RAPPORT DE RECHERCHE

L R I

ALLOWING CAMERA TILTS FOR DOCUMENT NAVIGATION IN STANDARD GUIs : A DISCUSSION AND AN EXPERIMENT

GUIARD Y / CHAPUIS O / DU Y / BEAUDOUIN-LAFON M

Unité Mixte de Recherche 8623
CNRS-Université Paris Sud – LRI

02/2006

Rapport de Recherche N° 1430

CNRS – Université de Paris Sud
Centre d'Orsay
LABORATOIRE DE RECHERCHE EN INFORMATIQUE
Bâtiment 490
91405 ORSAY Cedex (France)

Allowing Camera Tilts for Document Navigation in Standard GUIs: A Discussion and an Experiment

Yves Guiard [a] **Olivier Chapuis** [b] **Yangzhou Du** [b] **Michel Beaudouin-Lafon** [b]

[a] Laboratoire Mouvement et Perception
CNRS & Université de la Méditerranée, Marseille, France
yves.guiard@univmed.fr

[b] LRI & INRIA Futurs
Université Paris-Sud, Orsay, France
{chapuis, du, mbl}@lri.fr

ABSTRACT

We examine the simple idea of allowing users of graphical user interface to view their documents in perspective whenever they need to navigate them. We argue that the virtual camera metaphor involved in GUIs can be likened to a flight simulator where looking in the direction of motion is impossible because the camera points fixedly to the ground. While little seems to have done been so far in HCI research to understand the potentialities and problems of camera tilting for generic document navigation, we claim that flat electronic documents have a future and we list a number of reasons why viewing a 2D document in perspective should help navigation. We analyze the problem of scale implosion in perspective views, and we report the data of an experiment aimed at testing some predictions from our theoretic analyses.

1. INTRODUCTION

This paper discusses a simple idea: Why not allow users of graphical user interfaces (GUIs) to view their document in perspective while navigating it, as illustrated in Figure 1? This idea (at least the simplest version of it that we are considering here) seems to have attracted little attention in HCI research so far, yet we will suggest it has potential for improving the interaction betweens humans and electronic documents. Among the various ways of interacting with a document, one useful distinction is between local work (e.g., reading or editing text) and global navigation (reaching another region of the document): the latter, not the former, is our central concern in the present study.

Figure 1. Perspective view of a linearly arranged document.

2. A FLIGHT SIMULATOR IN THE STANDARD GUI

The virtual camera metaphor involved in computer graphics [10] has been often left implicit in HCI research, as if trivial or non-problematic. In our view, however, it is useful to unearth this component of the GUI architecture to explicitly discuss some of the design options it conceals.

2.1. A Tele-Operated Vehicle with an Onboard Camera

Every GUI—in fact, every window of a multiple-window GUI—can be viewed as a flight simulator. The reason why the user's viewpoint can fly over the document is because the interface can be described by a closed-loop video system made up of a virtual camera mounted onboard a virtual remote-controlled vehicle.

Figure 2 depicts this camera model, in a vertical plane perpendicular to the axis of rotation of the camera, with the camera tilted. The observation point is O and the field of view is the white cone AOC. The horizontal solid line at the bottom represents the document surface. The tilted solid line stands for the projection plane (the computer screen). What we call the *selection*, defined as the document subset that is being visualized, is shown as segment AC. The *view*, defined as the screen subset that is dedicated to the visualization, is shown as segment A'C'. If view size (A'C') is a constant, then the viewing angle (angle AOC) is determined by the camera's focal length, the distance of the projection plane from point O. The cross placed at B' stands for the screen cursor, whose function is to specify, and sometimes to grasp, a point within the document selection (point B); the cursor belongs to the projection space and is constrained to stay within the view A'C'.

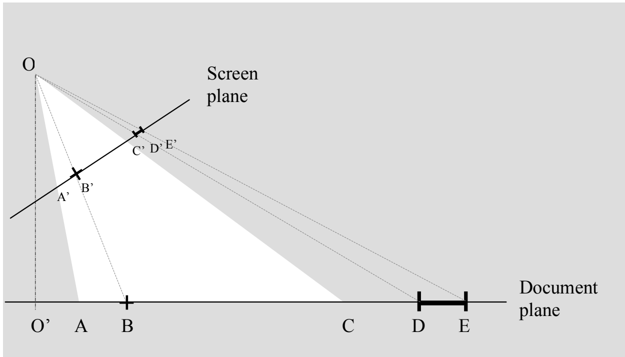

Figure 2. The virtual camera model, with both translations and rotations allowed, reduced to 2D space.

The figure also illustrates a target (segment DE), some text element or some graphical object the user wants to reach. Navigation is needed whenever this target is out of view, as is the case here (segment D'E' is not contained in A'C'). When the distance from the observation point to the distal boundary of the selection (point C of Figure 2) tends to infinity, with the horizon appearing in the view, the *visualization scale*, defined as the D'E'/ DE ratio, tends to zero at that boundary.

Let us describe a few interesting types of rotations that could be implemented in the model.

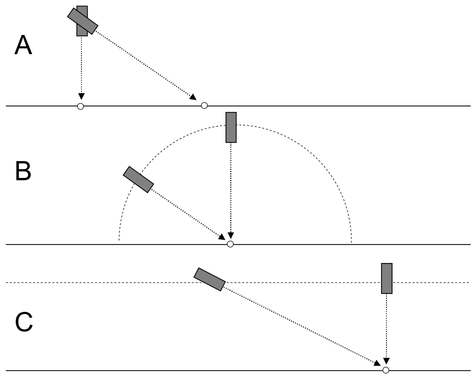

Figure 3. Three kinds of camera rotation, the panoramic rotation (A), the lunar rotation (B), and the trans-rotation (C). The camera's fixation point is shown as a small unfilled circle.

Assuming the user knows the location of the target, an effective way to navigate is to simply tilt the camera until this item enters the view, which we call a *panoramic* rotation (Fig. 3A). But if the user wants to peek at a distant region without losing sight of the local selection, a better option is the *lunar*[1] rotation (Fig. 3B), which consists in revolving the vehicle along a half-circle with the camera being constraining to remain oriented towards its current fixation point in document space. An alternative option consists in translating the camera at a fixed altitude while keeping it aimed at the same fixation point, which we call *trans-rotation* (Fig. 3C). Figure 3 illustrates these three navigation techniques, giving some sense of the design space that camera rotation opens up for document navigation.

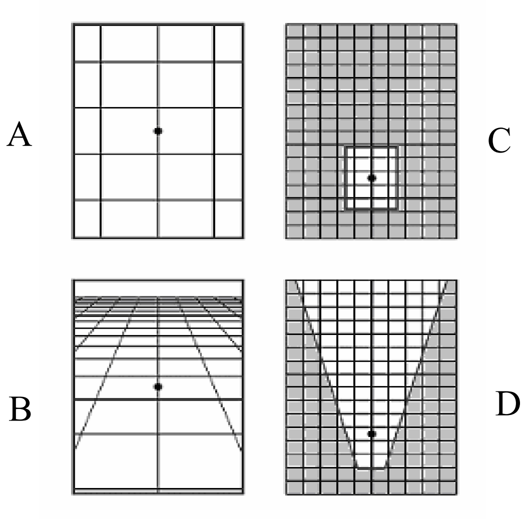

Figure 4. A lunar rotation of the camera shown in view space (from A to B) and in document space (from C to D).

We must distinguish two ways in which the mapping of document space to view space can be thought of, because each space provides a possible reference frame for describing the other. Figure 4 illustrates this by showing the effect of a lunar rotation both ways. Although the view-space display (left) is generally suitable to describe the user experience, it is only in document space (right) that the size of the selection relative to the whole document can be shown. For example, to understand what happens when the user's attention switches to another, currently invisible, region of the document, it helps to be able to describe which part of the document is currently *not* visualized.

2.2. The Scene: A Planar Document

Insofar as available software packages are concerned, we may say that 3D displays like those met in CAD applications are used by minorities of experts. What most computer users handle most of the time are electronic documents that are displayed to them as flat, or *planar* *surfaces* (we refer here to the impressive variety of text displays, spreadsheets, images and photos, web pages, MIDI or audio sequences, musical scores, etc.).

2.3. Two Strategic Variables: The Scene Dimensionality, the Degrees-of-Freedom for Camera Control

Today GUIs involve the metaphor of a camera that can only be translated, remaining invariably oriented perpendicular to the document, and with the document displayed as a planar surface. Starting from this state of the art, one can think of two main research directions for improving the GUI. One is to increase the dimensionality of the visualized *scene* by deploying the document in 3D space. Exploring a 3D layout is difficult with a fixed-orientation camera and so this change brings with it the necessity of camera rotations. Alternatively, we may just introduce one or two rotational degrees-of-freedom (DOF) in camera control, allowing camera tilting. This is a more conservative option in the sense that it can be investigated without questioning the current planar arrangement of documents.

3. PREVIOUS ATTEMPTS TO IMPROVE THE FLIGHT SIMULATOR OF GRAPHICAL INTERFACES

3.1. Laying out the Document Scene in 3D Space

Many attempts have been reported in the HCI literature to enrich the scene of the flight simulator by designing non-coplanar, 3D-space arrangements of the displayed documents, many of which have been concerned with the virtual reality issue [6]. Some authors have relied on the *environmental* metaphor of the room. Among many other realizations, let us mention the Perspective Wall [19], the Task Gallery [21], the Document Lens [22], the Workscape project [3], and the Web Forager [8]. Others prototypes have been based on an *object* metaphor, like 3Book [7], a virtual codex book whose pages, being bound on one side, can be turned about a hinge, affording browsing.

3.2 Allowing Rotational Control of the Camera

Although the utility of tilts has been investigated in the context of small device interfacing where tilting the device may serve to enter input commands without buttons [20], tilting a hand-held device should not be confounded with tilting the virtual camera. Indeed, a camera tilting facility has been implemented in a few *map* browsers (e.g., Google Earth or [20]), but the HCI literature, surprisingly, does not seem to report any study on the possibility of user-controlled camera tilts over flat documents considered as a generic equivalence class.

4. POTENTIALITIES AND CONSTRAINTS OF PERSPECTIVE VISUALIZATION FOR THE NAVIGATION OVER FLAT DOCUMENTS

We start this section with a brief detour through psychology aimed to suggest that, insofar as electronic document navigation is concerned, the current 2D layout of documents is suitable to future improvements of GUIs.

[1] The *lunar* metaphor exploits the property that the moon always faces the Earth from the same angle, revolving about our planet at the same pace as it rotates about itself.

4.1. Why a Planar Layout is Potentially Suitable in the Case of Large Electronic Documents

Consider a person enjoying immobility in an armchair. It would make little sense to show him/her a planar display (e.g., a page of text) whose optical size were very much larger than a computer screen placed at a distance of about 0.5m or an A4-format sheet of paper placed at about half that distance. For both optical and physiological reasons, the *optical array* [13] that can be efficiently scanned by means of gaze and head movements to inspect a planar display from a given point is limited.

The total surface area of most of our documents is too large to be displayed within a single optical array. One solution is *browsing*, as in the case of the familiar book or codex: segments of the document (pages) are displayed sequentially in the optical array, so the observer need not move. The alternative is observer's *self-motion*: the unfolded document being displayed on a large planar surface (e.g., a wall), the observer walks to explore it.

One obvious reason why the codex, an object for manipulation, is a preferable solution for displaying a large amount of visual information is that it dispenses the reader of self-motion — an effortful, slow, and terrestrial process for humans in the real world. In electronic worlds, however, self-motion (i.e., motion of the virtual camera) entails no energy costs whatsoever, it can be as fast as one likes, and it is gravity free. This, we believe, suggests that for electronic worlds documents planarity is *not* a problem. Provided that users have appropriate navigation techniques at their disposal, there is no reason to doubt the navigability of documents that are displayed as huge planar surfaces.

4.2. Geometric Advantages of Perspective Viewing

First and foremost, note that a perspective view (PV) display is inherently multi-scale. As noted by Mackinlay et al. [19], PV allows the visualization scale to vary continuously within the view, making it possible to display *simultaneously* the local detail and the remote context—a feature that obviously makes PV quite attractive for the visualization of large documents. Unlike the zooming technique, based on a time variation of scale, PV allows a whole range of visualization scales to be available at once.

It is also noteworthy that PV offers a convenient variation of visualization scale in view space. Compared with other techniques based on spatial multiplexing such as the bifocal [2] and the fisheye [11] views, PV offers four advantages. First, the scale variation is gradual, rather than abrupt, allowing a reliable, faultless mapping of the document selection to the view. Second, the perspective view rests on a visualization scale variation with distance that is simple, monotonic, and non-arbitrary, being based on the natural laws of ecological optics. Third, the range of scales that can be represented with PV is larger, allowing this technique to compete with the pan and zoom (P&Z) technique, the only one to date known to accommodate any range of scale and therefore to adapt to arbitrarily large documents [14]. Finally, PV is familiar, since after all it is in perspective that we see all the surfaces of the real world.

4.3. Dynamic Advantages of Perspective Viewing

4.3.1. Moving the Selection in the Attended Direction

Document navigation starts when the user decides to look for some document region not currently displayed. Figure 5, which codes the distribution of the user's field of interest along the vertical dimension with shades of grey, shows how the selection changes with a zoom out and a camera tilt.

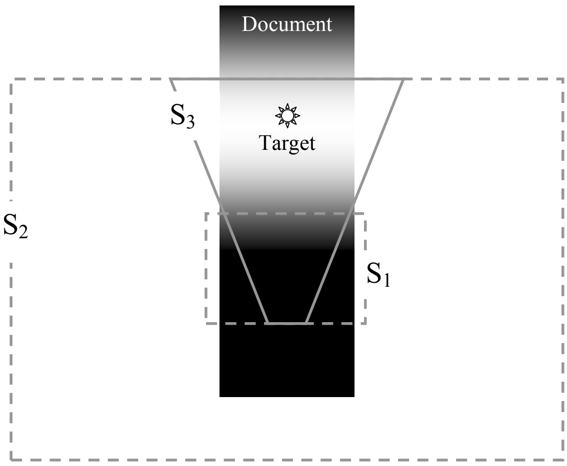

Figure 5. Change of the document selection with the P&Z (from S_1 to S_2) vs. the PV (from S_1 to S_3) technique. The document is shown as a rectangle whose shades of grey represent the distribution of the user's field of attention (lighter meaning more interest).

The user's interest has moved upward a long way away from the current selection (shown as S_1) to an out-of-sight target, so the plan is to get up there. Assuming the interesting part of the document is quite distant from S_1, the user's only option with the state–of-the art GUI is to scale up the selection from S_1 to S_2 by zooming-out. But notice that the change from S_1 to S_2 lacks directional selectivity altogether: S_2 includes all uninteresting (black) document regions; second, most view space serves to visualize irrelevant *void* around the document; finally, the detail of the starting selection S_1 is lost because the visualization scale has collapsed everywhere in the view. In contrast, a camera tilt (here a lunar rotation) in the direction of interest, which reshapes the selection from rectangle S_1 to trapezoid S_3, leads to a selection that excludes most of the uninteresting material and contains little void. Moreover, the initial working spot remains at the same location, with all its initial resolution (if the user was just editing text in S_1, that text is still there, readable and editable).

So a clear advantage of a camera tilt is that, unlike the zoom-out, it allows the user to look *selectively* in the direction of interest before starting the navigation.

4.3.2. Optical-Flow fields with Prospective Information

The best account of the dynamic control of locomotion under visual guidance has been provided by students of ecological optics, along the lines of Gibson [13]. A useful concept is the *optical flow field* induced by self-motion relative to the terrestrial environment. Figure 6, after Gibson [13] (Fig. 7.4, p. 124), illustrates an optical flow field available onboard an aircraft. Each vector illustrates the optical velocity (as measurable on some projection plane) of a grain of optical texture within the optical array available to observer scanning. The figure shows a strongly asymmetrical radial pattern whose focus coincides with the vanishing point on the horizon, uniquely specifying a flight parallel to the ground.

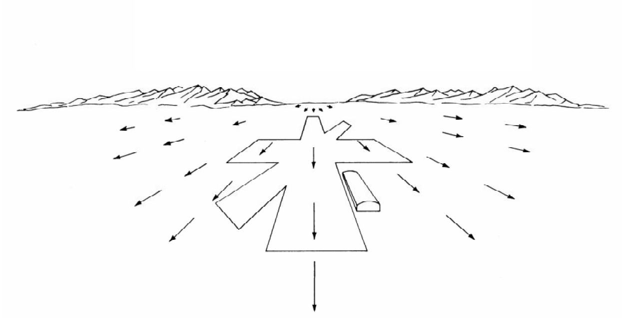

Figure 6. The optical flow field available to an observer who is looking ahead, onboard a plane flying parallel to the ground.

In fact, the pattern of Figure 6 obtains when looking ahead — undoubtedly an obligation for the pilot. This well chosen optical flow field provides information on the immediate future of locomotion, allowing *prospective* control. The pattern in this example is anticipatory evidence that the aircraft is going to cross, at the same altitude, the gap between the two hills that can be spotted near the horizon.

The need for such prospective information in locomotion is reflected by the morphologies of the visual equipments of animals which, by default, point forward during locomotion Likewise, the design of vehicles, whether aircrafts or cars, features windscreens that invariably face to the front.

Unfortunately, the document navigation technology implemented in current GUIs fails to conform to this general design law, as shown in Figure 7. Zooming-in induces a symmetrical radial optical expansion pattern, specifying a rectilinear, perpendicular dive (for zooming out, vector directions reverse). The trouble is that zooming out amounts to nothing but a strategic detour: to see the target by zooming out, observation distance must be actually increased. As for panning, we have a laminar flow field (all vectors the same size and running in parallel), specifying rectilinear motion for an observer looking at the ground through the vehicle's floor. Even though panning allows the virtual camera to progress toward the goal, the relevant *prospective* information is missing for lack of the possibility of looking ahead.

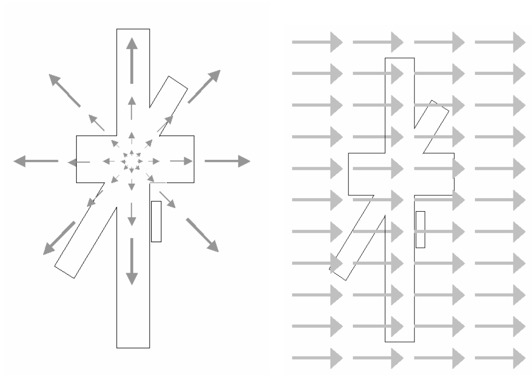

Figure 7 The optical flow fields available to the user of a standard GUI during zooming-in (left) and panning (right).

When users navigate to reach some remotely-located target, they are the pilots. Then, why not offer them a DOF of camera tilt to make it possible for them to look ahead?

4.4 The Main Difficulty with Perspective Visualization: A Scale Implosion Near the Horizon

Perpective visualisation creates a document view with a non-uniform scale. This section shows the problems raised by the fact that the variation of this scale is highly non-linear.

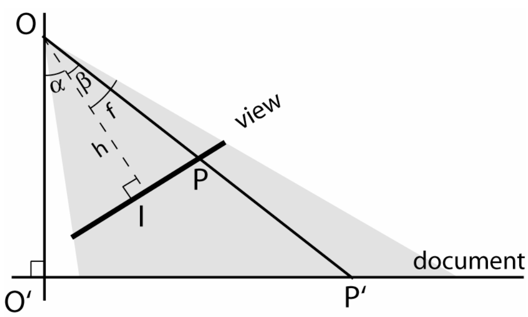

Figure 8 - Perspective visualization.

Figure 8 shows a perspective visualisation defined by a rotation angle α and a view plane at distance OI = h from the camera O. The view itself is defined by its half-size v (not shown in Fig. 8) and its half-angle f, such that $\tan f = v/h$. O' is the orthogonal projection of O on the document plane and is also the origin of the document. When the view angle α is zero, the document is viewed orthogonally, as in traditional GUIs. Without loss of generality, we assume that the height of the camera (distance OO') is equal to h, the distance to the view plane. This means that the document is viewed at uniform scale 1 when the view angle is zero.

When the angle is non zero, a point at coordinate IP = x in the view corresponds to a point at coordinate O'P' = $p(x)$ in the document. Using the triangles OIP and OO'P', we get respectively:

$$\tan \beta = IP/OI = x/h,\ \tan(\alpha+\beta) = O'P'/OO' = p(x)/h$$

Using trignometric calculations we solve for $p(x)$ and get

$$p(x) = \frac{x + h\tan\alpha}{1 - \frac{x}{h}\tan\alpha} \qquad (1)$$

Deriving $p(x)$ gives the inverse of the local scale $S(x)$ of the document at the point of coordinate x in the view:

$$S(x) = \frac{dp(x)}{dx} = 1/\left(\cos\alpha - \frac{x}{h}\sin\alpha\right)^2 \qquad (2)$$

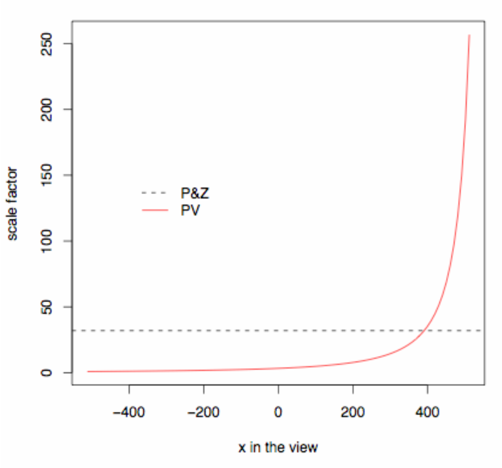

Figure 9 - Scale implosion for perspective visualization, compared with the constant scale of pan-and-zoom for α = 56.90 which allows to see a target at a distance of 16400 pixels (v = 512, f = 30°, ID = 11 in our experiment). Note that the plotted curve is the inverse of the visualization scale.

Plotting $S(x)$ (Figure 9) shows the scale implosion as the x coordinate gets closer to the vanishing point of the perspective for a faraway target. It also shows that the scale is close to 1 in the bottom half of the view ($x = [-v, 0]$).

This means that a click-and-drag technique similar to Adobe Acrobat Reader may be effective to navigate the document: click on a faraway target near the horizon, then drag the cursor downward until the corresponding section of the document reaches the bottom of the view where it has a scale of about one. One problem with this approach is that pixels in the view that are closer and closer to the vanishing point represent larger and larger sections of the document. If the target is located too far, the user may miss the target when clicking and dragging the document through the perspective view. More precisely, if the size of the section of the document "behind" one pixel is larger than half the view size, when the user clicks that pixel and drags down, the corresponding section of the document is enlarged. If the section gets larger than the half the view, the target may get out of the view or may be still too small and the user may miss it, requiring additional click-and-drag actions to reposition the target. This situation occurs when the scale at the top of the view $S(v)$ is larger than half the view size v. We now evaluate the minimum index of difficulty at which this occurs, i.e. such as $S(v) = v / 1 = v$.

For a given position d in the document, let us compute the minimum rotation angle α needed to bring that portion of the document into view, i.e. the rotation angle that brings position d of the document at the top of the view ($x = v$). We solve Equation (1) for α, with $x = v$ and $p(x) = d$:

$$\alpha = \arctan\left(\frac{d - v}{h + dv/h}\right) \qquad (3)$$

Using Equations (2) and (3), we can compute the distance d_{max} such that $S(v) = v$ and therefore, assuming targets of minimal size 8 (as in our experiment), the index of difficulty ID_{max} beyond which targets cannot be reliably selected in one click-and-drag action. For a typical display (v = 512, f = 30°, h = 295), we get α = 57.08°, d = 23169 and ID_{max} = 11.5. This theoretical limit is confirmed by our experimental data: The average number of drags for IDs 9, 11, 13 and 15 were respectively 1.04, 1.11, 1.81 and 2.59 for PV and 1.17, 1.27, 1.64 and 2.28 for P&Z.

Another result of this theoretical analysis is to compare the visualization scale of the target when it enters the view in the PV and P&Z techniques. Figure 10 shows the inverse of the scale factor at the target against the distance to the target. We observe a scale explosion for PV, but a linear growth for P&Z, meaning that PV is likely to work less and less well than P&Z as targets get farther away.

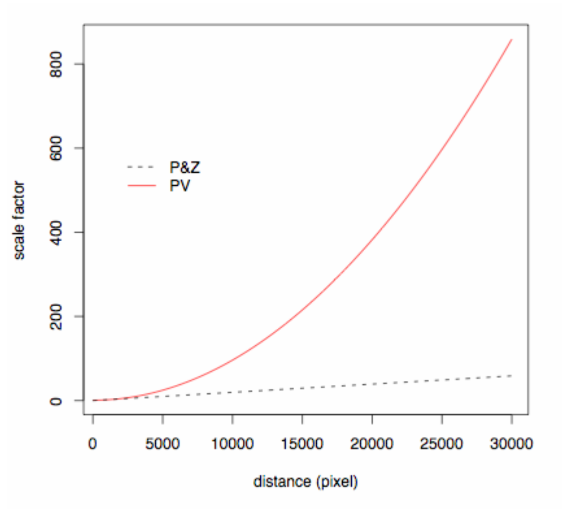

Figure 10 - Scale implosion in PV as the target gets farther away, compared with the linear growth of P&Z.

5. AN EXPERIMENT ON TARGET REACHING WITH PERSPECTIVE-VIEW SCROLLING

This exploratory study did not aim to demonstrate that the new PV navigation technique outperforms the most efficient state-of-the-art techniques—we thought that such an attempt, which demands careful parameter optimization and, to be fair, a reasonable amount of participant practice, was a separate and logically subsequent task. Rather, our main goal here was to set the stage for further studies, by providing a basic understanding of the properties of PV for target-reaching tasks. Because of the scale implosion inherent in PV, our prediction was that target reaching with PV should *not* obey Fitts' law—specifically, one must expect a concave up curvature in the *MT* vs. *ID* function. Our experiment was designed primarily to test this specific hypothesis, and secondarily to provide some sense of how performance compares between PV and the standard P&Z technique

5.1. The Experimental Paradigm: Fitts' Pointing Task

We used Fitts' pointing task, a paradigm [9] that has been abundantly used in HCI research to quantify target-acquisition performance in GUIs [15,18]. Fitts' law links target-acquisition time (or movement time, *MT*) to the ratio of target distance (*D*) and target width (*W*), namely, $MT = k_1 + k_2 * \log_2(D/W + 1)$, where k_1 and k_2 are adjustable coefficients ($k_2 > 0$) and $\log_2(D/W + 1)$ stands for the task *index of difficulty* (*ID*). Recent work has shown that Fitts' paradigm adapts to the case of pointing in a multi-scale interface as well as pointing with one's view rather than a screen cursor [14]. With *D* and *W* measured in document space (O'D and DE in Figure 2), the *ID* measures task difficulty independently of the navigation technique used to zero out distance O'D, so the paradigm is quite suitable for a comparison of target reaching with different navigation techniques.

5.2. Methods

We decided to investigate PV interaction in as simple and as familiar a setting as possible. We adopted the standard wheel mouse and presented our participants with a familiar scrolling technique based on mouse-grasping and dragging the document (e.g., as in Adobe Acrobat Reader). To allow a neat comparison between PV and P&Z document navigation, we designed two task conditions that differed by a single feature, all other things being equal: the wheel served to control either a translation DOF (camera height) in the P&Z condition or a rotation DOF (camera tilt) in the PV condition, thus keeping a constant two input DOF (horizontal mouse-body motion was ignored).

5.2.1. The Equipment, the Document, and the Task

The experiment was run on a 3.4 GHz PC with 512Mo of RAM running Linux and X-Windows, using a 19' monitor with a 1280x1024-pixel resolution driven by a powerful video card. The program, created with C++ and OpenGL, was run in full-screen mode. We used a standard optical wheel-mouse.

The document was an oblong rectangular surface with an aspect ratio of 10 (height/width=1,048,560/104,856 pixels), appearing in white over a dark-grey background. The targets were two constant-size (8-pixels high, 283 pixels wide) blue-filled rectangles situated at varying distances one above the other.

The document was entirely covered with a pattern of black concentric circles centered around the target (the radius of the smallest circle was 141 pixels, with a constant 700 pixels for circle spacing). This background pattern served to provide our participants with ubiquitous information on the direction and distance of the current target and hence to preclude disorientation altogether — as at least one arc was visible from any position at the highest scale [16].

Clicking the target rectangle caused the concentric pattern to be instantly rearranged around the other rectangle. Whenever the target was less than 3 pixels high in the view, the program resorted to semantic zooming [5], replacing the zoomable blue target with a *beacon*, a green 2-pixel thick horizontal line that crossed the document from left to right.

5.2.2. Navigation Techniques

In the P&Z condition, the wheel controlled the height of the camera (always pointed perpendicular to the document), with one wheel notch enlarging/reducing the vertical extension of the documents selection by 10%. In the PV condition, turning the mouse wheel tilted the camera panoramically (pitch axis), with one notch enlarging/reducing document selection size by 10% in the direction of interest. For rapid turns (over 10 notches/s), the wheel effect was 'accelerated' for both techniques according to the formula $\Delta\% = (6–5*\text{notch-time-interval}/100)*10$. Thus, for a 50ms notch time interval in the P&Z condition document magnification was 35%, rather than 10%.

At the beginning of a trial, the target was located out of view. Navigation had to be initiated, using the mouse wheel, by either zooming-out or tilting the camera. Once the target had appeared in the view, whether the latter was perpendicular or oblique, document navigation relied on mouse *dragging*. Depressing the left button of the mouse turned the screen cursor, a red crosshair made up of two 1-pixel thick 17-pixel long segments, into a document-*grasping* instrument. Another fixed crosshair was also present at view center to help navigation in the P&Z (but not the PV) condition.

5.2.3. Pointing Difficulty and Speed-Accuracy Instructions

The P&Z and the PV techniques were tested with four *ID*s, 9, 11, 13 and 15bits — an *ID* of 15bits corresponding to a *D*/*W* ratio of 32,767. The participant was instructed to perform as fast as possible while refraining from committing errors. Any click error had to be corrected at once, hence a constant 0% error rate in our data.

5.2.4. Design

Sixteen adult volunteers participated in two 40-mn sessions, one for each technique, the order being balanced between participants. Each session was divided into trials each of which consisted of 11 clicks made alternatively on the upper and lower target, yielding 10 measures of *MT* at a given *ID* level (the first four measures of *MT* were ignored as warm up). The *ID* was varied pseudo-randomly from trial to trial. Each session included a pseudo-random sequence of 21 trials with *ID* = 9, 11, 13 and 15bits. The first five trials were ignored as warm up, leaving up 4 x 6 = 24 actual measurements of *MT* per level of *ID* and per participant for each condition.

5.3. Results and Discussion

Figure 11 shows for both techniques mean *MT* defined as the time elapsed between two consecutive successful target clicks.

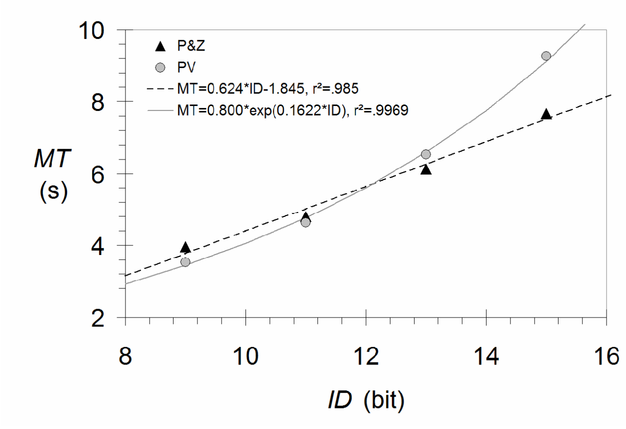

Figure 11. *MT* vs. task difficulty for the two conditions.

For the P&Z condition, in keeping with previously reported findings [16], the *MT* vs. *ID* relationship was essentially linear, with all individual r^2 values in the .915-.999 range. In contrast, notice that the PV curve exhibits a distinctive upward concavity, as predicted. A good fit obtained with an exponential, as visible in the figure, which also reports the coefficients of best fitting. Importantly, the concave-up curvature for the PV condition was quite systematic, being present in all 16 subjects (p<.002, two-tailed sign test). Such a curvature suggests that using this bare, unaided implementation of PV, navigation is likely to fail for much higher levels of *ID* due to unacceptably long *MT*s. To check that this non-linearity was specific to the PV curve, we compared for each technique and each participant the r^2 improvement obtained by taking a log transform of *MT*. Indeed, the r^2 increase was consistently larger for the PV than P&Z technique (t_{15} = 3.02, p = .004, one-tailed).

A two-way repeated measures ANOVA run on a log transform of *MT* with the technique and the *ID* as factors revealed a significant main effect of the *ID* ($F_{3,45}$=840, p>.0001), a significant interaction ($F_{3,45}$=29, p>.0001), but no main effect of the technique ($F_{1,15}$<1). For the most difficult task with *ID* = 15bits, performance was better with the P&Z technique (t_{15} = 5.77, p = .005 with Bonferroni correction), but the reverse was true for the least difficult task with *ID* = 9bits (t_{15} = 3.88, p = .006 with Bonferroni correction). For the two intermediate levels of *ID*, no difference was detected.

6. IMPLICATIONS FOR DESIGN AND FUTURE WORK

While most of our interactions with electronic worlds continue to rest on the same basic GUI design [4], we feel that more conceptual and empirical work is worth dedicating to camera control in 2D document navigation than has been thought so far. This research shows that perspective viewing is indeed viable to navigate relatively large planar documents.

There are many ways in which a GUI might be enriched with camera rotations. For example we can study different assignments of the two DOFs used in our experimental setting (our design excluded parallel control of the DOFs) and extend the types of rotations and rotation-translations that are under user control. Each possible design involves options and parameters that require optimization based on experience and should allow to improve the overall performance of the technique. In a separate contribution [1], we study an automatic coupling of viewing angle with camera tilt and a trans-rotation of the camera. Combining perspective navigation with pan-and-zoom or other techniques is another area for future work.

Another extension to this work is to offer first-order control over the computer-powered flight of the virtual camera, that is, treating the user literally as the pilot of a flight simulator. Specifically, this means designing mouse and/or keyboard commands such that positions of the control translate into velocities of the vehicle that carries the camera. Let us recall again that in everyday tasks users periodically *alternate* between low-scale document navigation and high-scale local work. Therefore, there is no design antinomy whatsoever between allowing people to fly over a document with a perspective view and letting them work on the document with the classic orthogonal view.

REFERENCES

1. Submitted manuscript. Omitted for blind review.
2. Apperley, M. & Spence, I.T.R. 1982. A bifocal display technique for data presentation. *Proc. Eurographics*, 27-43.
3. Ballay, J.M. 1994. Designing Workscape: an interdisciplinary experience. *Proc. CHI'94,* ACM, 10-15.
4. Beaudouin-Lafon, M. 2004. Designing interaction, not interfaces. *Proc. AVI*, 15 – 22.
5. Bederson, B.B., Hollan, J.D. 1994. PAD++: A Zooming graphical interface for exploring alternate interface physics. *Proc. UIST 94*, ACM, 17-26.
6. Bowman, D.A., Kruijff, E., Laviola, J.J., 2004. 3D User Interfaces: Theory and Practice. Addison Wesley Publishing Company.
7. Card, S. K., Hong, L., Mackinlay, J. D., & Chi, E. H. 2004. 3Book: A scalable 3D virtual book. *Ext. Abstracts CHI '04,* ACM 1095-1098.
8. Card, S. K., Robertson, G. G., & York, W. 1996. The WebBook and the Web Forager: an information workspace for the World-Wide Web.

Proc. CHI'96, ACM, 111-ff.

9. Fitts, P.M.A. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *J. Exp. Psychol. 47*, 381-391.
10. Foley, J. D., van Dam, A., Feiner, S. K., & Hughes, J. F. 1990 (2dn Ed.). *Computer graphics: Principles and practice*. Boston, MA, Addison-Wesley Longman.
11. Furnas, G. W. 1986. Generalized fisheye views. *Proc. CHI '86,* ACM, 16-23.
12. Furnas, G. W. & Bederson, B. B. 1995. Space-scale diagrams: Understanding multiscale interfaces. *Proc. CHI'95*, ACM, 234-241.
13. Gibson, J. J. (1979, 1986). *The Ecological Approach to Visual Perception*. Boston: Houghton Mifflin.
14. Guiard, Y. & Beaudouin-Lafon, M., 2004. Target acquisition in multiscale electronic worlds. *Internat. J. Hum.-Comp. Studies 61*(6), 875-905.
15. Guiard, Y. & Beaudouin-Lafon, M. (Eds.), 2004. Fitts' law fifty years later: Applications and contributions from human-computer interaction. Special issue of *Internat. J. Hum.-Comp. Studies 61*(6).
16. Guiard, Y., Bourgeois, F., Mottet, D., & Beaudouin-Lafon, M. 2001. Beyond the 10-bit barrier: Fitts' law in multiscale electronic worlds. *Proc. JHCI'01 and IHM'01.* Springer-Verlag, 573-587.
17. Johnson, J., Roberts, T. L., Verplank, W., Smith, D. C., Irby, C. H., Beard, M., & Mackey, K. 1989. The Xerox Star: A retrospective. *Computer 22*(9), 11-26.
18. MacKenzie, I. S. 1992. Fitts' law as a research and design tool in human-computer interaction. *Human-Computer Interaction 7*, 91-139.
19. Mackinlay, J. D., Robertson, G. G., & Card, S. K. 1991. The perspective wall: detail and context smoothly integrated. *Proc. CHI'91.* ACM, 173-176.
20. Rekimoto, J. 1996. Tilting operations for small screen interfaces. *Proc. UIST'96.* ACM, 167-168.
21. Robertson, G., van Dantzich, M., Robbins, D., Czerwinski, M., Hinckley, K., Risden, K., Thiel, D., & Gorokhovsky, V., 2000. The Task Gallery: a 3D window manager. *Proc. CHI '00*. ACM, 494-501.
22. Robertson, G. G. and Mackinlay, J. D. 1993. The document lens. *Proc UIST '93*. ACM, 101-108.