Assisting Target Acquisition in Perspective Views

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

[a] LRI & INRIA Futurs
Bât 490, Univ. Paris-Sud,
91405 Orsay, France
{du, chapuis, mbl}@lri.fr

[b] Mouvement et Perception
CNRS & Univ. de la Méditerranée,
Marseille, France
yves.guiard@univmed.fr

This article introduces an interaction technique designed to assist target acquisition in flat documents that are visualised in perspective. One reason to allow camera tilts in graphical interfaces is that perspective views provide users with a gradual variation of scale, allowing them to see local detail in the context of a general overview. Our analysis, however, shows that the non-linearity of scale variation in a perspective view jeopardises the acquisition of very remotely located objects. We introduce and experimentally evaluate a solution in which (1) viewing angle is automatically coupled with tilt angle and (2) the tilt is constrained so that the virtual camera stays at a constant altitude and remains pointed to a fixed point on the document. Our results show that with our enhanced perspective navigation technique targets are easy to reach even for extremely high levels of difficulty. Target acquisition time obeys Fitts' law and performance becomes as rapid as with the familiar pan and zoom technique.

Categories and Subject Descriptors: H.5.2 [User Interfaces]: Graphical user interfaces, Input devices and strategies, Interaction styles, Theory and methods.

Keywords: Fitts' law, target acquisition, multiscale pointing, multiscale navigation, perspective view.

BCS-HCI: Engage, September 11-15, 2006, London UK.

1 Introduction

Considerable efforts have been made over the last two decades to devise satisfactory interfaces for viewing exponentially increasing amounts of information through limited screen displays. Multi-scale interfaces now allow users to interact with information objects at different scales [Furnas & Bederson 1995; Guiard & Beaudouin-Lafon 2004]. Notable instances of these multi-scale interfaces are those using the bi-focal [Tzavaras & Spence 1982], the fish-eye [Furnas 1986], and the pan and zoom [Perlin & Fox 1993] technique. The last technique is the most widely used nowadays in the field of interactive information visualisation.

With all the above mentioned techniques, the viewing direction is always oriented perpendicular to the document plane. By contrast, this paper will focus on the case in which the user is allowed to freely tilt the camera relative to the document plane. Our aim is to develop novel multi-scale document-navigation techniques that exploit perspective viewing (PV). Fig. 1 depicts what was seen by participants in the target acquisition experiment with PV to be reported below. Not only was camera tilting permitted, but the user could navigate the document while viewing it in perspective—of interest here is the case of interactive perspective viewing.

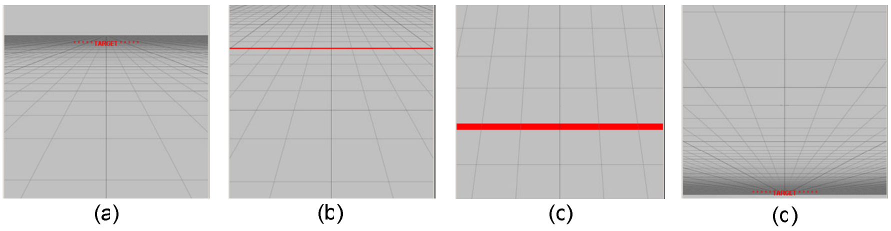

Figure 1: Performing a pointing task in our PV navigation interface. (a) Being at first too far away to be displayed, the target can only be shown as a symbolic beacon. (b) The target is now visible as a 1-pixel thick line but it is still difficult to click. (c) The target is near enough to be clicked. (d) The clicked target has disappeared, and the next target is on the other end of the document, so the user tilts the camera in the opposite direction.

Starting from a local perpendicular view, our participants were to reach and click a very remote target, a narrow red strip located a large distance away in a linearly arranged document but in a known (upward or downward) direction. By tilting the virtual camera upward (Fig. 1a), the target could be made to enter the view but, being at first located too far away, it could be displayed only as a beacon (the text string "TARGET"), which provided a rough indication of the target location, in keeping with the semantic-zoom principle of Bederson & Hollan [1994]. The next step was to move to the target by horizontally translating the tilted camera relative to the document (or, in an equivalent description, by dragging the document relative to the tilted camera) until it became not just visible (Fig. 1b) but large enough for cursor selection (Fig. 1c). As soon as a successful mouse click was recorded by the system, the target jumped to another remote place and the camera had to be tilted in the opposite direction (Fig. 1d)—thus, we used the so-called reciprocal version of Fitts' task [Fitts 1954], with the camera having to be tilted up and down and translated back and forth to perform the task.

A PV interface exhibits the valuable property of presenting the viewer with a smooth transition from the local detail to a global overview that shows remote regions of the document, as emphasised by Mackinlay et al. [1991]. The perspective view obtained by camera tilting involves a continuous gradient of visualisation scale: the farther an item in the document, the lower its visualisation scale. Compared with the standard zooming technique, which conserves a uniform scale all over the view, with scale varying over time, a navigation technique that exploits PV can vary the visualisation scale over space too, with a whole range of scales displayed at the same time in the same view. When this technique is embedded into a visualisation interface, the user is able to look at the global context without losing the current detail.

One noteworthy difference between the tilt-based navigation technique and the familiar zooming technique is that camera tilting is direction-specific whereas zooming is omni-directional. For example, the user will selectively tilt the camera upward to look for some item known to be located in the beginning of the document. By contrast, zooming-out means searching indistinctively in all directions of the document, even when the user knows in what specific region the search should proceed. Since a view has a limited number of pixels, the omni-directionality of the zooming-out operation has a cost. Namely, the document region that was of interest to the user just before the zoom-out inevitably shrinks out like everything else in the view. Unless the user has no idea whatsoever of the target location, a directional search seems preferable. Also, there are cases where losing the contents of the detailed view from which the navigation starts is problematic—for example the final selection of the target may well require reference to some information available in this detailed view. Thus there is reason to believe that PV has a potential for documents visualisation, at least as a complement to the traditional zooming interface.

In a recent work [Guiard et al., in press B], we have pleaded for the PV visualisation of documents and reported a preliminary experimental evaluation of the efficiency of PV for large planar document navigation. Using a pointing task that could be very difficult (with indices of difficulty up to 15 bits, using Fitts' [1954] metric), we found that target acquisition with a simple PV technique was about as efficient as with the standard pan-zoom technique for tasks of moderate levels of difficulty. However, for extremely distant targets (a case to be possibly met in the case of extremely large documents) the PV interface with a bare implementation was no longer workable. We concluded that to handle this limitation of the PV technique, some artificial techniques need to be designed to assist navigation.

Below we present a detailed analysis of the target acquisition problem in a PV interface for very high levels of difficulty and introduce a practical solution. While the view angle (or the field of view) was fixed in our previous implementation of basic PV interface, in the assisting technique we propose it is now allowed to vary and it is actually coupled with tilt angle. In addition, a new kind of camera rotation is exploited, in which the camera travels at a fixed altitude while being constrained to remain oriented to a fixed point in document space. We will report an evaluation experiment showing that with these assisting techniques arbitrarily difficult pointing tasks can be performed in a PV interface, with the users' performance being similar to that obtained in the usual pan-zoom technique.

2 Related Work

In this section we review the main techniques described in the literature that have been devised to assist multi-scale document navigation.

To facilitate the browsing of large document in pan-zoom interface, Igarashi and Hinckley [2000] proposed the Speed-dependent automatic zooming technique. As the scrolling speed increases, the system automatically adjusts the zoom level so that the velocity of the optical flow field keeps constant. To alleviate the focus-targeting difficulty under fisheye-view distortion, Gutwin [2002] presented a technique called speed-coupled flattening, which dynamically reduces the distortion level of a fisheye based on pointer velocity and acceleration. While discussing navigation in 3D, one may think of assisting navigation techniques that are sensitive to objects and the environment. These techniques base their camera control on the 3D world coordinate system and on sensing the surfaces of the objects in the scene. Khan et al.'s [2005] "HoverCam", Zeleznik et al.'s [1999] "UniCam", and Tan et al.'s [2001] navigation system are instances of this category. Arsenault & Ware [2002] have reported a study that helps to understand target acquisition with PV visualisation. They isolate the observer's visual field of view (determined by a screen) from frustum field of view (determined by computer graphics geometry).

Our approach differs from the above in that we address PV navigation in the case of a *planar* surface, a case quite worthy of consideration given the ubiquity of planar documents in current interfaces, whether for desktops or hand-held devices. The assisting technique we discuss below addresses the pointing difficulty raised by scale implosion in PV visualisation. To our knowledge, and to our surprise, there has been virtually no research directly relevant to this topic in the HCI field.

3 The Problem of Grasping Difficulty in Perspective Views

As mentioned earlier, we showed that a bare implementation of the PV concept does not accommodate pointing tasks over some threshold of difficulty [Guiard et al., in press B]. Let us analyze this phenomenon in some detail. We are considering the case of a user allowed to click and drag a document with the screen cursor like, say, in Adobe Acrobat Reader. The new feature is that the drag operation can now be done on a document that is visualized in perspective. An important property of this setting is that the scrolling speed of the document, in its plane, now varies in a highly non-linear way depending on the location of the point that has been grasped. If the cursor has grasped the document at some near location, then very small scrolls will be obtained. However, given the non-linearity of the function, the same drag movement performed at farther locations in the view (hence involving another level of visualisation scale) will have an effect on document scrolling that may differ by several orders of magnitudes.

Suppose the user tilts the camera until a very remote target item enters the view, and the beacon that land marks this item appears near the horizon, toward the edge of the document surface. If the user places the screen cursor slightly above the beacon to drag the target to her/him, the target (in fact along with the whole document) will travel at a tremendous speed and end at a huge distance on the opposite side of the observation point, meaning that the user will lose the target. Alternatively, if the user grasps the document with the screen cursor slightly below the beacon, the target beacon may squarely refuse to move at all. Last but not least, if the cursor grasps the documents exactly at the screen pixel that is land-marked by the beacon, a gentle drag may cause an undetermined effect between the previous two. The reason is because visualisation scale in a PV interface varies in a highly non-linear way across the screen, leading to a scale implosion at some critical distance. One screen pixel in a PV display corresponds to a small document length for near regions and obviously to much larger document lengths for farther regions. If the screen pixel falls near the horizon, the corresponding length in document space may be huge, even infinite. As the resolution of document grasping for mouse-dragging cannot be finer than one screen pixel, the nonlinear variation of view-document mapping causes uncontrollable errors, as illustrated in Fig. 2.

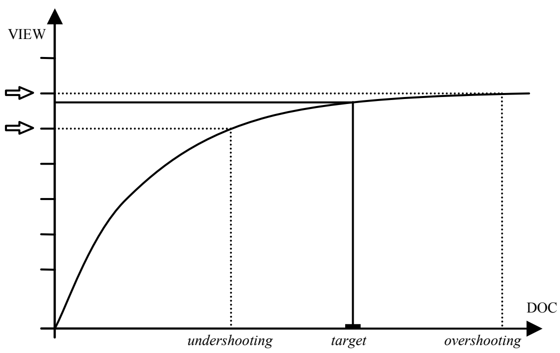

Figure 2: A 1D illustration of the grasping error problem for very remote targets in PV (the dimension considered is the line covered by the camera focus as the user tilts the camera. The view-document mapping actually follows a function of the form $y = 1/x$.

While discussing the overshooting problem in 3D space, it seems timely to recall the Point-of-interest navigation technique introduced by Mackinlay et al. [1990]. The key idea of these authors is to approach a point of interest (i.e., the target) logarithmically, by moving the same relative percentage of distance to the target on every animation cycle. This strategy, however, does not seem to be applicable in the case of a document viewed in perspective. Because of the highly nonlinear property of PV visualisation, the initial aiming point may be far away from the real target and the target can be definitively lost in the very first animation cycle.

This problem, which we call the PV grasping problem, becomes particularly tricky when the index of difficulty (*ID*) exceeds 15 bits, using Fitts' metric (see Appendix 1 for a detailed mathematical explanation of this critical value). Due to the grasping error, the whole target-acquisition procedure becomes quite difficult or takes unexpectedly long time while dealing with tasks of higher levels of difficulty.

The definition of the *ID*, which measures the level of difficulty in Fitts' target-reaching paradigm [MacKenzie 1992], should be made explicit here:

$$ID = \log_2\left(\frac{D}{W}+1\right) \qquad (1)$$

where *D* and *W* stand for target distance and target width, respectively. Fitts' law states the empirical fact that in general the minimum time required for target-acquisition, or movement time (*MT*), varies linearly with the *ID*:

$$MT = a + b * ID. \qquad (2)$$

The coefficients of this linear function, the intercept *a* and the slope *b*, can be used to quantify users' performance with a given interface. This allows us to run rigorous experiments to evaluate the usability of our new designs relative to the state of the art.

Please note that in this paper the *ID* is always calculated in the document plane, rather than in the view plane. This allows us to estimate the difficulty of the navigation task regardless of whether or not the target can be geometrically represented on the screen. As long as the target corresponds to less than one screen pixel, calling for a symbolic representation (the beacon), the difficulty of reaching it cannot be estimated in the view plane.

4 A Formal Analysis and a Solution

As already mentioned, the special difficulty of performing very high *ID* tasks with PV comes from the grasping error in the vicinity of the horizon. This section formally addresses the problem and then presents our solution. Note that we treat the problem in 1D space, along the line covered on the document by the camera focus as the user tilts the camera. The non-linearity concerns the longitudinal dimension of PV (in Fig. 1, for example, only the vertical height, not the horizontal width, of grid squares decreases non-linearly with observation distance). First, let us define the concept of "Visualisation Scale" (S_v) as the ratio of object size in the documents and its projection size on the screen:

$$S_v = \frac{\delta_{doc}}{\delta_{prj}} \qquad (3)$$

where δ_{doc} denotes the original size of the object on document plane and δ_{prj} denotes the size of its projection. Whereas S_v is a constant all over the projection plane in the perpendicular view of a usual zooming interface, it varies dramatically in PV. Fig. 3 shows the pan-zoom situation where the camera is oriented perpendicular to the document plane. A number of camera parameters are introduced here,

H camera height,

h	focal length,
α	field of view, or viewing angle,
D_c	covering distance.

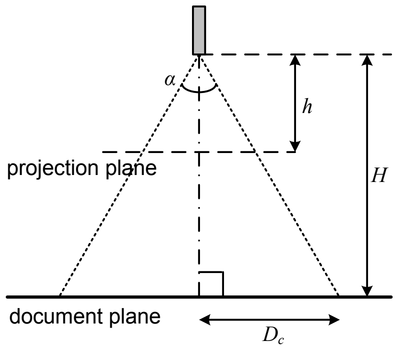

Figure 3: The virtual camera setup in the case of the classic zooming interface.

Note that D_c is a measure of how far the camera can see in the document, it is defined in the document plane from the camera location to the boundary point. Since the axis of the camera is perpendicular to the document plane, the relationship $\delta_{doc} / \delta_{prj} = H / h$ holds and the value of S_v is constant over the whole projection plane:

$$S_v = \frac{H}{h}$$

Now consider the ratio of S_v over D_c. This ratio reflects how the visualisation scale varies with a distance covered by the camera during the navigation. In the situation of Fig. 3, $D_c = H \tan(\alpha / 2)$, we have

$$\frac{S_v}{D_c} = \frac{1}{h \tan(\alpha / 2)} . \quad (4)$$

As h and α are constants, the above ratio is also a constant. Visualisation scale is simply proportional to covering distance D_c during pan-zoom navigation. This is perhaps one of the valuable properties of the zooming technique that make the pan-zoom interface popular.

Fig. 4 shows the case in perspective projection. As the axis of the camera is no longer perpendicular to the document plane, two boundary points in the area covered by the camera must be distinguished: the inner boundary (IB) and the outer boundary (OB). The axis of the camera intersects the document plane at the Fixation Point (FP). Two additional notations need to be introduced here,

θ	tilting angle of camera axis,
r	½ projection screen size.

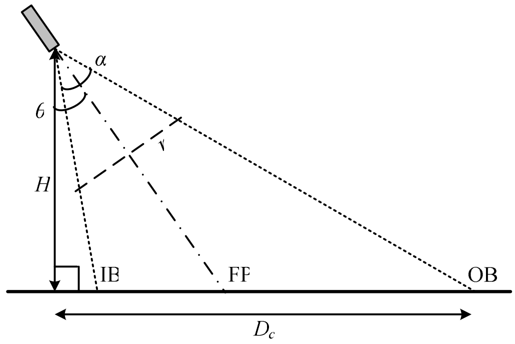

Figure 4: The virtual camera setup in the case of PV.

In perspective projection, obviously, the visualisation scale is no longer constant over the view. We will pay special attention to S_v at the point where OB is projected, because S_v changes here a lot more than at any other point, and actually the PV grasping problem mainly occurs here. In the case shown in Fig. 4, the following equation depicts a local length δ_{doc} in the document plane that is projected to the screen plane:

$$\delta_{prj} = \delta_{doc} \cdot \cos(\theta + \alpha/2) \cdot \frac{r/\sin(\alpha/2)}{H/\cos(\theta+\alpha/2)} \cdot \frac{1}{\cos(\alpha/2)}$$

In the right hand side of this equation, δ_{doc} (a very short length at OB) is followed by three factors. The first factor is there because of the camera tilt. The second item scales the length by the distance ratio. The last one is due to the fact that the screen is a plane and the view direction is not exactly perpendicular to this plane. As a result, the value of S_v at OB is

$$S_v = \frac{\delta_{doc}}{\delta_{prj}} = \frac{H\sin(\alpha/2)\cos(\alpha/2)}{r\cos^2(\theta+\alpha/2)}$$

As $D_c = H * \tan(\theta + \alpha/2)$, the ratio of S_v and D_c becomes

$$\frac{S_v}{D_c} = \frac{\sin\alpha}{r * \sin(2\theta + \alpha)}. \qquad (5)$$

When the field of view α is a fixed value, this ratio depends on θ only. If the camera is tilted until $\theta+\alpha/2$ tends to $\pi/2$ (with OB becoming almost parallel to the document plane), the denominator of the right hand side of Equation 5 cancels and the visualisation scale gets a huge value relative to covering distance D_c. This explains the PV grasping difficulty.

It would help if S_v could have a fixed linear relationship to D_c, as is the case with the pan-zoom technique, that is, like in Equation 3. To achieve this goal, the solution is just to set the right-hand side of Equation 5 to be constant. Thus, let us change the value of α so that the relationship below holds:

$$\sin\alpha = \sin(2\theta + \alpha)$$

It seems reasonable to impose limitations on the viewing angle: $0 \le \alpha \le \pi/2$ and $\pi/2 \le 2\theta + \alpha \le \pi$. Thus, $\pi - \alpha = 2\theta + \alpha$, i.e.,

$$\boxed{\alpha = \frac{\pi}{2} - \theta}, 0 \le \theta \le \pi/2 \qquad (6)$$

We have obtained a very simple relationship between α and θ. If α varies according to θ as described, we will obtain a fixed linear relationship between visualisation scale and covering distance, like in the pan-zoom case. We call this strategy the automatic coupling of viewing angle and camera tilt. Note the important difference between the basic PV implementation and this new one: the field of view was originally fixed in any case but is now allowed to vary on the direction of view. In fact, this new strategy takes some similarity with a property of our visual perception. When people stand on an endless plain and cast eye on a very remote object, he/she will concentrate on the target and try to reduce the field of view (subjectively, of course). While it is actually implemented with our assisting technique, it amounts to a zooming-in effect associated with the camera tilt, since with constant screen size, reducing the field of view implies increasing the focal length.

5 Choosing an Appropriate Type of Camera Rotation

Assuming we follow the strategy proposed above, the next question is about the location of the centre of camera rotation. Starting with an upright view of camera, (Fig. 5a), we first chose a panoramic (i.e. camera-centred, Fig. 5b) kind of rotation for our PV navigation technique, but we soon realised there was a serious shortcoming. The part of the document covered by the camera moved forward so quickly that the user most of the time missed remotely located targets. (Note this case does not appear very conspicuously in Fig. 5b. It will become more and more remarkable when the target is located far from the observation.) We then turned to the 'lunar' (fixation-point centred) rotation, as shown in Fig. 5c. Now, while the selection area was able to expand/stretch around the fixation point, the outer boundary OB in Fig. 4 was never too far thanks to the reduced field of view, but camera height varied too much. Finally the most effective solution was a rotation around the fixation point coupled with the constraint that camera altitude is fixed, as shown in Fig. 5d. We called this manipulation a *trans-rotation* of the virtual camera because this kind of rotation is a mixture of a translation and a rotation, one centred at the fixation point (FP).

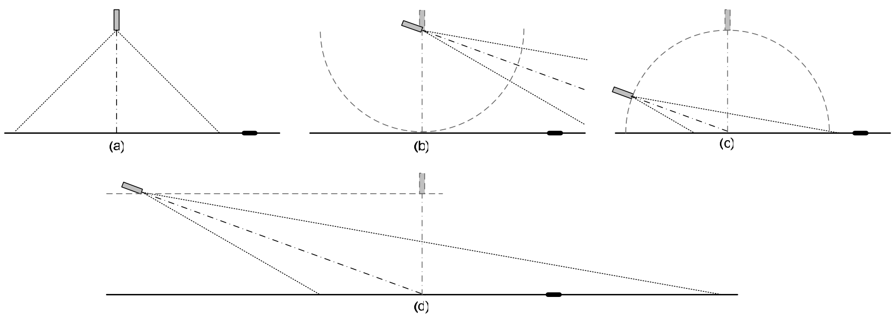

Figure 5: Three kinds of camera rotation with the strategy of coupling view angle to tilt angle. The target on document plane is represented as a thicker segment on

the horizontal line. (a) An up-right view. (b) Panoramic rotation, showing the likely case of a missed target due to a tilting overshoot. (c) Hemi-spherical or 'lunar' rotation, showing the difficulty of visualising the target because the distal boundary of the selection hardly moves at all. (d) Trans-rotation, which ensures that the target will be seen.

Note that in the case of a FP-centred rotation, target distance should be redefined as that measured from FP to the target, rather than from the camera to the target. Thus, camera rotation will not change target distance and hence the *ID*, which will only change with a camera translation. Although these three kinds of camera rotation were described in our previous study [Guiard et al., in press B], only the panoramic rotation was tested in the experiment. In the present study the trans-rotation was actually adopted and fully practised.

We discovered through experience that a trans-rotation had desirable properties for our PV interface. First, the selection area can be extended by tilting the camera direction at an infinite distance. This means the remote target can be reached, no matter how far it is. Second, as the fixation point in document space does not move, the current local focus is not lost during a camera tilt. Finally, since the altitude of the camera never changes during the search, the initial visualisation scale will be restored as soon as the camera returns to a perpendicular orientation.

In order to facilitate the automatic coupling of viewing angle and tilt angle, another idea is to limit the maximum camera tilting angle θ within $\pi/2$ and to allow for a finer control when θ approaches $\pi/2$. The camera tilt being driven by mouse-wheel, we found it quite useful to introduce a non-linear mapping between mouse-wheel turning and camera tilting. The mouse wheel rotates in a discrete way, on a notch-by-notch basis. In our previous implementation with a bare, unaided PV interface, every notch in the wheel resulted in 5 degrees of rotation in camera. This caused the document region covered by camera to grow so quickly (following a tangent function in fact) that the region seen with one notch sometimes seemed to be disconnected from that seen with the preceding notch. In order to obtain visual consistency as is the case with a PZ interface, two successive covering regions should be varied in size proportionally. In the current design, when the mouse-wheel rotates forward or backward by one notch, we change θ so that $\tan\theta$ increases/decreases by 20 percent, i.e. $\tan\theta_{n+1} = 1.2 \tan\theta_n$, as illustrated in Fig. 6. In this way, the camera is able to cover fairly large distances within a finite number of notches—yet the tilting angle of the camera will never exceed $\pi/2$. In fact, the mapping of angles between the mouse-wheel and the camera has become nonlinear. This mapping, however, is unsuitable for small camera tilts—the rotation of the camera is too slow or it squarely stops (if the camera is upright). To remedy this, we use a linear mapping for $\theta<\pi/4$, with one wheel notch causing $\pi/36$ of rotation on the camera tilt. To summarise the above, the change of tilting angle with the notch variation is depicted by the equation below:

$$\theta_{n+1} = \begin{cases} \theta_n + \pi/36, & \theta_n \le \pi/4 \\ \tan^{-1}(1.2\tan\theta_n), & \theta_n > \pi/4 \end{cases} \tag{7}$$

With the above described auxiliary strategies, when the user continuously rotates the camera, the beacon will come closer and closer to the screen centre, an effect quite reminiscent of the effect of zooming-out. Then the user needs to translate the camera while rotating it back. When the camera is almost back to a perpendicular position, the beacon disappears because the target has become large enough that it can be visualised. So the whole procedure seems consistent with that of pan-zoom technique. Our experience is that a new user manages to master the technique within minutes.

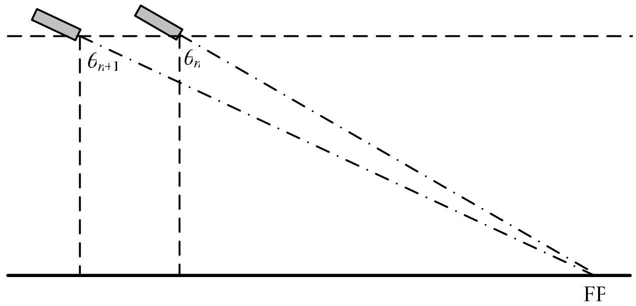

Figure 6: The camera tilt variation caused by one notch of mouse-wheel rotation.

6 User's Performance Evaluation

A formal evaluation experiment was run on a 2.8 GHz PC with 512M of RAM running Linux and X-Window, using a 17-inch monitor with a 1280x1024-pixel resolution driven by a powerful video card. The program, created with Java3D, was run in an 800x800-pixel window. A screen shot of the document used for the experiment is offered in Fig. 1, which shows a PV display.

The test was carried out with our enhanced PV navigation technique as well as with the familiar pan-zoom (PZ) technique, which provided a reference for comparison. The only input device was a standard optical wheel-mouse. Both in PV and PZ, the camera panning was driven by mouse moving. The camera zooming in PZ and tilting in PV were controlled by mouse wheel turning. The details of Fitts' pointing experiment in PZ interface can be found in Guiard et al. [2001].

Eight unpaid volunteers participated in the performance test. The experiment was divided into trials, each trial consisting of 6 movements (i.e., 7 successful target clicks). Four values of *ID* were chosen for the experiment (12, 17, 22 and 27 bits) each participants running four trials at each level of *ID*. The trial sequences were arranged pseudo-randomly according to Latin squares. There were a total of 16 trials per participant and per technique. Four participants started the session with PV and switched to the PZ technique, the other four ran the experiment in the reverse order. The participants were invited to have a rest between the two parts of the session.

Before the test, the participant was offered a few minutes of practice to familiarise with the new interface and the navigation techniques. When the participant felt confident, the formal test started, with the program recording all users' activities as well as all motions of the virtual camera. The recorded data were processed with the following rules. In each trial, the first two movements were ignored as warm up. Movement time (*MT*) was defined as the time elapsed between two successful clicks. Note that the program ignored unsuccessful clicks, just waiting until a target hit was recorded before presenting the next target. Since, therefore, the error rate was a forced 0%, we may now use *MT*, on its own, to estimate performance quantitatively.

Fig. 7 shows the performance or the participants in our enhanced PV interface and in the standard PZ interface. The *MT* measures reported in our previous paper [Guiard et al., in press B] for the unaided implementation of the PV navigation technique are also shown for comparison.

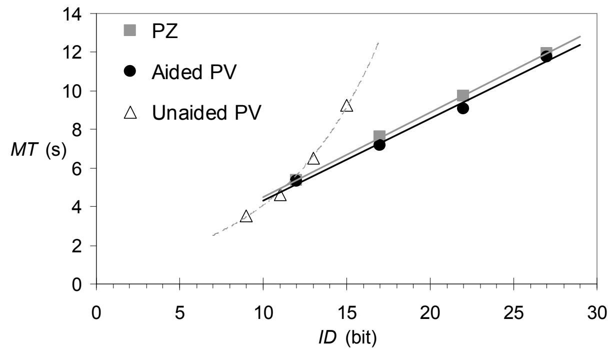

Figure 7: Movement time for the enhanced PV and the PZ techniques, plotted as a function of the *ID*. Also shown are the *MT* measures obtained with the unaided PV technique. *MT* is averaged over all participants.

With the unaided PV technique of Guiard et al., *MT* raised non-linearly with the *ID* up to a critical *ID* of 15 bits—beyond which the task was actually no longer workable. In contrast, the figure shows that with our enhanced PV technique extremely difficult pointing tasks, up to *ID*=27 bits in the present test, could be successfully handled. It should be realised that applied to the case of a document formatted according to the norms of the IEEE proceedings, a task with an *ID* of 27 bits would mean reaching and selecting one particular text line located at a distance of 3 million pages.

For both the PZ and the enhanced PV conditions, *MT* varies linearly with the *ID*, in keeping with Fitts' law. The linear equations of best fit are

PZ: $MT = 0.437 * ID + 0.132$, $r^2 = 0.9999$

Enhanced PV: $MT = 0.423 * ID + 0.083$, $r^2 = 0.9905$

The important implication of the finding that *MT* does obey Fitts' law in our enhanced PV interface is that because time now raises linearly with the *ID*, any target reaching task, however difficult, can be carried out—the upper limit of difficulty characteristic of unaided PV navigation has vanished.

The other point to be made from Fig. 8 is that our enhanced PV navigation technique yielded a performance quite similar to that obtained with the usual PZ technique. In particular, the intercepts are very close and the curves are roughly parallel, with no obvious crossing point. In both cases, performance bandwidth (the inverse of Fitts' law slope) is about 2.3 bits/s. A two-factor ANOVA run on *MT* with repeated measures on the *ID* factor (12, 17, 22 and 27 bits) and the technique factor (PZ vs. aided PV) confirmed the visual impression gained from the figure. Beside the trivial finding of a highly significant *ID* effect ($F_{3, 21} = 281.05$, $p < 0.0001$), neither the main effect of the technique nor the two-factor interaction was statistically reliable ($F_{1, 7} < 1$ and $F_{3, 21} < 1$, respectively).

7 Conclusion

We have described a novel multi-scale visualisation techniques based on perspective viewing, the main merit of which is that it offers a gradual transition from local detail to remote view [Mackinlay 1991]. With a basic, system-unaided implementation of the PV technique, we showed that it is almost impossible for users to perform a difficult pointing task (*ID*>15 bits), because a severe accuracy problem arises for selecting an appropriate grasping point at very large distances in the document. We have offered a formal analysis of this problem and proposed a solution. First, we automatically couple viewing angle and tilt angle. To further facilitate navigation, we describe another two assistance techniques, one consisting of a nonlinear mapping between the mouse-wheel and the camera tilt, and the other based on a new kind of camera rotation, trans-rotation. Our evaluation data showed excellent pointing performance with the newly developed PV interface. *MT*s conform to the classical Fitts' law, up to a considerable 27 bits, without any evidence that the technique will fail at higher levels of *ID*.

The technique we have described and evaluated in this study should be thought of as an optional resource for PV navigation rather than a permanent feature, because obviously reducing the viewing angle has a cost for the user. The trick we designed makes it possible to reach objects located in the documents at an extremely large distance from the current observation point (a case that needed to be considered as it may occasionally happen), but the shortcoming is that it sacrifices what is perhaps the fundamental advantage of PV—the possibility of seeing both the local detail and the global environment during navigation, just like in real-world vision where all surfaces are actually seen in perspective. Therefore the best way to implement our assistance technique is presumably in the form of an easily reversible mode within PV navigation.

In view of the considerable promise of PV visualisation, we feel that efforts to improve PV navigation by means of appropriate assistance tricks like those we described in this paper are quite worth the while. From the moment the PV navigation technique has been cured of its specific weakness, that which we designate as the grasping problem for extremely high *ID*s, we may seriously contemplate proposing camera tilts as a basic facility for document navigation in graphical user interfaces. After all, introducing a camera tilting facility in interfaces simply means providing users with extended (translational *plus* rotational) control over the virtual-camera, an improvement relative to the current pan and zoom technique, which only exploits camera translations. Although quite simple conceptually and technically easy today, given the power of current tools like graphical cards and OpenGL programming, the change we recommend is likely to have diverse and far reaching consequences. Identification of these consequences is an intriguing problem tailored for human-computer interaction research. It will take time, demanding both formal evaluation experiments and user experience with realistic prototypes.

Our next step will be to compare, using the same experimental platform, our enhanced PV navigation technique with the best multi-scale visualisation techniques that have been recently proposed, such as the Speed dependent automatic zooming technique (SDAZ) of Igarashi & Hinckley [2000] and the OrthoZoom technique of Appert & Fekete [2006]. In the present study, as has been traditionally the case in Fitts' law studies, the pointing task was performed on an empty plane—a blank, if textured, document. We are now implementing a 3D visualisation of a real large document which consists of the 150,000 verses of William Shakespeare's complete works [Guiard et al., in press A]. We believe that asking people to find one particular line on a real text document rather than an abstract graphical strip on an empty plane makes up a more interesting and meaningful experimental task, as well as one that allows the experimental evaluation of new multi-scale document navigation techniques under more realistic conditions.

Acknowledgements

We gratefully acknowledge the generous financial support of the French Ministère délégué à la recherche et aux nouvelles technologies (ACI Masses de données, Micromegas Project).

References

Appert C. & Fekete J.D. [2006]. OrthoZoom Scroller: 1D Multi-Scale Navigation. *Proceedings of ACM CHI* 2006, 21-30. ACM Press.

Arsenault, R. & Ware, C. [2002]. Frustum View Angle, Observer View Angle and VE Navigation. *Proceedings of SVR 2002*.

Bederson, B.B. & Hollan, J.D. [1994]. PAD++: A Zooming graphical interface for exploring alternate interface physics. *Proceedings of UIST 94*, ACM, 17-26.

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. (Reprinted in *Journal of Experimental Psychology: General 121*, 262-269, 1992).

Furnas, G.W. [1986]. Generalized fisheye views. *Proceedings of ACM CHI 86 Human Factors in Computing Systems*, ACM press, 16-23.

Furnas, G. W. & Bederson, B. B. [1995]. Space-scale diagrams: Understanding multiscale interfaces. *Proceedings of ACM CHI'95 Human Factors in Computing Systems*, ACM Press, 234-241.

Guiard, Y. & Beaudouin-Lafon, M. [2004]. Target acquisition in multiscale electronic worlds. *International Journal of Human-Computer Studies 61*, 875-905.

Guiard, Y., Bourgeois, F., Mottet, D., & Beaudouin-Lafon, M. [2001]. Beyond the 10-bit barrier : Fitts' law in multi-scale electronic worlds. *Proceedings of IHM-HCI 2001* (pp. 573-587). London : Springer.

Guiard, Y., Du, Y., Fekete, J.-D., Beaudouin-Lafon, M., Appert, C., & Chapuis, O. [in press A]. Shakespeare's complete works as a benchmark for evaluating multiscale document-navigation techniques. *Proceedings of BELIV06*, a Workshop of the AVI 2006 International Working Conference.

Guiard, Y., Chapuis, O., Du, Y., & Beaudouin-Lafon, M. [in press B]. Allowing camera tilts for document navigation in the standard GUI: A discussion and an experiment. *Proceedings of Advanced Visual Interfaces* (AVI 06). Venice (Italy), May 2006. ACM Press.

Gutwin, C. [2002] Improving focus targeting in interactive fisheye views. *Proceedings of ACM CHI 2002*, 267-274.

Igarashi, T. & Hinckley, K. [2000]. Speed-dependent automatic zooming for browsing large documents. *Proceedings of* UIST'00, San Diego, CA, November 5-8, 139-148.

Khan, A., Komalo, B., Stam, J., Fitzmaurice G., & Kurtenbach, G. [2005]. HoverCam: interactive 3D navigation for proximal object inspection, *Symposium on Interactive 3D Graphics: Proceedings of the 2005 symposium on Interactive 3D graphics and games*, 73-80.

MacKenzie, I. S. [1992]. Fitts' law as a research and design tool in human-computer interaction. *Human-Computer Interaction 7*, 91-139.

Mackinlay, J.D., Card S.K., & Robertson, G.G. [1990]. Rapid Controlled Movement Through a Virtual 3D Workspace. *Computer Graphics 24*, 171-176.

Mackinlay, J.D., Robertson, G. G. & Card., S. K. [1991] The perspective wall: Detail and context smoothly integrated. *Proceedings of ACM CHI'91*, 173-179.

Perlin, K. & Fox, D. [1993]. Pad: An Alternative Approach to the Computer Interface. *Proceedings of ACM SigGraph '93*, Anaheim, CA, 57-64.

Tan, D., Robertson, G. & Czerwinski, M. [2001]. Exploring 3D navigation: combining speed-coupled flying with orbiting. *ACM CHI*, 418-425.

Tzavaras, I., Spence, R. [1982]. A bifocal display technique for data presentation. P*roceedings of Eurographics 82*, 27-43.

Zeleznik, R. & Forsberg, A. [1999]. UniCam - 2D Gestural Camera Controls for 3D Environments. *Proceedings of ACM Symposium on Interactive 3D Graphics*, 169-173.

Appendix 1

Here to explain why an *ID* of 15 bits seems to be a critical value when the grasping difficulty is raised indefinitely. Fig. 9 shows the target projection with a tilted camera. The centre of the camera, O, is at a height *H*. For the sake of simplicity a spherical rather than planar screen is used (no essential difference will result). The view is represented as a quarter of circle PQMN. The direction OP being parallel to the document plane, P appears at the horizon in the view. When the target of width *W* is at a short distance, it is projected on the screen as arc MN. When the target is at a considerable distance *D* from the observation point, its projection size becomes smaller than one pixel, its beacon marking the single point Q.

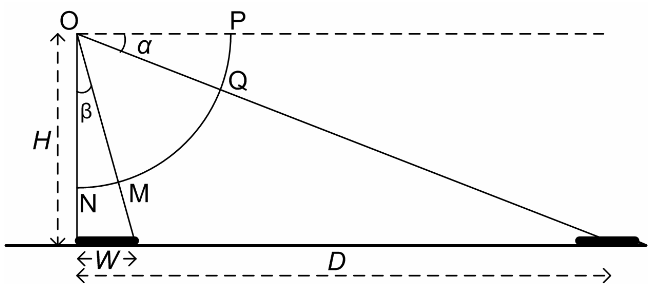

Figure 9: Elements of the grasping difficulty problem in PV.

Now suppose that the target is very far away, at a distance such that PQ exactly clips the highest pixel of the view. It will be difficult to grasp a target whose distance is farther than D because from the target to infinity everything will be mapped to this last pixel. If the screen cursor grasps this pixel to drag the target back, the mapped position of the mouse cursor in document space will be essentially undetermined.

Let us estimate the *ID* for this limiting case. The screen has a resolution of *R* pixels evenly distributed over arc PQMN. As PQ exactly clips one pixel, the angle between OP and OQ should be $\alpha = \frac{1}{R}\frac{\pi}{2}$. When the target has a maximum projection size, say *S* pixels (camera upright), there are *S* pixels on the arc MN, then $\beta = \frac{S}{R}\frac{\pi}{2}$. As $\tan\alpha = \frac{H}{D}$ and $\tan\beta = \frac{W}{H}$, we have,

$$\begin{aligned} ID &= \log_2\left(\frac{D}{W}+1\right) \\ &= \log_2\left(\frac{D}{H}\frac{H}{W}+1\right) \\ &= \log_2\left(\frac{1}{\tan\alpha\tan\beta}+1\right) \\ &= \log_2\left(\frac{1}{\tan\frac{\pi}{2R}\tan\frac{S\pi}{2R}}+1\right) \end{aligned}$$

It can be seen that this critical value of the *ID* depends on screen resolution *R* and the maximum projection size of target *S*. In a practical condition, *R*=800 pixels and *S*=10 pixels. By the above equation, we obtain an estimate of 14.7 bits for the critical *ID*.