

# View Size and Pointing Difficulty in Multi-Scale Navigation

Yves Guiard<sup>1</sup> Michel Beaudouin-Lafon<sup>2</sup> Julien Bastin<sup>1</sup> Dennis Pasveer<sup>2</sup> Shumin Zhai<sup>3</sup>  
<sup>1</sup> LMP  
CNRS & Univ. Méditerranée  
Marseille, France  
{guiard,bastin}@laps.univ-mrs.fr  
<sup>2</sup> LRI & INRIA Futurs  
Université Paris-Sud,  
Orsay, France  
mbl@lri.fr  
<sup>3</sup> IBM Almaden Research Center  
San Jose, CA, USA  
zhai@us.ibm.com

## ABSTRACT

Using a new taxonomy of pointing tasks which includes view pointing beside traditional cursor pointing, we introduce the concept of multi-scale pointing. Analyzing the impact of view size, we demonstrate theoretically and experimentally that (1) the time needed to reach a remotely located target in a multi-scale interface still obeys Fitts' law and (2) the bandwidth of the interaction (i.e., the inverse of Fitts' law slope) is proportional to view size, a relationship bounded by an early ceiling effect. We discuss these results with special reference to navigation in miniaturized and enlarged interfaces.

## Categories and Subject Descriptors

ACM Classification Keywords:

H.1.2 [Models and Principles]: User/Machine Systems - Human information processing.

H.5.2 [Information Interfaces and Presentation (e.g., HCI)]: User Interfaces - Graphical user interfaces, Input devices and strategies, Interaction styles, Theory and methods.

General Terms: Human Factors, Theory, Experimentation.

## Keywords

Input and Interaction Technologies. Analysis Methods. Empirical Methods. Fitts' law. Quantitative.

## 1. INTRODUCTION

Visual displays have been used for decades in human-computer interaction (HCI). Screens first served to display text and simple forms, but with the advent of graphical user interfaces (GUIs) [19] computer screens turned into analog devices that produce bitmap images. In this paper, we consider the case in which the depicted scene consists of a very large, two-dimensional (2D) flat document that appears behind the screen, parallel to it, and in which the user can navigate by zooming and panning.

The size of documents computers are able to handle has increased considerably over the last two decades. It is quite common today, even with an inexpensive desktop computer, to have access to huge

documents like a complete world atlas. Yet, the size of our screens has not increased, far from it. Screens have remained about the same size, but modern GUIs encourage the opening of multiple windows, at the cost of actually reduced views. Also, research on mobile computing has produced a whole family of devices, like PDAs or cell phones, that offer drastically reduced views. So view size has actually tended to shrink, leading to an increasing mismatch between the size of the information worlds users interact with and the size of the views they enjoy into these worlds.

In the present study we resort to the conceptual framework of Fitts' pointing paradigm [6,15] to tackle the problem of view size in the specific context of multi-scale navigation. One central assumption of our analysis is that to visualize some target region of a large document is tantamount to *pointing with one's view*. In the next two sections, we proceed as follows. We first explain how Fitts' paradigm may serve to model not only cursor pointing, but also view pointing. We then introduce the dimension of scale, generalizing the paradigm to account for multi-scale view pointing.

### 1.1. Cursor Pointing and View Pointing in Fixed-scale Interfaces

Here we consider a generalized pointing task where the target need not be visible in the current view. Acquiring a target that is out of view requires first moving the view so as to make the target visible, then pointing at the target. We call the former *view pointing* and the latter *cursor pointing*. We start in the present section with the case of a view that can be moved (or scrolled) in the absence of a zooming facility.

#### 1.1.1. Nesting and Precedence

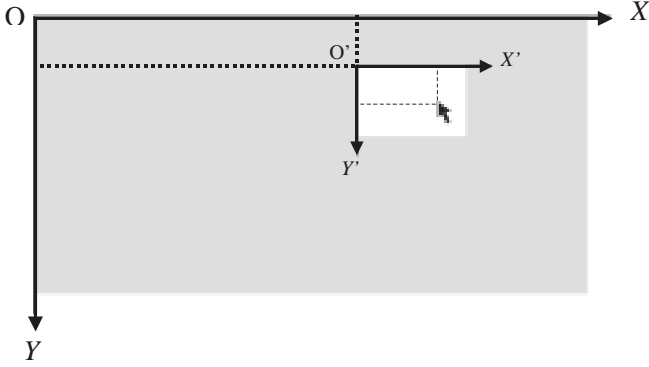
First, note that cursor pointing is *nested* within view pointing, as illustrated in Figure 1. Whereas cursor pointing takes place in the local space of the view, view pointing takes place at the global level of the document. Strictly speaking, cursor position can be defined either locally, relative to the view (dashed lines in the figure), or globally, relative to the document. But view position can only be defined relative to the global coordinate system XOY of the document (bold dotted lines in the figure). The nesting relationship is clearly reflected by the fact that the view, when it is moved, automatically carries the cursor with it, while the reciprocal is not true.

A typical sequence of action, in keeping with the global precedence principle familiar to the students of human perception [17] and

To appear in : *Proceedings of Advanced Visual Interfaces, AVI 04*, Gallipoli, Italy, May 25-28, 2004. New York: ACM Press

<sup>1</sup> Rooted in Weber's law [5], Fitts' law in essence quantifies the human capability to resolve or control any variable within a set resolution.

action [9], is as follows: the user first moves the view in order to visualize the appropriate region of the document, and then proceeds to cursor pointing to eventually reach some object visible in the new local view.



**Figure 1.** The three objects of interest: the cursor (tilted black arrow), the view (small empty rectangle), and the document (large gray rectangle). Within the global coordinate system of the document (XOY), the view defines a movable, local coordinate system (X'O'Y') for cursor motion.

### 1.1.2. A Generic Concept of Pointing

The task that Fitts [6] used for the demonstration of his well-known law consisted of reaching a specified target with the sharp tip of a stylus. Fitts' law is usually stated as

$$MT = a + b \log_2(D/W + 1) \quad (1)$$

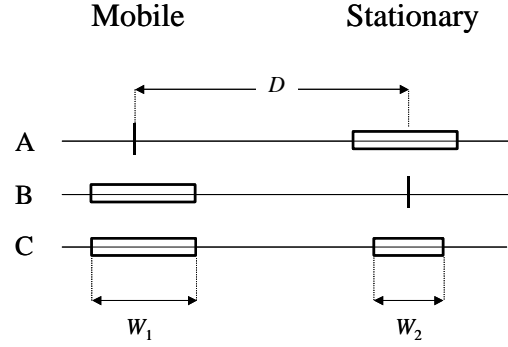
where  $MT$  stands for target acquisition time,  $D$  and  $W$  for target distance and width. The log term defines the task's index of difficulty ( $ID$ ), and  $a$  and  $b$  stand for empirically adjustable coefficients ( $b > 0$ ) [15].

Fitts' law has long been known to apply just as well to the most common HCI version of Fitts' task, cursor pointing, in which the user must move a screen cursor to a graphical object [4], but in this article we also consider view pointing. At first sight, these are two different kinds of a movement task. In fact, as far as the variables of Fitts' pointing paradigm are concerned, there is no essential difference between view pointing and cursor pointing. Figure 2 illustrates, in 1D space, the three possible cases that can be met in a pointing task—ignoring the case in which both the cursor and the target are points, because we want to treat tolerance as a variable. In case A, an essentially extensionless cursor must be moved to reach some target interval. Case B, symmetrical to case A, is that identified by Kabbash and Buxton [14] as 'Prince' pointing, in which some interval (an area cursor in 2D space) must be moved so as to eventually include some point target.

Even though cases A and B seem to be qualitatively different, they just differ by their arbitrary reference frames, as noted by Mottet et al. [16]. Although  $W$  is a measure of the target in one case and of the area cursor in the other, the computation of the  $ID$  is identical.

Case C, in which falls view pointing, does not seem to have been considered so far in Fitts' law literature, basic or applied. In 1D

space, we have an interval that must be moved so as to overlap with another interval.



**Figure 2.** A taxonomy of pointing tasks: cursor pointing (A), 'Prince' pointing (B), and view pointing (C). Note that intervals along the horizontal axis, represented as rectangles, must be thought of as extending only in 1D.

The quantification of task difficulty is just as straightforward in view pointing as it is in the other two cases. Target distance  $D$  is simply the distance separating the centers of the two intervals. Target width  $W$  can be quantified in one of two ways, depending on the goal pursued in pointing. If the user only aims at selecting an object, then an overlap will suffice, and hence  $W$  must be computed as the sum of the width of the mobile interval  $W_1$  and the width of the stationary interval  $W_2$ :

$$ID = \log_2 \left( \frac{D}{W_1 + W_2} + 1 \right) \quad (2)$$

If, however, the user aims at reaching the target to visualize it, as is normally the case with view pointing, then the larger interval must include the smaller interval, and hence  $W$  is the absolute-value difference between  $W_1$  and  $W_2$ :

$$ID = \log_2 \left( \frac{D}{|W_1 - W_2|} + 1 \right) \quad (3)$$

In either case the  $ID$  receives an unequivocal definition. A non-trivial implication of Equations 2 and 3 is that Fitts' law can be assessed in view pointing just as easily as it has been so far in usual cursor pointing. The concept of view pointing, associated with Equations 2 and 3, helps to understand that Fitts' law can serve not only to model classic pointing with a cursor, but also document navigation—obviously a broad class of user activities in HCI.<sup>2</sup>

## 1.2. View Pointing in Multi-Scale Interfaces

When navigating a very vast document over huge distances in comparison with both view size and target size, it is necessary to

<sup>2</sup> In case C, unlike case A or B, the target may remain invisible during a large proportion of the movement. This means that the perceptual-motor control mechanisms involved certainly differ between cases A-B and C. However, this is another issue, separate from Fitts' law.

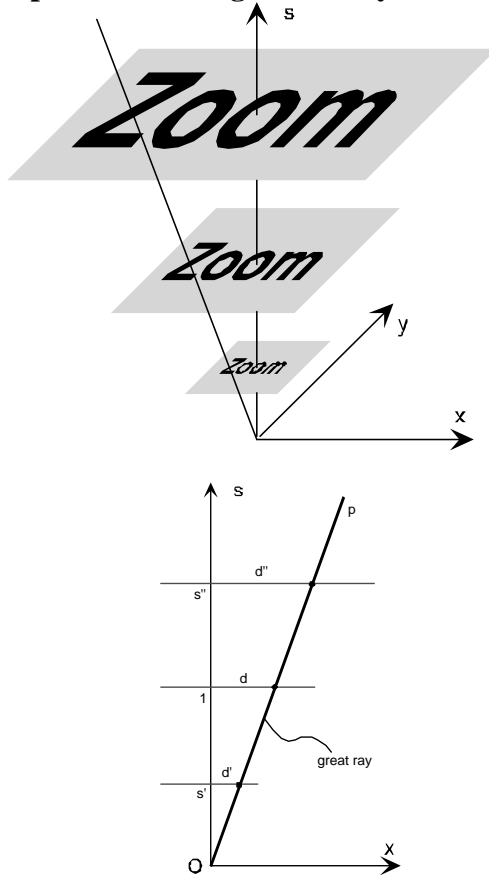
introduce multiple scales. Many techniques have been introduced to support multi-scale interaction, including zoomable user interfaces [18] and fish-eye views [7]. In the simplest case, which is the focus of the present paper, the whole document can be uniformly rescaled so as to give an overview of a larger part of the document or to show greater detail on a smaller subset.

In this situation, view pointing is not properly modeled by equation 3 above, because view size  $W_1$  is constant across scale changes while target size  $W_2$  is scale-dependent. Applying Equation 3 would lead to an  $ID$  that changed with the zoom, a spurious feature. Obviously, in the case of multi-scale interfaces we need to attack view pointing from a different angle.

## 2. MODELING THE EFFECT OF VIEW SIZE ON MULTI-SCALE NAVIGATION TIME

That multi-scale navigation, conceptualized as view pointing, is indeed amenable to Fitts' law has been demonstrated recently [11]. However this empirical work did not rely on a theoretical model. In this section we set to describe such a model.

### 2.1. A Space-Scale Diagram Analysis



**Figure 3. Space-scale diagrams for a 2D document (above) and a 1D document (below). The scaling invariant corresponds to the similarity between triangles:  $d/1 = d'/s' = d''/s''$ .**

In order to view and navigate a document that is far larger than the available view, the user needs to manipulate the dimension of scale ( $s$ ). We limit ourselves to uniform scaling. Viewing the document at scale  $s$  means that all distances are multiplied by  $s$ . If  $s = 1$ , the document is viewed at its 'natural' (actually conventional) size; if  $s > 1$ , the document is magnified (zoomed-in); if  $0 < s < 1$ , the document is 'minified' (zoomed-out). Scaling is captured by the following *scaling invariant*: distance  $d$  at scale  $s$  is identical to distance  $d'$  at scale  $s'$  if and only if

$$d/s = d'/s'. \quad (4)$$

It is convenient to represent multi-scale document navigation with a space-scale diagram [8], which plots scale on its vertical axis (Figure 3). A point of coordinate  $p$  in the document, viewed at scale  $s$ , is represented by the point  $(p.s, s)$  in the diagram. When  $s$  varies, the set of images of point  $p$  in the diagram is a great ray, or semi-line going through the origin.

#### 2.1.1. Viewing and navigation

Viewing the document requires a display window or *view*, typically a rectangle. For a 1D document, the view is an interval of width  $V$ .

Navigating the spatial dimensions is achieved by *panning*. Given an arbitrary origin,  $O$ , for the document, panning changes the distance, in document space, between  $O$  and view center. Note that the user controls panning in display space. Panning a distance  $d$  on the screen when the document is displayed at scale  $s$  actually pans by  $d/s$ .

Navigating the scale dimension is achieved by *zooming*. Zooming-in goes up the scale dimension, while zooming-out goes down (Figure 3). Note that view size does not change with scale: it is constant across the space-scale diagram.

Zooming is non-linear. Rather than increasing or decreasing scale by a constant factor, zooming multiplies or divides the current scale by a constant *zoom factor*,  $f > 1$ :

$$\text{zooming-in: scale } s \text{ becomes } s' = s.f$$

$$\text{zooming-out: scale } s \text{ becomes } s' = s/f$$

In order to measure distances on the scale dimension, we introduce the *zoom index*,  $z = \log s$ . Navigating the scale dimension increases or decreases the zoom index by a constant factor  $z_f = \log f$ :

$$\text{zoom-in: } z' = \log s' = \log s.f = \log s + \log f = z + z_f$$

$$\text{zoom-out: } z' = \log s' = \log s/f = \log s - \log f = z - z_f$$

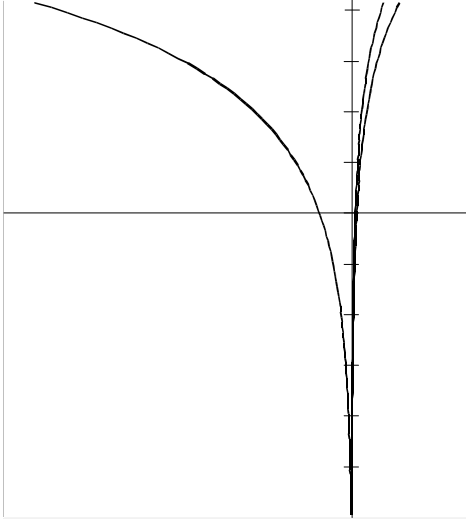
In order to normalize the zoom-index dimension, we use the logarithm in base  $f$  and therefore  $z_f = 1$ .

Controlling the zoom index rather than the scale means that the user can quickly navigate very large distances in scale space. With a zoom step<sup>3</sup> of 0.1, the zoom-index range  $[-50, 50]$  corresponds to a scale range of approximately  $[1/100, 100]$ , i.e. a ratio of 10000 between the smallest and largest scale, while the zoom-index range

<sup>3</sup> The zoom step is simply  $f-1$ . A larger zoom step makes the interface more sensitive to zooming, while a smaller zoom step makes it less sensitive. In practice, a value of 0.1 proves to be a good compromise.

[-100, 100] corresponds to a scale range of approximately [1/15000, 15000], that is, a ratio of 225 million.

If we replace the scale dimension by a zoom-index dimension in the space-scale diagram, the great rays become exponential curves (Figure 4). Such a diagram is more true to the experience of navigating a multi-scale world with zooming and panning, but less easy to work with than the original one [8]. Note that, since the space dimension is not changed, we still have the property that the viewing window has a constant size across the diagram.



**Figure 4.** Space-scale diagram of a multi-scale pointing task with the zoom index shown on the vertical axis.

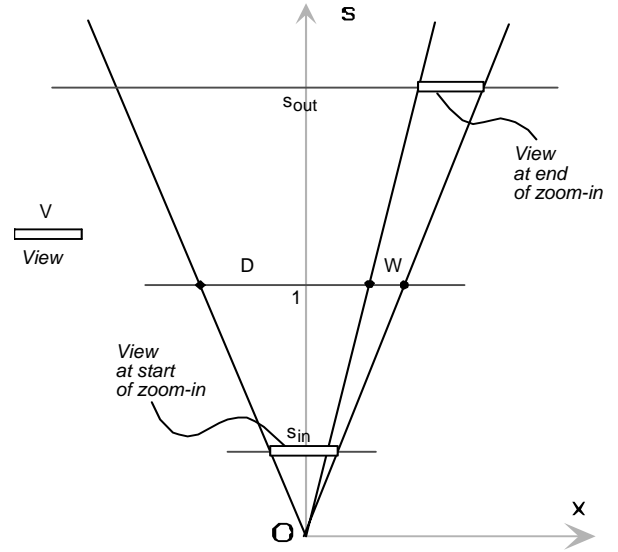
### 2.1.2. View pointing in multi-scale worlds

Target acquisition in a multi-scale interface typically involves three consecutive phases of action: (1) zooming-out, (2) zooming-in-while-panning, and (3) pointing to the target with the cursor [8,11]. Movement time  $MT$  corresponds to the sum of the durations of these phases. Note that phase 2 is the most complex of the three. While phase 1 involves just scale, and phase 3 just space, phase 2 demands an intricate coordination of action in both space and scale. Furthermore, phase 2 is by far that which lasts most in difficult tasks [11]. This is why below we focus on zooming-in, ignoring the initial zooming-out and the terminal cursor-pointing phases.

Consider a view pointing task defined by a view size  $V$ , a distance  $D$  to the target and a target size  $W$  ( $D$  and  $W$  are specified in document space, e.g. at scale 1, and  $V$  is independent of the scale). We are interested in the case where target distance is very large compared with target size, that is, tasks with a high index of difficulty  $ID = \log_2(D/W + 1)$ .

To start with (Phase 1), one must zoom-out until the target (in fact, the semantic ‘blob’ that represents it) enters the view. Let us call  $s_{in}$  the scale at which this happens (Figure 5). Once the target is within the view, the user can pan and zoom-in (Phase 2) towards the target. Navigation ends when the target is big enough in the view so as to be pointed at. Since we are interested in view pointing rather

than the final act of cursor pointing, let us consider the limiting case where navigation ends when the target completely fills the view, calling  $s_{out}$  the scale at which this happens (Figure 5). On the scale dimension, the user has navigated from  $s_{in}$  to  $s_{out}$  (again, we ignore the zoom-out phase to concentrate on zoom-in). In zoom-index space, which is the only space where distances in scale can be computed, the zoom-index distance ( $ZD$ ) is  $\log s_{out} - \log s_{in}$ . Let us compute  $ZD$ .



**Figure 5.** Multi-scale pointing towards a target.

Using the scaling invariant, we first compute  $s_{in}$  and  $s_{out}$ . At scale  $s_{in}$ , the view (size  $V$ ) covers the distance to target,  $D$ , and the target itself (size  $W$ ):

$$(D + W) / 1 = V / s_{in} \text{ or: } s_{in} = V / (D + W)$$

At scale  $s_{out}$ , the view (size  $V$ ) covers the target (size  $W$ ):

$$W / 1 = V / s_{out} \quad \text{or: } s_{out} = V / W$$

We now compute  $ZD$ :

$$\begin{aligned} ZD &= \log s_{out} - \log s_{in} = \log V/W - \log V/(D+W) \\ &= \log (D+W)/W = \log (1 + D/W) = ID \end{aligned}$$

Therefore the distance to navigate in zoom-index space is exactly the index of difficulty of the task. Note that this is independent of view size  $V$ : indeed, a smaller view would require starting at a lower scale, but the task would be completed at a lower scale as well.

### 2.1.3. From the Steering law to Fitts' law

The view pointing task described above is quite reminiscent of a steering task [1]: when the target is in the view, zooming-in to it is similar to driving a vehicle through a tunnel: zooming-in is like operating the throttle of a vehicle as it amplifies the current aiming error; panning is like steering the wheel in order to correct the error. If the target moves out of the view (the user is off the road), the

user must zoom-out (back up) until the target gets back into the view and then zoom-in again. For a successful navigation to occur, the target must stay within the view while zooming-in, exactly as the car must stay inside the tunnel while traversing it. Therefore we can use the expression of Accot and Zhai's steering law [1,21] for tunnels to predict the navigation time:

$$MT = k d / w$$

where  $d$  is the length of the tunnel and  $w$  its width. In our case,  $d$  is the distance navigated along the scale dimension, which we have shown to be equal to the index of difficulty  $ID$ , and  $w$  is the target size of the view, which we call  $V$ . Therefore we have:

$$MT = (k / V) ID \quad (5)$$

Since  $V$  is fixed for a given navigation, this shows that multi-scale view pointing follows Fitts' law. As the motor control involved in multi-scale navigation is quite different from that of a traditional pointing task, this is a nontrivial finding. Fitts' law is known to hold for movements whose speed goes up, then down, without a plateau. However in our case, zooming-in requires a continuous control where panning movements correct the error revealed by the increased scale. When starting the task, the user does not know the size of the target, just its approximate position, thanks to the non-zoomable semantic 'blob' that serves to mark the position of the target until it can be visualized. The user 'cruises' through the tunnel at a fairly constant speed determined by her/his ability to process information. Therefore there is no intuitive reason to expect multi-scale pointing to follow Fitts' law.

This, however, is the case indeed, as has been shown empirically by Guiard et al. [11] in an experiment that involved indices of difficulty up to 30 bits, far beyond anything attempted before. In this study we were able to show that multi-scale navigation time is proportional to the  $ID$ : along with quite accurate fits ( $r^2$  on the order of .99), we found linear regressions with virtually 0 intercepts.

So the above derivation offers a theoretical account for Guiard et al.'s result. But by the same token it delivers a new piece of information, namely, the prediction that navigation time must be inversely proportional to view size. It is on this hypothesis, not tested in [11], where we used a fixed view size in full-screen mode, that we now focus.

An inverse proportionality is a highly non-linear relationship. Equation 5 says that gradually decreasing view size should lengthen  $MT$  in an accelerated way, thus suggesting that  $V$  is a critical factor that needs to be taken into account in modeling multi-scale navigation. Equation 5 has a problem, however: it implies that as  $V$  rises toward infinity,  $MT$  should tend to 0, obviously an implausible prediction. In the next section we handle this difficulty by enriching the model with a supplementary assumption.

## 2.2. Human Limited Capacity for Handling Information Flows: A View Within the View

Equation 5 can be recast as

$$ID/MT \text{ (bits/s)} = k V, \quad (6)$$

simply noting  $1/k$  as  $k$  (since the inverse of a constant is a constant). This equation states quite simply that the bandwidth of navigation

in a zoomable interface is proportional to view size. Such a proportionality makes sense for small views. However, as the view becomes fairly large, we must obviously assume some ceiling constraint to reflect the limited capacity of humans for exploiting the information outflow from a computer—for example, in 2D space one cannot reasonably predict that the navigation bandwidth will be doubled if the view is increased from  $1\text{m}^2$  to  $2\text{m}^2$ .

This human-factor limitation can be construed in terms of a simple fluid-dynamics model where some fluid must cross two pipes mounted in series, with the first pipe modeling the view offered by computer ( $V_c$ ) and the second pipe the maximal view a human can exploit ( $V_h$ ), assumed here to be constant. In this model the flow of liquid that successfully traverses the system is proportional to the cross section of the *smaller* pipe. That is,

$$\text{for } V_c < V_h, \quad ID/MT \text{ (bits/s)} = k V \quad (7a)$$

$$\text{but for } V_c \geq V_h, \quad ID/MT \text{ (bits/s)} = \text{constant} \quad (7b)$$

Below we report two experiments we ran to test Equations 7a and 7b. Our major question concerned the general shape of the relation linking navigation bandwidth to view size, hypothesized to be a strict proportionality combined with a ceiling effect. Experiment 1 explored a broad range of view sizes to try to roughly identify at what level of  $V$  the slope would start to level off. Experiment 2, focused on a lower range of  $V$  values, was aimed at testing the proportionality hypothesis. Hereafter,  $V$  will implicitly stand for  $V_c$ , the view offered by the computer, our main independent variable.

## 3. EXPERIMENT 1: BROAD RANGE OF VIEW SIZES

### 3.1. Methods

#### 3.1.1. Input device

Participants were provided with two input devices, one for each hand. They controlled panning by moving a stylus with their preferred hand on a Wacom USB Ultra-Pad A4 tablet set to the relative mode. Moving the stylus rightward caused the document to move leftward on the screen, so one felt one was moving the view over a stationary document. The display-control gain was constant, with the so-called mouse 'acceleration' function disabled. The vertical coordinates sent by the tablet were ignored by the software, and so the document could be moved only along the left-right dimension.

To control the zoom, participants manipulated with their non-preferred hand the throttle of a Saitek Cyborg 3D joystick (8-bit precision over a  $110^\circ$  course). One magnification level was associated with each angular position of the throttle (zero-order control).

#### 3.1.2. The document, the screen, and the view

The document displayed for navigation was a synthetic map of virtually infinite extension drawn in gray color on a white background. The map exhibited a pattern of concentric circles centered around the target [11], which ensured that the participant was always aware of both target direction (specified by the orientation of the intercepted arcs) and target distance (the inverse

of arc curvature)—this precluded the so-called desert-fog problem [13] often met in multi-scale navigation. As is usually necessary in vast multi-scale worlds, we had recourse to semantic zooming [18]: at low levels of scale, the target was only represented as a ‘blob’, a green, constant-size cross that served to roughly mark target location. At some critical point in zooming-in progression, the target proper, a green, fully zoomable disc, replaced the blob. During zooming-in, the document expanded around the center of the view, with the concentric pattern being periodically replaced by another, more finely grained concentric pattern, thanks to a self-similar fractal generation mechanism.

We used a 17-in. (43.2cm) screen set to a resolution of 1024\*768 pixels. Observation distance was fixed, thanks to a chin rest located at a 60cm (23.6 in.) distance from the screen.

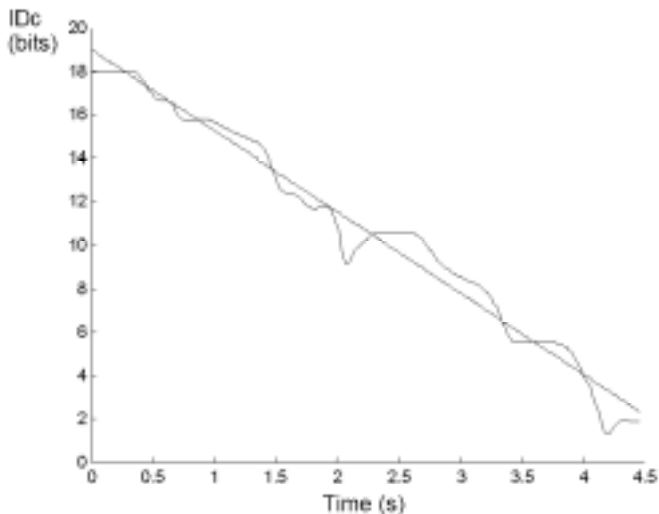
The view appeared in the middle of a black screen as a rectangular aperture showing a selection of the document. View center, coincident with screen center, was permanently marked with a red-colored cross-hair.

### 3.1.3. The Task

Participants were to alternatively reach and click two green discs located at considerable distances on the document (see below). The pattern of concentric circles, always centered around the currently-to-be-reached disc, switched from one target to the other as soon as a hit was recorded.

### 3.1.4. Dependent variables

Since Equation 6, the most telling prediction of our theoretical analysis, links a variable that has the dimension of an information flow (bits/s) to view size, we resolved to directly measure that flow, rather than  $MT$ , and to treat it as our central dependent variable.



**Figure 6.** Evolution of the current level of  $ID$  over a representative target-reaching movement. The straight line shows the linear equation of best fit ( $r^2=.96$  in this example).

To this end, we resorted to a simple linear regression analysis (Figure 6). The first step was to assess, for each sample of each stylus-position time series, the current  $ID$  level, noted  $ID_t$  and defined as  $\log_2(D_t/W+1)$  where  $D_t$ , in document space, stands for the remaining distance to the target at time  $t$ . Since  $D$ , but not  $W$ , is gradually reduced during task progression, the ratio  $D/W$  gradually drops, and so does the current  $ID$ . As visible in Figure 6, which shows the evolution of  $ID_t$  over one representative instance of a target-reaching movement from one participant, the reduction rate of  $ID_t$  over time oscillates around a fairly stable value. Then a linear regression analysis over the whole movement—save the short initial zoom-out phase, during which  $ID_t$  remains constant—suffices to obtain an estimate of the mean slope. It is this slope that we took as our measure of the characteristic information flow, in bits/s, for each individual movement. It must be noted that we always obtained excellent linear fits,  $r$  squares below .9 being quite exceptional.

### 3.1.5. Independent variables and procedure

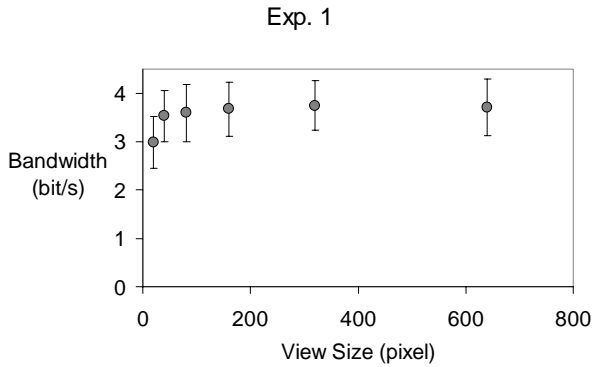
There were two independent variables: the  $ID$  and  $V$ . We used two levels of difficulty,  $ID=14.6$  and  $17.9$  bits—just notice these are pretty high levels: a 17.9-bit pointing task is one with, say, a 1-cm large target and a 2.5-km distance. However, since the only effect of the (initial)  $ID$  in our data-reduction strategy is to provide a longer time series for the evaluation of the bandwidth, as can be understood from Figure 6, below we leave this factor aside.

Concerning  $V$ , it must be noted that one cannot selectively manipulate one dimension of a rectangle without altering its aspect ratio. Since view shape per se might have influenced navigation performance, to avoid a confound we resolved to vary view size in 2D, keeping a constant 1.33 aspect ratio. We explored six levels of  $V$ , spread over a large range. Defining  $V$  as the view’s half-diagonal, these were 20, 40, 80, 160, 320, and 640 pixels (full screen).

Nine unpaid volunteers with normal or corrected-to-normal vision (two female, seven male) participated in the experiment. Each performed 30 movements for each of the 12 cells of the design (2  $ID$ s x 6  $V$ s), leading to a total of 360 movement recorded in a single session, which lasted about an hour. For each participant, order effects were counterbalanced over conditions using Latin squares.

## 3.2. Results and Discussion

As visible in Figure 7, the information flow was essentially constant over the broad range of view sizes selected for Experiment 1, save its extreme lower end. Although the effect of  $V$  was globally significant ( $F(5,35)=17.32$ ,  $p<.05$ ), only the leftmost data point (for  $V = 20$  pixels) significantly differed from others ( $p<.05$ , Newman-Keuls post-hoc test). The form of the mean curve shown in Figure 7 suffered surprisingly little between-individual variability: while the bandwidth systematically increased as  $V$  was raised from 20 to 40 pixels (this effect being present in all nine participants), in none of them was there any notable variation of bandwidth over the higher range of view sizes. So the critical values of  $V$  appeared to be situated in a lower range of view sizes than expected, below 40 pixels.

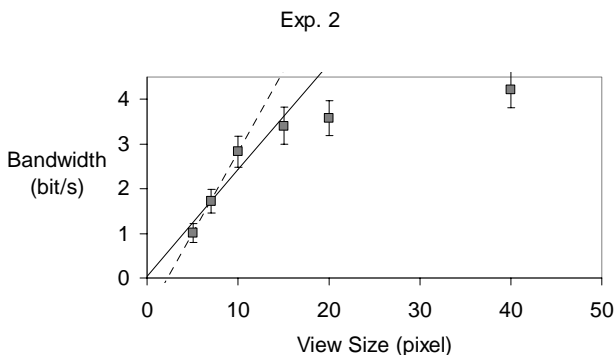


**Figure 7. Bandwidth for a broad range of view sizes. Error bars, represent  $\alpha=.05$  confidence limits based on between-participant variability. The y variable is the inverse of the slope of the  $ID_t$  vs. time function of Figure 6.**

## 4. EXPERIMENT 2: LOWER RANGE OF VIEW SIZES

The aim of Experiment 2 was to check if the variations of  $V$  below 40 pixels does indeed influence bandwidth and, if yes, to evaluate if the proportionality relation stated by Equation 7a holds. The task, the levels of  $ID$  (14 and 17 bits), and the design were the same as in Experiment 1. The only change was that the levels of  $V$  were now 5, 7, 10, 15, 20, and 40 pixels. This experiment involved six new unpaid male adult volunteers with normal or corrected-to-normal vision.

### 4.1 Results and Discussion



**Figure 8. Bandwidth for a lower selection of view sizes.**

The results, illustrated in Figure 8, confirmed our suspicions from Experiment 1. Not only did the bandwidth decrease as  $V$  was reduced below 20-40 pixels ( $F(5,25)=185.14$ ,  $p<.05$ ) but the shape of the curve was consistent with our hypothesis that the relationship starts as a proportionality. Evaluated over the leftmost three and four points of the plot, the  $r^2$  for fitting a straight line was fairly high in all six participants (mean $\pm$ sd = .988 $\pm$ .009 and .913 $\pm$ .046, respectively) and the intercept small enough to give credit to the

proportionality hypothesis ( $-0.82\pm 0.32$  and  $0.03\pm 0.34$  bit/s, respectively).

## 5. IMPLICATIONS FOR HCI

More work is needed to test the proportionality hypothesis of Equation 7a with a finer experimental resolution. However, we believe the above data already provide reasonably firm evidence for the statement that in multi-scale view pointing (1) the bandwidth of the interaction varies proportionally with view size up to a certain critical point and (2) beyond this critical point, the effect vanishes.

The first result amounts to an interaction. Namely, for smaller displays, the effect of task difficulty (i.e., the slope of Fitts' law) depends on view size: the smaller the view, the steeper the slope of Fitts' law. One obvious implication of this interaction is that display miniaturization has a cost in terms of navigation time. What we showed with our mathematical model and our simplified laboratory paradigm raises a concern for the design of dramatically miniaturized (e.g., wrist-watch) interfaces [22]. If the bandwidth is proportional to view size, then navigation time should increase in an accelerated fashion as the view is made smaller and smaller.

However, the question whether a navigation task like ours should actually take longer with a device the size of a PDA or a cell phone rather than with a comfortably sized desktop computer remains open. Recall that the ceiling effect we observed took place rather early in the curve, i.e. at a point where the view's half-diagonal had hardly reached 50 pixels—this corresponding to a view of 80x60 pixels. This finding presumably reflects the strong non-linearity of the impact exerted on target approach by the panning error. Since the next magnification steps will each multiply all visualized distances by a constant zoom factor, the current error will tend to explode, making it rather risky for the user to tolerate too large deviations from the 'blob'. It would seem that the pixels that really count for controlling the zooming-in approach in a multi-scale interface are those few that surround the blob. Apparently this is good news for the design of small, portable computing devices like PDAs.

The second result of this study, the ceiling effect, leads to the strong suggestion that display magnification far above the standard size [12] should exert no notable facilitation effect on the bandwidth of multi-scale view pointing.

If, as demonstrated in this study, view miniaturization dramatically enhances the impact of task difficulty on multi-scale navigation performance, then new solutions need to be designed to assist pointing in the specific context of miniature multi-scale interfaces. Over the last few years a number of Fitts' law based techniques aimed at artificially facilitating pointing have been described in the HCI literature [e.g., 3,10]. Many of these solutions are quite effective, but they apply only to fixed-scale GUIs. The present theoretical framework might offer guidelines to design new pointing-facilitation techniques adapted to the case of multi-scale interfaces in general, with special reference to miniaturized devices.

## 6. CONCLUSIONS AND FUTURE WORK

We feel that the main potential benefits of the present study for HCI research are conceptual. The present work, whose main features are a more general taxonomy of pointing tasks and a principled account

of the role played by the scale variable in target acquisition tasks, is a preliminary attempt at renewing Fitts' pointing paradigm in the direction we have outlined in [11]. Admittedly, Fitts' classic paradigm has proved remarkably useful to HCI research [4,15], but it has served so far to model cursor pointing performance only in the context of *fixed-scale* WIMP interfaces. Given the increasing importance of multi-scale viewing in the current HCI technology, we believe sustained research efforts to quantitatively model multi-scale navigation are now urgently needed.

The present theorizing calls for research developments in several directions. Let us just mention the need to consider the case where the variation of scale is non-uniform. In this paper we have assumed the visualization scale to vary as a whole in the view. However, other treatments of the scale variable, in particular to accommodate the case of fisheye views, will have to be included in future work.

To take a specific example of a possible application, the present theoretical framework might help evaluate quantitatively the cost of interface miniaturization. Our framework suggests a method of quantifying that cost based on the first derivative of navigation time with respect to view size in Equation 5. The cost being expressed in terms of the time added when one more pixel is suppressed, it should be possible, for a given multi-scale task, to identify a *useful* range of view sizes: this range is bounded, on its lower end, by the excessive cost of still miniaturizing the display and, on its higher end, by the immaterial benefit of still enlarging it.

## REFERENCES

1. Accot, J. and Zhai, S. (1997). Beyond Fitts' law: Models for trajectory-based HCI tasks. *Proc. of ACM CHI'97* (pp. 295-302).
2. Bier, E., Stone, M., Pier, K., Buxton, W., & De Rose, T. (1993). Toolglass and magic lenses : the see-through interface. *Proc. of SIGGRAPH* (pp.73-80).
3. Blanch, R., Guiard, Y., & Beaudouin-Lafon, M. (in press). Semantic pointing: Improving target acquisition with control-display ratio adaptation. *Proceedings of ACM CHI'04*.
4. Card, S.K., English, W.K., and Burr, B.J. (1978). Evaluation of mouse, rate-controlled isometric joystick, step-keys, and text keys for text selection on a CRT. *Ergonomics*, 21, 301-613.
5. Falmagne, J.Cl. and Iverson, G. (1979). Conjoint Weber's Law and Additivity. *Journal of Mathematical Psychology*, 20, 164-183.
6. Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, 47, 381-391.
7. Furnas, G.W. (1986). Generalized fisheye views. *Proc. of ACM CHI '86* (pp. 16-23).
8. Furnas, G.W. & Bederson, B.B. (1995). Space-scale diagrams: Understanding multiscale interfaces. *Proc. of ACM CHI'95* (pp. 234-241).
9. Guiard, Y. (1987). Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. *Journal of Motor Behavior*, 19, 486-517.
10. Guiard, Y., Blanch, R., & Beaudouin-Lafon, M. (in press). Object vs. pixel selection in graphical user interfaces. *Proc. of Graphics Interface*. London, Ontario, Canada, May 17-19, 2004. New York : ACM Press.
11. Guiard, Y., Bourgeois, F., Mottet, D. & Beaudouin-Lafon, M. (2001). Beyond the 10-bit barrier : Fitts' law in multi-scale electronic worlds. *Proc. of IHM-HCI 2001* (pp. 573-587). London : Springer.
12. Hascoët, M. (2003). Throwing models for large displays. *Proc. of HCI'2003, Designing for Society, Vol. 2* (pp. 73-77).
13. Jul, S. & Furnas, G. W. (1998). Critical zones in desert fog: Aids to multiscale navigation. *Proc. of ACM UIST'98* (pp. 97-106). ACM Press.
14. Kabbash, P. & Buxton, W. (1995). The "Prince" technique: Fitts' law and selection using area cursors. *Proc. of ACM CHI'95* (pp. 273-279).
15. MacKenzie, I.S. (1992). Fitts' law as a research and design tool in human-computer interaction. *Human-Computer Interaction*, 7, 91-139.
16. Mottet, D., Guiard, Y., Bootsma, R. J., & Ferrand, T. (2001). Two-handed performance of a rhythmical Fitts task by individuals and dyads. *Journal of Experimental Psychology : Human Perception and Performance*, 26, 1275-1286.
17. Navon, D. (1977). Forest before trees: The precedence of global features in visual perception. *Cognitive Psychology*, 9, 353-383.
18. Perlin, K. & Fox, D. (1993). Pad: An alternative approach to the computer interface. *Proc. of ACM SIGGRAPH'93* (pp. 57-64).
19. Shneiderman, B. (1997). *Designing the user interface*. Reading, MA: Addison-Wesley.
20. Zhai, S. (2003). High precision touch screen interaction. *Proc. of ACM CHI'03* (pp. 105-112). ACM Press.
21. Zhai, S., Woltjer, R. (2003). Human movement performance in relation to path constraint: The law of steering in locomotion. *Proc. of IEEE Virtual Reality 2003* (pp. 149-156).
22. Linux on a wrist-watch. <http://www.research.ibm.com/WearableComputing/factsheet.htm>.