For high-precision touch-screen pointing Sears and Shneiderman [19] proposed a two-stage model with five parameters that includes a term for gross arm movement and a term for fine-tuning motions of the fingers. However they write that the analysis of their modification was inconclusive and do not provide any experimental data.

CHI 2008, April 5–10, 2008, Florence, Italy.

Magic Lens Pointing

The term *magic lens* is used here to denote augmented reality interfaces that consist of a camera-equipped mobile device being used as a see-through tool. It augments the user's view of real world objects by graphical and textual overlays. When the device is held above an object or surface, for example a map, visual features in the scene can be highlighted and additional information overlaid in real-time to objects on the device's display (see Figure 1).

There are many applications envisioned and implemented. For example, a map at a bus stop can show a graphical overlay depicting the current positions of busses. In tourist guide and city applications, information on various sights and events can be accessed by moving the phone to the respective targets and observing the graphical overlays on the mobile device's display [12,18]. In gaming applications, a poster or paper can represent fixed portions of the game space, for example the goal frame in a soccer penalty shootout game, and the rest of the game is to be viewed and interacted with through the magic lens that recognizes its position and orientation on the fixed frame [17].

Whereas magic lens interfaces are based on the idea of real-time augmentation of the real world scene, *peephole interfaces* [24] denote a class of interfaces where the viewport of a mobile device is used as a window into a *virtual* workspace and no visual context is available outside the display. Traditional *static* peephole interfaces move the virtual workspace behind the static peephole, whereas *dynamic* peephole interfaces move the peephole across a static workspace [15]. The latter require a spatial tracking method in order to compensate for the movement of the peephole, such that the workspace appears at a constant position in space. Yee [24] presents several example applications, such as a drawing program and a personal information space anchored to the user's body as a frame of reference.

Magic lens pointing can be regarded as an extension of dynamic peephole pointing, in which additional visual context is provided in the background. Both are ways of improving information navigation on handheld devices and overcoming the limitations of the display size. Since typically only a small part of a document can be visualized on a handheld device display at a time, the user needs effective mechanisms to continuously navigate to different parts of a document in order to mentally create a holistic understanding. Magic lens pointing appears to be a particularly promising kind of interaction, since it allows augmenting large scale information presentation with private and up-to-date information on the personal display. The background surface can be a passive printed paper document or an active electronic display. The large scale background surface allows the user to quickly and effectively acquire the global structure of a document and then examine a small focus area in detail. A large scale city map, for example, allows for a much quicker orientation than the small device display alone.

Target acquisition or *pointing* is a fundamental gesture in today's human-computer interfaces and has thus been thoroughly researched in numerous studies and for a wide range of input devices [10,14,21]. As a tool for predicting the time for directed movements, Fitts' law [6] has been used extensively in human-computer interaction. An excellent overview of Fitts' law research is provided by MacKenzie [14]. According to Fitts, the movement time for a target pointing task involves a tradeoff between speed and accuracy: The larger the distance to be covered and the smaller the size of the target, the higher the movement time. While Fitts' experiments only examined one-dimensional movements, Fitts' law has also been shown to hold for two- [14] and three-dimensional movements [9,16].

When visual feedback on the movement cannot be directly observed, but is mediated by some sensing mechanism, lag and update rate play a role. The effects of lag and update rate in mediated visual feedback have been evaluated by Ware and Balakrishnan [23], Graham and MacKenzie [8], and others. Magic lens pointing, which we investigate in this paper, has unique characteristics in that during the first phase of the interaction the target and the hand's movement towards the target can be directly observed, while during the second phase the target is occluded by the magic lens and can only be observed through the display, which introduces some non-negligible delay in the visual feedback.

Camera-based interaction with the above mentioned interfaces can be understood in terms of a Fitts' task. Wang et al. [22] show that optical flow processing on a camera phone follows Fitts' law. For both magic lens and dynamic peephole pointing the fundamental components of interaction are rapid precise movements towards a point target or a spatially extended target. Consequently, according to Fitts' law movement time in such a task depends on the distance to be covered and the size of the target. Nevertheless, there are important differences between the case of camera-based selection and the general case of 2D selection:

- Area selection [11] instead of point selection. Depending on the implementation, the complete target might have to be present in the camera image to be recognized by the system.
- Screen distance range. Depending on the granularity of visual features of the background surface, there is a certain distance range within which the phone can detect those features. The user has to adapt selection distance accordingly.
- Delay introduced by the system. When targets are observed through the display rather than directly on the background surface an additional delay is introduced by the recognition system. This delay is detrimental to performance [23].
- Maximum movement velocity. The upper limit of the movement velocity is bound not only by the user's motor capacity, but also by the limits of the recognition system. If the movement quickly sweeps over the surface, it

might appear blurred in the camera image, which reduces the probability of recognition.

- Display update rate. The frame rate of the camera – and hence the update rate of the display – is limited. It lies typically between 15 and 30 Hz on current devices. Yet, this is sufficient for perception of a smooth movement.
- Device movement takes place in 3D space. In comparison to the original experiments of Fitts, the z-coordinate of the cursor position has an effect on the appearance (size and angle) of the target. Moreover, the target can be selected from a wider selection space than what is possible with many other pointing devices. Taken together, these factors may lead to more variable selection trajectories and more variable involvement of muscle groups.
- Gaze deployment between figure (device screen) and ground (background surface). The phone shows an augmented view of the background, but the hand occludes part of the background. The user has to decide whether to acquire information from the background or through the magic lens and has to move hands so as to not occlude required information.

ANALYSIS

In dynamic peephole interfaces the target can only be observed through the device display when the device is positioned over the target. The target is not present in the physical world. In this case the basic Fitts' law [14]

$$MT = a_o + b_o\, ID \quad \text{with} \quad ID = \log_2(D / W + 1) \qquad (1)$$

is expected to lead to a good prediction of movement times. *MT* is the movement time that the model predicts. *ID* is the index of difficulty [6], *D* is the distance from the starting point to the target, and *W* is the width of the target. Lag and low frame rates increase the coefficients a_o and b_o compared to direct observation of the target [23].

Our hypothesis is that with magic lens pointing the situation is different because there is an initial phase in which targets can be directly observed and a second phase in which the view on the target is mediated through the device. We try to justify this hypothesis in the analysis below.

The magic lens situation is depicted in Figure 2. We denote the first phase of magic lens pointing as *physical pointing*: The target (denoted as *T* in Figure 2) can be directly observed in the physical world. At some point during the movement towards the target, the target falls below the magic lens and can no longer be observed directly, but only through the magic lens. With a screen width of *S* the split point is located at a distance of *S*/2 at which half of the target is visible on the screen and half of it can be directly observed. If we postulate a virtual target of width *S*, centered at the real target *T*, the first phase can be modeled as (see Figure 2, left):

$$MT_p = a_p + b_p \log_2(D / S + 1). \qquad (2)$$

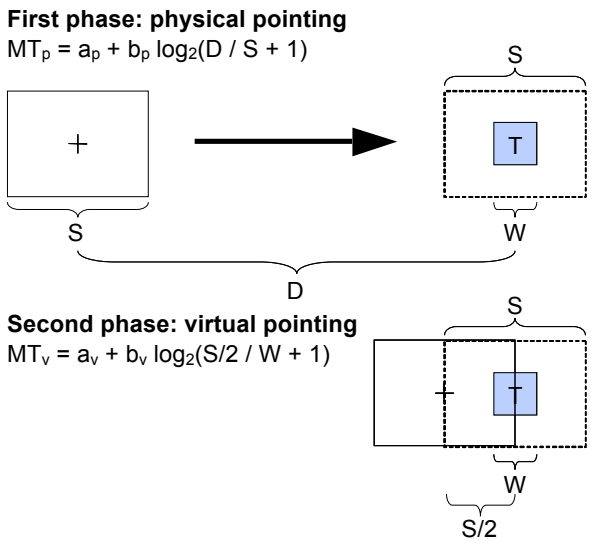

Figure 2. Magic lens pointing is split in a direct observation phase (physical pointing) and a device-mediated phase (virtual pointing). Movement proceeds from left to right.

At the split point, the second phase – *virtual pointing* – begins: The target can now only be observed through the device. The second phase starts at a distance of *S*/2 and can be modeled as (see Figure 2, bottom)

$$MT_v = a_v + b_v \log_2(S/2 / W + 1). \qquad (3)$$

If we attribute half of the transition period between physical to virtual pointing to each of the two, the total movement time for the two-part Fitts' law model is

$$MT = MT_p + MT_v$$
$$= a_p + a_v + b_p \log_2(D / S + 1) + b_v \log_2(S/2 / W + 1)$$
$$= a + b \log_2(D / S + 1) + c \log_2(S/2 / W + 1). \qquad (4)$$

As soon as a target falls below the lens, the characteristics of the mediation by the camera-display-unit come into play. As summarized in the introduction, these include delay, display update rate, maximum distance and movement speed at which targets are recognized. Moreover, especially for small targets, *jitter* – noise in the cursor position – becomes an issue. Delay in particular has a direct influence on the control loop that governs rapid aimed movements.

Control Loop in Physical Pointing

It has been found that movements longer than 200 ms are controlled by visual feedback [14]. The Fitts' law constants *a* and *b* can thus be interpreted in terms of a *visual feedback loop* or *control loop* that is assumed to underlie targeted movements. The *deterministic iterative corrections model* [5] assumes that a complete movement is made up of a series of *n* ballistic submovements, each taking a constant time *t* and covering a constant amount 1-ε of the remaining distance. Thus the first submovement starting at distance *D* ends at distance εD, the second starts at εD and ends at $\varepsilon^2 D$, and so on, until a submovement ends within the target, i.e., $\varepsilon^n D \leq W/2$. Solving for *n* yields $n = \log_\varepsilon(W / (2D)) = k$

$\log_2(2D / W) = k\ ID_{orig}$ with $k = -1/\log_2(\varepsilon)$ and ID_{orig} the original formulation of the index of difficulty [6]. The total time is $n\ t = -\log_2(2D / W)\ t / \log_2(\varepsilon)$. Estimates for t are in the range of 135 to 290 ms and for ε 0.04 to 0.07 [14].

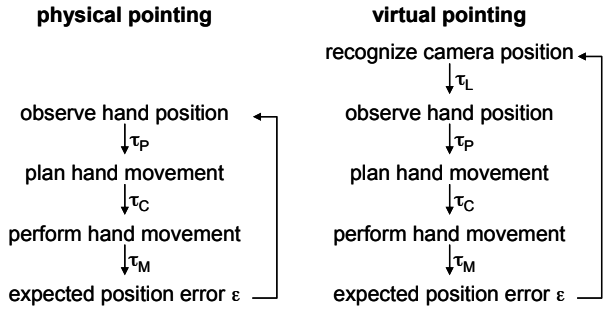

Figure 3. Control loops for physical pointing (left) and virtual pointing (right).

The movement process starts with detecting the stimulus and initiating a ballistic movement. In physical pointing (Figure 3, left), in which targets are directly visible and not mediated through the device, the control loop consists of perceiving the current distance to the target, planning the next ballistic micromovement, and effecting hand movement. In their Model Human Processor [4] Card et al. assume characteristic durations of τ_P = 100 ms for the Perceptual Processor, τ_C = 70 ms for the Cognitive Processor, and τ_M = 70 ms for the Motor Processor to perform these tasks. Hence the total duration for one cycle is $t = \tau_P + \tau_C + \tau_M$ = 240 ms, which is in the range cited in [14].

Control Loop in Virtual Pointing

Ware and Balakrishnan [23] analyze the contributions of lag and frame rate to the constant b in basic Fitts' law (1). If the observation of the targets is mediated by the device – i.e., the targets are only visible through the device display – then a *machine lag* component is introduced into the control loop (see Figure 3, right). In both magic lens and dynamic peephole pointing, the integrated camera of the device is used as a position sensor. Images are taken at regular intervals, for example with a frame rate of 15 Hz. First, there is a delay m_1 caused by the image acquisition hardware, i.e., when a frame reaches the tracking algorithm it shows the situation m_1 milliseconds ago. The time the algorithm needs to process the frame adds another component m_2. The time to render the result on the display is m_3. Hence when the sensed position becomes visible on the display it shows the situation $\tau_D = m_1 + m_2 + m_3$ milliseconds ago. Assuming a uniform distribution of the perception in the frame interval T_F, the total machine lag is on average $\tau_L = \tau_D + 0.5\ T_F$. With the devices and algorithms used in the experiments, the total machine lag amounted to 118 ms for Experiment 1 (magic lens pointing, Nokia 6630) and 294 ms for Experiment 2 (dynamic peephole pointing, Nokia N80). In the setup we used, the computational complexity of the dynamic peephole interface was higher than for the magic lens interface, which required a more powerful device.

Equation (4) can be rewritten in terms of the time needed to make a corrective submovement t and in terms of machine lag τ_L if we write b and c as $b = \beta\ t$ and $c = \gamma\ (t + \tau_L)$ [23]:

$$MT = a + \beta\ t \log_2(D / S + 1) + \gamma\ (t + \tau_L) \log_2(S/2 / W + 1) \quad (5)$$

To empirically assess the two-part Fitts' law model derived in this analysis, we conducted two experiments. The first experiment examined magic lens pointing, and the second dynamic peephole pointing.

EXPERIMENT 1: MAGIC LENS POINTING

The experiments were carried out utilizing the *cyclical multi-direction pointing task paradigm* of ISO 9241-9 [13]. Put briefly, there are nine targets visible; on a large background surface in Experiment 1 (Figure 4, left), and in the virtual plane in Experiment 2 (Figure 4, right). One target at a time is marked in a circle of nine targets, and that target should be selected by moving the crosshair on the phone's display and pressing the joystick button. Preferring the multi-directional over the one-directional task was natural, because in real world applications objects and areas are dispersed on a larger area surface.

Method

Participants

Twelve subjects (8 male, 4 female, age 19-31) were recruited, most from TU Berlin and the rest from college-level institutes. Ten subjects were right-handed, one was left-handed and one ambidextrous. Only two used the camera on their camera phone regularly. The subjects were paid a small incentive for participation. All subjects were healthy and had normal or corrected-to-normal vision.

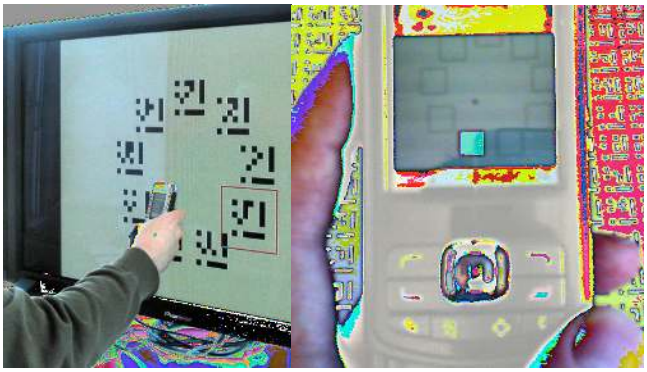

Figure 4. The magic lens pointing task of Exp. 1 (left) and the dynamic peephole pointing task of Exp. 2 (right).

Experimental Platform

The experiment was conducted on a custom-tailored system consisting of a PC communicating over Bluetooth with the mobile phone to control the highlighting of the target item on a large display (see Figure 4, left).

A Nokia 6630 (weight 130 g) was utilized as the selection device. Its camera frame rate is 15 fps, the resolution of the view finder is 160x120 pixels and the display area is 32x24 mm. The application on the phone was written in

Symbian C++. It showed the camera view finder stream and a crosshair in the center of the screen that indicated the cursor hotspot position. The application also highlighted recognized visual markers in the camera image with yellow rectangles. Users could select a recognized visual marker by pressing the phone's joystick button. A Java application on the PC received user input via Bluetooth and updated the display between the trials accordingly.

The targets were presented on a 43" Pioneer plasma display (1024x768 / 16:9) in an area of 72x54 cm (4:3 mode). The display center was positioned 1.5 m above the floor. The display showed 9 visual markers in black on white background in a circular arrangement with an angular spacing of 40°. The to-be-selected target was presented with a red frame appearing around the visual code.

Standing position in front of the display was fixed by positioning a stopper on the floor to a distance where the subject could touch the screen with an extended arm.

Task and Design

In the cyclical selection paradigm, targets always appear in the same order: starting from the top item, the next item is always opposite and slightly clockwise from the selected one. One *block* consists of all nine items selected three times. The subjects were instructed to select the highlighted item as quickly and accurately as possible. Even though within a block the subjects know where the next target will be, they still have to perform a goal-directed movement as fast as possible.

As in the classic Fitts' law studies [6], we varied target width *W* and distance *D*. The obtainable *W* and *D* combinations were limited by the size of the plasma display and the minimal marker size that the system could recognize. *W* ranged from 13 to 97 mm, with steps of 6.5 mm. Distances between successive targets ranged from 55 to 535 mm. For each target width, three distances were specified to cover a wide range of *index of difficulty* (*ID*) values: The minimum distance such that the targets on the sphere would not overlap; the maximum such that all targets would fit on the large display; and a distance with *ID* computed as the mean of the above. 33 combinations of *W* and *D* were generated in this way. Each W, D combination was held constant for three rounds (27 selections), after which another W, D pair was selected. Each participant was presented with a unique randomly generated permutation of the combinations.

Altogether 9 non-randomized practice blocks were carried out by each subject. Thus, the total number of selections per subject was 9 blocks x 3 rounds per block x 9 selections per round = 243 selections for practice; and 33 blocks x 3 rounds x 9 selections = 891 selections for the actual experiment.

Procedure and Instructions

Before starting a block, the subjects were allowed to "*calibrate*" *z-distance*, i.e., they were free to move the device on top of the first target, in order to fix an effective distance from the camera to the surface according to personal preferences (the only instruction was that performance should be as quick and accurate as possible). To start the block, the subjects had to move the crosshair in the view finder on top of the target and press the joystick button. If a target was missed, a brief beep sound was played. In such a situation, subjects were instructed *not* to try to correct the error, but to continue to the next target. After each block, there was a resting period of at least 15 seconds and, after the experiment, background information of the subject and verbal accounts of selection strategies were collected.

Results

The experiment yielded 10692 data points (12 subjects x 33 conditions x 3 rounds x 9 selections). Responses for which the system could not detect a marker (3% of the responses) and first selection in a block (4% of the responses) were not included in the *movement time (MT)* analysis. These removals left 9940 data points.

Mean Movement Time and Error Rate

Collapsed over the experimental conditions, the mean *MT* was 1.22 sec (standard deviation SD = 0.49 sec) with a relatively high error rate of 7%. An ANOVA on the error rate showed a significant effect of *W* ($F_{13,143}$=23.3, p<0.001). As shown in Figure 5, the error rate was high for small targets only. This is partly due to the limits of automatic marker recognition being pushed by hand jitter. For targets greater than 40 mm, the error rate is below 4%. This is quite comparable to reports of other input devices on mobile phones, such as joysticks [20].

Participants' performance improved during the experiment. The slope and intercept of the regression line were -0.009 and 1.354, respectively. The small slope implies that only minor learning effects occurred after the practice trials.

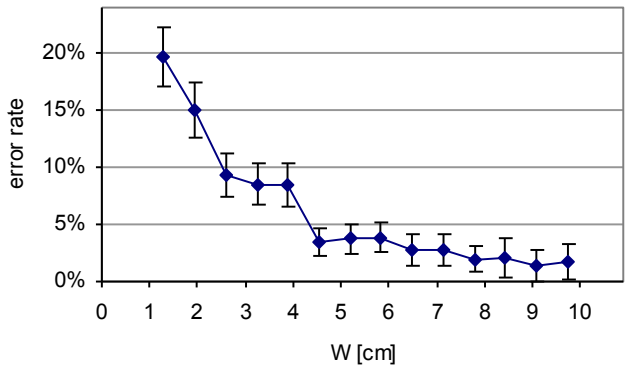

Figure 5. High error rates for magic lens pointing concentrate at target widths of less than 4 cm.[1]

[1] In all figures the vertical bars denote 95% confidence intervals, unless otherwise noted.

Effects of Width and Distance on Movement Time

MT decreases with growing *W*, but levels off at about 50 mm. Further increasing *W* does not decrease *MT*. The effect of target distance *D* on *MT* is more complex due to interaction with *W*.

We observed fast movement times and low error rates for large-enough *ID* values, enabling what was called the *line-of-sight selection strategy*. In such a situation the silhouette of the device is used as a selection cue, enabling more attention to eye-hand-coordination than the display of the device. The strategy leads to superior performance, but is only possible when the targets are not too densely spaced so that occlusion of the target by the device occurs only in the final phase of selection.

Z-Distance between Camera Phone and Target on Screen

The size with which the target appears on the magic lens display depends on the vertical distance (*z-distance*) between the camera lens and the background surface. The closer the camera, the larger the target appears on the display; the further away, the smaller the target gets. There was a linear relationship mapping target width *W* to z-distance (linear regression $R^2 = 0.98$). On average, large targets were selected from a distance of 22 cm, small ones from 10 cm, medium-sized targets falling in between.

Figure 6 plots z-distance by target width *W*. The boxplots for each *W* show the 25% quartile, the median, and the 75% quartile of the z-distance, as well as the minimum and maximum values. The three lines with different slopes are:

- Blue line (a): The closest z-distance such that the complete target is contained in the camera image. The target fills the whole display.
- Red line (b): The maximum z-distance at which the target is still reliably recognized (larger distances are possible). The target appears very small on the display.
- Green line (c): The z-distance at which cursor pointing turns into view pointing (see below). The height of the target equals half the display height.

Cursor Pointing and View Pointing

The traditional case is *cursor pointing*, which involves translating a point cursor onto the target. *View pointing* is defined as adjusting the view such that the target becomes visible and the view contains all parts of the target. View pointing has been defined in the context of multiscale user interfaces, in which pointing involves navigation to the appropriate scale to make the target visible [11]. Fitts' law then becomes $ID = \log_2(D / |W_1 - W_2| + 1)$, where W_1 is the width of the view and W_2 is the width of the target visible. The border distance between cursor pointing and view pointing is reached when the target's height is half the height of the camera image. Beyond that, the presence of the complete target on the display implies that the cursor is on the target. The figure shows that participants preferred cursor pointing over view pointing, because the median z-distance is always located above line (c). Even the 25% quartile is mostly located above (c). In the following we hence focus the discussion on traditional cursor pointing.

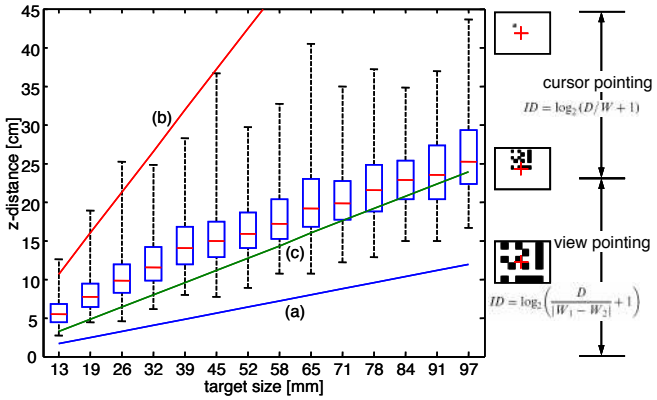

Figure 6. Vertical z-distance of camera to target by *W*.

Fitts' Law Modeling

For the basic Fitts' law model we only used target distance *D* and target width *W* as independent variables, with $ID = \log_2(D / W + 1)$. For all 33 combinations of *D* and *W* we computed mean *MT* values. Each combination contains 289 selections on average. In each group outliers of more than 2 SD from the mean were removed in this calculation.

We follow the reasoning (see below) of Ware and Balakrishnan [23] and analyze the data in terms of the unmodified index of difficulty, using the real target width rather than the *effective target width* W_e [14, Sect 3.4]. W_e is computed post-hoc based on the standard deviation of the spread around the target. The aim of the modified *ID* is to provide a more accurate measure of the rate of information processing. The first reason to use the unmodified *ID* is that it accounts for more of the variance. The second reason is that it can be used to predict actual performance in a particular situation, since it is based on the real target width.

For basic Fitts' law the resulting linear regression has a poor fit with only $R^2 = 0.57$ (adjusted $R^2 = 0.54$, Figure 7):

$$MT = 0.581 + 0.191 \log_2(D / W + 1), \quad R^2 = 0.57$$

When including all outliers the resulting regression is:

$$MT = 0.629 + 0.192 \log_2(D / W + 1), \quad R^2 = 0.53$$

The main variability was located at small *ID* values. This reflects the conditions in the experiment where targets were densely spaced and participants relied on a more display-based strategy: The targets were observed through the magic lens for a longer time than for the larger *ID* values, i.e., the duration of the first phase (physical pointing) relative to the duration of the second phase (virtual pointing) was shorter for small *ID* values than for larger *ID* values. With increasing *ID* values, i.e., more widely spaced targets, the physical pointing phase was longer relative to the virtual pointing phase, enabled by the fact that the phone occludes the targets later in the pointing process.

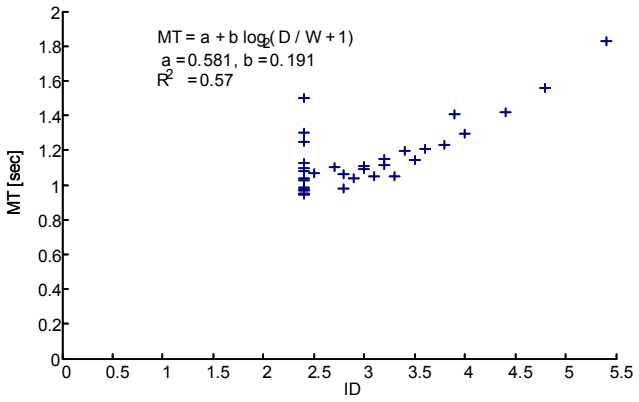

Figure 7. Magic lens pointing results in low correlation with basic Fitts' law, especially for low ID values ($R^2 = 0.57$).

In Figure 8 the basic Fitts' law prediction is illustrated in a slightly different way. For each of the (D,W) combinations it shows on the x-axis the movement time predicted by the model and on the y-axis the movement time actually measured. For a perfect model, all data points would lie on the bisecting line. For the basic Fitts' law model there is a particularly large spread of measured MT values (0.95-1.50 sec) for a predicted MT of 1.04 sec. These (D,W) combinations all have the same $ID = \log_2(D / W + 1) = 2.4$ and denote the most densely spaced targets.

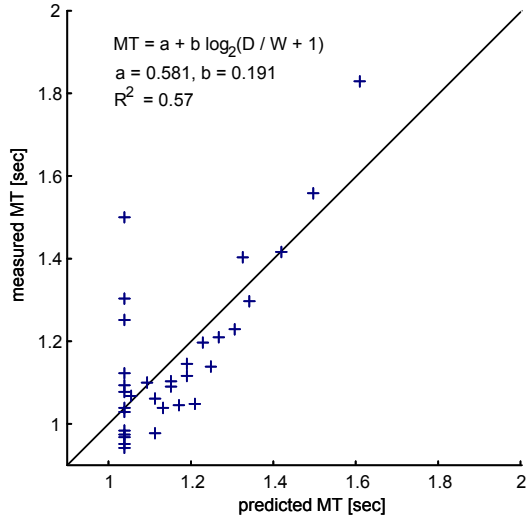

Figure 8. Same data as in Figure 6. Comparing measured and predicted movement time ($R^2 = 0.57$).

When modeling the situation with the two-part Fitts' law for magic lens pointing and the same set of independent variables (D and W), the prediction is much better (R^2 = 0.88, adjusted R^2 = 0.87). On the Nokia 6630 we used a display area of 3.2 x 2.4 cm, thus we set the split point S to the average of width and height (S = 2.8 cm).

$$MT = 0.383 + 0.112 \log_2(D / S + 1) + 0.811 \log_2(S/2 / W + 1), \quad R^2 = 0.88$$

When including all outliers the resulting regression is:

$$MT = 0.443 + 0.107 \log_2(D / S + 1) + 0.839 \log_2(S/2 / W + 1), \quad R^2 = 0.86$$

The parameters were estimated with the *MATLAB* function *nlinfit* for nonlinear modeling. It returns the least squares parameter estimates, i.e., the parameters that minimize the sum of the squared differences between the observed responses and their fitted values.

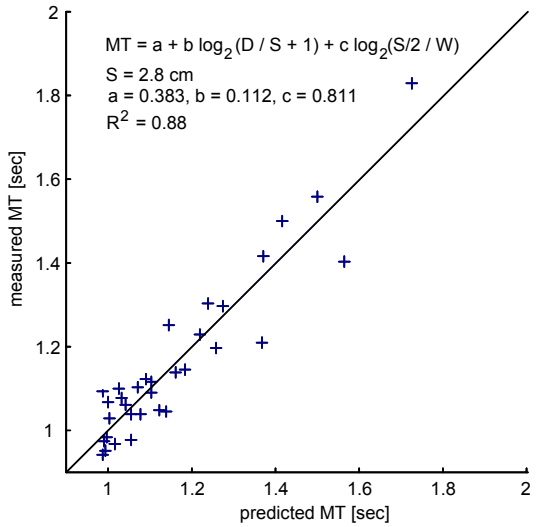

Figure 9. The two-part Fitts' law achieves a considerably better fit for magic lens pointing ($R^2 = 0.88$).

When treating S as a parameter in the above equation, *nlinfit* estimates S to 2.99 cm (2.52 cm without outlier removal), which is close to the value plausible for the display size. When using this value of S, the correlation reaches the same $R^2 = 0.88$ (adjusted $R^2 = 0.87$):

$$MT = 0.384 + 0.113 \log_2(D / 2.99 + 1) + 0.776 \log_2(2.99/2 / W + 1), \quad R^2 = 0.88$$

When including all outliers the resulting regression is:

$$MT = 0.441 + 0.105 \log_2(D / 2.52 + 1) + 0.902 \log_2(2.52/2 / W + 1), \quad R^2 = 0.86$$

EXPERIMENT 2: DYNAMIC PEEPHOLE POINTING

In this experiment, the targets were not visible on a physical surface but only in the virtual space. A physical surface of A0 size was utilized for the phone to recognize its position in the 3D space.

Method

The method was similar to the first experiment in many respects. In the following we explain all differences.

Participants

Twelve subjects (10 male, 2 female, age 22-33) were recruited from the Helsinki University of Technology. Eleven subjects were right-handed and one was left-handed. Three had more than sporadic experience with camera phones. The subjects were not paid for their participation. All subjects were healthy and had normal or corrected-to-normal vision.

Experimental Platform, Task, Procedure, and Instructions

A Nokia N80 (weight 134g) was utilized in the experiment. It features a camera able of 15 fps and a 3.5x4.1 cm display with a resolution of 352x416 pixels. As in Experiment 1, a crosshair in the center of the screen indicated the cursor hotspot position. The targets were rendered on the screen according to 3D position recognized from the camera image. Again, the target was highlighted with a red frame and the subject should select it by pressing the phone's joystick button. All feedback (beeps for errors and highlighting of target items) were provided on the mobile device.

The targets were again circularly arranged. The circle was always centered at the middle of the tracking surface. At the beginning of each block participants were instructed to move across the tracked area to learn the positions of the targets. Following a camera view model, participants could get more overview by pulling back from the tracking surface to zoom out. Beyond that, no visual aid was given during the trials.

A visual marker grid printed on a landscaped A0 paper sheet was used as the background surface for recognizing the position of the phone. The size of the tracked area was the same as in Experiment 1.

Results

Experiment 2 yielded 10572 data points (12 subjects x 33 conditions x 3 rounds x 9 selections; 120 selections were lost due to a participant accidentally exiting the test application). First selections in a block (4% of the responses) were not included in the *MT* analysis, leaving 10176 data points.

Mean Movement Time and Error Rate

Collapsed over the experimental conditions, the mean *MT* was 2.13 sec (standard deviation SD = 1.25 sec) with an error rate of 5%. ANOVA on error rate again showed a significant effect of *W* ($F_{13,143}$=14.2, p<0.001). As shown in Figure 10, the highest error rates again appear at low target widths. For target widths *W* > 4.8 cm the error rate is below 4%. Error rates are lower than for physical pointing, possibly because overall movement speed was slower in the dynamic peephole case. Participants could not see the targets in the first phase of pointing and hence chose another point in the speed-accuracy tradeoff.

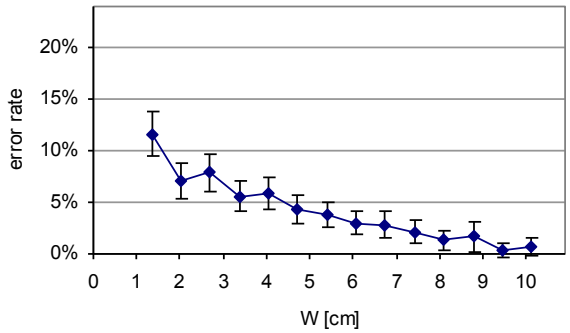

Figure 10. High error rates for dynamic peephole pointing concentrate at target widths of less than 4 cm.

Again, participants' performance improved during the experiment. The slope and intercept of the regression line were -0.0121 and 2.3274, respectively. Thus, as in the magic lens pointing experiment, only minor learning effects occurred after the initial practice trials.

Fitts' Law Modeling

In the analysis below we follow the same groupings (D,W) as in Experiment 1. Again, in each group outliers of more than 2 SD from the mean are excluded, unless otherwise noted. In dynamic peephole target acquisition the basic Fitts' law model leads to quite accurate predictions (R^2 = 0.93, adjusted R^2 = 0.93, see Figure 11):

$$MT = -0.636 + 0.881 \log_2(D / W + 1), \quad R^2 = 0.93$$

When including all outliers the resulting regression is:

$$MT = -0.835 + 0.981 \log_2(D / W + 1), \quad R^2 = 0.93$$

For the basic Fitts' law model the spread of measured *MT* values is considerably lower than with magic lens pointing.

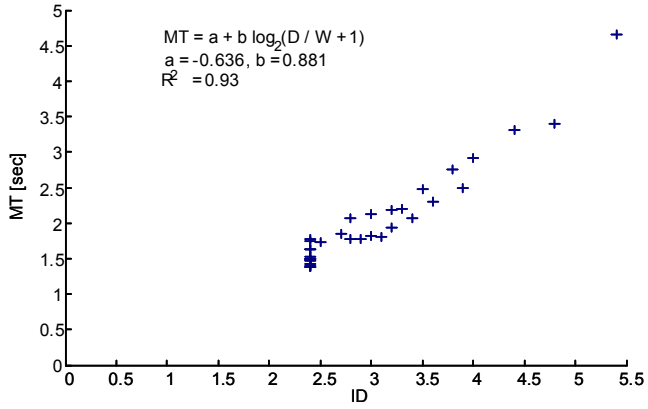

Figure 11. Dynamic peephole pointing produces a high correlation with basic Fitts' law (R^2 = 0.93).

If we nonetheless use the two-part Fitts' law model for dynamic peephole pointing for the same set of independent variables *D* and *W*, the prediction becomes marginally better (R^2 = 0.95, adjusted R^2 = 0.94). We performed Experiment 2 on a Nokia N80 and used a display area of 3.5 x 2.9 cm. We again set the split point S to the average of width and height (*S* = 3.2 cm). The resulting regression is

$$MT = -2.377 + 0.935 \log_2(D / S + 1) + 2.426 \log_2(S/2 / W + 1), \quad R^2 = 0.95$$

When including all outliers the resulting regression is:

$$MT = -2.744 + 1.028 \log_2(D / S + 1) + 2.735 \log_2(S/2 / W + 1), \quad R^2 = 0.96$$

When we treat screen size S as a third parameter, the resulting correlation does not change (R2 = 0.96, adjusted R2 = 0.96), but with 22.4 cm (18.53 cm without outlier removal) the estimate of S is far from the real value:

$$MT = 1.887 + 1.600 \log_2(D / 22.44 + 1) + 0.984 \log_2(22.44/2 / W + 1), \quad R^2 = 0.96$$

When including all outliers the resulting regression is:

$$MT = -2.210 + 1.618 \log_2(D / 18.53 + 1) + 1.176 \log_2(18.53/2 / W + 1), \quad R^2 = 0.96$$

That the same prediction accuracy can be reached with the two-part model is not surprising, since every Fitts' law task can be regarded as made up of smaller component Fitts' law tasks with identical characteristics in terms of delay.

DISCUSSION

Augmented reality interfaces project digital information on the real world scenery in real time. In augmented reality interaction using camera-equipped mobile devices in particular, this layer is visualized through the narrow viewport of the device. Since the digitally projected space does not fit in the display all at once, the user must actively move the "keyhole" to explore the space and its objects of interest, all the time relying on system feedback for the identity and location of overlaid objects. The limits of the acuity of the human visual system, the physical size of displays in mobile devices, and the computational costliness of real-time processing of visual image data all speak for the claim that this problem cannot be expected to disappear for a while.

Although augmented reality interaction on mobile devices have this one characteristic in common, even a cursory examination of the numerous application ideas reveals that interaction types fall into two quite distinct categories:

- the objects of interest are visible on the physical surface used for positioning or
- the objects of interests do not map to the real world but exist only in the projected virtual space.

In the first case the augmented reality information is projected on real world objects, meaning that these physical objects unambiguously mark the location of the digital information. In the second case only the real world space is utilized, all projected objects are "new" and there is no direct mapping to features of the environment. Analyzing these two situations, we arrived at the conclusion that these two situations map to two different interaction tasks known in the literature: magic lens pointing and dynamic peephole pointing, respectively.

It has not been previously reported how users perform in these two situations and if our standard methods of modeling apply here. To address this issue, we reported two controlled experiments utilizing the cyclic pointing paradigm. The results indicate that there is a fundamental difference in the nature of the tasks themselves.

We found that the standard Fitts' law model predicts performance quite well in the dynamic peephole pointing task (adjusted $R^2 = 0.93$), but not in the magic lens pointing task (adjusted $R^2 = 0.54$). The presumption of the standard model that the feedback loop is governed by constant processing times throughout the interaction is violated in the magic lens case. Building on the iterative corrections model, we extended the basic model by splitting it into two parts – one that describes visual feedback via direct observation and one mediated by the magic lens. We arrived at a model that includes just one additional parameter in comparison to the basic model. This model predicts movement time in the magic lens case much better (adjusted $R^2 = 0.87$). When treating the split point, i.e., the display size of the magic lens, as an additional parameter, the least squares parameter estimation predicts a display size close to the actual display size. This supports the validity of the model.

DESIGN IMPLICATIONS

Augmented reality interaction is crucially different in the two tasks we explored. While our data comes from a specific setting, the model allows for speculatively exploring the effects of changing parameters. The proposed model can be used directly to make hypotheses in similar interaction situations. For example, by varying the parameters lens size S and machine lag τ_L, we arrive at the following implications:

- Increasing lens display size S means that the first logarithmic term in equation (5) becomes smaller and the second logarithmic term becomes larger, which results in a shorter physical phase relative to the virtual phase. Since the multiplicative factor associated with the virtual phase is larger, the overall movement time should increase. On the other hand, you cannot decrease S too much, because although it minimizes occlusion, it provides less screen real estate to display information.
- Lag is proportional to movement time, since τ_L is a multiplicative factor for the second logarithmic term in equation (5). It is thus critical that lag is minimized.

It is possible that performance on both task types could be significantly improved with advance cues that help guide movement before the target candidate is on the display. Such cues can relate the location of the target – as in techniques utilizing *halos* [2] – and perhaps its identity, and they can give *overviews* or *maps* of the distribution of targets in the space – as in applications of *focus+context* techniques [1].

CONCLUSION

We analyzed target acquisition with camera phones as magic lenses and as dynamic peephole displays. In the first case, some external visual context is augmented by the device. In the second case the device is spatially tracked, but there is no visual context outside the device's display. We have shown that dynamic peephole pointing can be modeled by Fitts' law. In dynamic peephole pointing the whole interaction is mediated by the device in a uniform way – there are no distinguishable phases as in magic lens pointing. By contrast, even though the magic lens had a shorter machine lag than the dynamic peephole interface, it was not adequately explainable by Fitts' law. Magic lens pointing can be divided into an initial coarse physical pointing phase, in which the target can be directly observed, and a second fine-control virtual pointing phase, in which the target can only be observed through the device. Since the

device introduces some non-zero delay, the characteristics of visual feedback are different in the first and the second phase. In the magic lens setup, this leads to a weak prediction of movement times when basic Fitts' law is used (adjusted $R^2 = 0.54$). To more adequately model the situation of magic lens pointing we introduced a two-part model with three parameters (target width *W*, target distance *D*, and display size *S*) that led to more accurate predictions (adjusted $R^2 = 0.87$). We expect that magic lens interaction will become more popular in the future, since a large range of applications are conceivable if robust camera-based tracking is available for camera-equipped mobile devices.

ACKNOWLEDGMENTS

We would like to thank Tico Ballagas for helpful comments on a draft of this paper. The second author's work was supported by the Academy of Finland project ContextCues.

REFERENCES

1. Baudisch, P., Good, N., Bellotti, V., and Schraedley, P. Keeping things in context: a comparative evaluation of focus plus context screens, overviews, and zooming. In *Proc. CHI 2002*, ACM Press (2002), 259–266.
2. Baudisch, P. and Rosenholtz, R. Halo: A technique for visualizing off-screen objects. In *Proc. CHI 2003*, ACM Press (2003), 481–488.
3. Bier, E. A., Stone, M. C., Pier, K., Buxton, W., and DeRose, T. D. Toolglass and magic lenses: the see-through interface. In *Proc. SIGGRAPH 1993*, ACM Press (1993), 73–80.
4. Card, S. K., Newell, A., and Moran, T. P. *The Psychology of Human-Computer Interaction*. Lawrence Erlbaum Associates, Inc., Mahwah, NJ, USA, 1983.
5. Crossman, E. R. F. W. and Goodeve, P. J. Feedback control of hand-movement and Fitts' law. *Quarterly J. Experiment. Psych.* 35A (1983/1963), 251–278.
6. Fitts, P. M. The information capacity of the human motor-system in controlling the amplitude of movement. *J. Exp. Psychol.* 47 (1954), 381–391.
7. Fitzmaurice, G. W., Zhai, S., and Chignell, M. H. Virtual reality for palmtop computers. *ACM Trans. Inf. Syst. 11*, 3 (1993), 197–218.
8. Graham, E. D. and MacKenzie, C. L. Physical versus virtual pointing. *In Proc. CHI 1996*, ACM Press (1996), 292–299.
9. Grossman, T. and Balakrishnan, R. Pointing at trivariate targets in 3D environments. *In Proc. CHI 2004*, ACM Press (2004), 447–454.
10. Guiard, Y. and Beaudouin-Lafon, M. Preface: Fitts' law 50 years later: Applications and contributions from human-computer interaction. *Int. J. Hum.-Comput. Stud.* 61, 6 (2004), 747–750.
11. Guiard, Y., and Beaudouin-Lafon, M. Target acquisition in multiscale electronic worlds. *Int. J. Hum.-Comput. Stud.* 61, 6 (2004), 875–905.
12. Hecht, B., Rohs, M., Schöning, J., and Krüger, A. Wikeye – Using magic lenses to explore georeferenced Wikipedia content. In *Proc. 3rd Intl. Workshop on Pervasive Mobile Interaction Devices (PERMID)*, 2007.
13. International Organization for Standardization. *Ergonomic requirements for office work with visual display terminals (VDTs) – Requirements for non-keyboard input devices. ISO/IEC 9241-9*, 2000.
14. MacKenzie, I. S. Fitts' law as a research and design tool in human-computer interaction. *Human-Computer Interaction* 7, 1 (1992), 91–139.
15. Mehra, S., Werkhoven, P., and Worring, M. Navigating on handheld displays: Dynamic versus static peephole navigation. *ACM Trans. Comput.-Hum. Interact.* 13, 4 (2006), 448–457.
16. Murata, A., and Iwase, H. Extending Fitts' law to a three-dimensional pointing task. *Human Movement Science* 20, 6 (2001), 791–805.
17. Rohs, M. Marker-based embodied interaction for handheld augmented reality games. *Journal of Virtual Reality and Broadcasting* 4, 5 (2007).
18. Rohs, M., Schöning, J., Krüger, A., and Hecht, B. Towards real-time markerless tracking of magic lenses on paper maps. *Advances in Pervasive Computing. Austrian Computer Society* (2007), 69–72.
19. Sears, A. and Shneiderman, B. High precision touchscreens: Design strategies and comparisons with a mouse. *Int. J. Man-Mach. Stud. 34*, 4 (1991), 593–613.
20. Silfverberg, M., MacKenzie, I. S., and Kauppinen, T. An isometric joystick as a pointing device for handheld information terminals. In *Proc. Graphics Interface 2001*, Canadian Information Processing Society (2001), 119–126.
21. Soukoreff, R. W. and MacKenzie, I. S. Towards a standard for pointing device evaluation, perspectives on 27 years of Fitts' law research in HCI. *Int. J. Hum.-Comput. Stud.* 61, 6 (2004), 751–789.
22. Wang, J., Zhai, S., and Canny, J. Camera phone based motion sensing: interaction techniques, applications and performance study. In *Proc. UIST 2006*, ACM Press (2006), 101–110.
23. Ware, C. and Balakrishnan, R. Reaching for objects in VR displays: lag and frame rate. *ACM Trans. Comput.-Hum. Interact.* 1, 4 (1994), 331–356.
24. Yee, K.-P. Peephole displays: Pen interaction on spatially aware handheld computers. In *Proc. CHI 2003*, ACM Press (2003), 1–8.