Navigation techniques for small-screen devices: an evaluation on maps and web pages

Stefano Burigat, Luca Chittaro and Silvia Gabrielli

HCI Lab, Dept. of Math and Computer Science, University of Udine, Via delle Scienze 206, 33100, Udine, Italy

Abstract

Several techniques have been proposed to support user navigation of large information spaces (e.g., maps or web pages) on small-screen devices such as PDAs and Smartphones. In this paper, we present the results of an evaluation that compared three of these techniques to determine how they might affect performance and satisfaction of users. Two of the techniques are quite common on mobile devices: the first one (DoubleScrollbar) is the standard combination of two scrollbars for separate horizontal and vertical scrolling with zoom buttons to change the scale of the information space, the second one (Grab&Drag) enables users to navigate the information space by directly dragging its currently displayed portion, while zooming is handled through a slider control. The last technique (Zoom-Enhanced Navigator or ZEN) is an extension and adaptation to mobile screens of Overview&Detail approaches, which are based on displaying an overview of the information space together with a detail view of a portion of that space. In these approaches, navigation is usually supported by (i) highlighting in the overview which portion of space is displayed in the detail view, and (ii) allowing users to move the highlight within the overview area to change the corresponding portion of space in the detail area. Our experimental evaluation concerned tasks involving maps as well as web page navigation. The paper analyzes in detail the obtained results in terms of task completion times, number and duration of user interface actions, accuracy of the gained spatial knowledge, and subjective preferences.

Key words: Small-screen devices, navigation techniques, user study, mobile interaction

Email addresses: `burigat@dimi.uniud.it`, `chittaro@dimi.uniud.it`, `gabrielli@dimi.uniud.it` (Stefano Burigat, Luca Chittaro and Silvia Gabrielli).

1 Introduction

Mobile devices such as PDAs and Smartphones are increasingly used to support information needs of users on the move. As a consequence, information spaces that have been traditionally available only to desktop and laptop users, such as documents, pictures, web pages, maps, are moving to small screens as well, presenting application designers with new challenges. Indeed, the common form factors of mobile devices constrain screen space to a small fraction of what is available on a desktop. For example, a typical 240x320 pixels display of a PDA has less than 1/16 the area of a typical 1280x1024 desktop display (see Fig. 1). Such limitation makes it extremely difficult for users to navigate information spaces that do not fit a single screen, unless appropriate techniques to simplify navigation are provided.

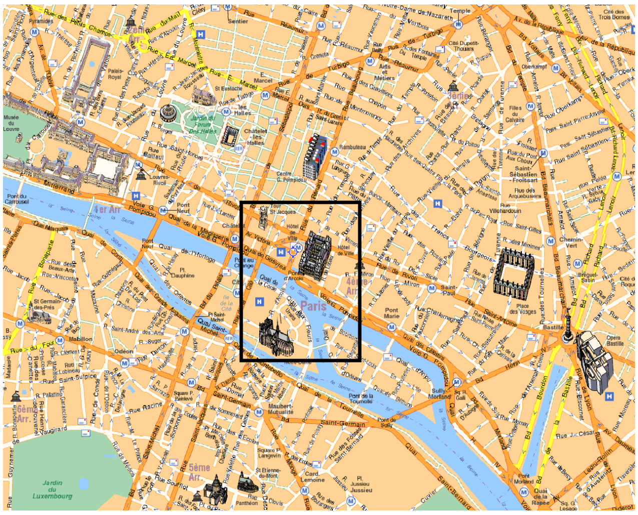


Fig. 1. Comparison between the size of a typical 240x320 PDA screen (the area highlighted by the black rectangle) and a common 1280x1024 desktop screen (the whole picture).

In this paper, we concentrate on the most common approach that has been proposed to face this problem, i.e., using the small screen as a window (hereinafter, *viewport*) onto the larger information space and providing users with appropriate means to change the portion of that space that is displayed through the viewport. Various techniques have been proposed in the literature to implement this approach, usually by adapting solutions that were originally designed for the desktop. However, the effects of these different techniques on performance and satisfaction of users have been scarcely studied, and techniques which have been shown to be effective in the desktop domain are not guaranteed to provide the same level of support when ported to mobile devices.

In this paper, we evaluate three different techniques for supporting user navigation of large information spaces on small-screen devices. Two of the techniques

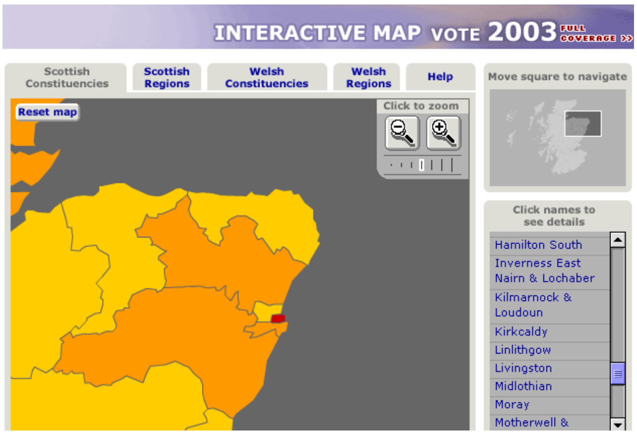


Fig. 2. Example of an Overview&Detail approach in a desktop application. The overview is displayed in the upper right corner and highlights (through a viewfinder) which portion of map is displayed in the detail view on the left. The viewfinder can be directly moved to navigate the map. Source: BBC News web site.

have been considered because they are commonly used in mobile applications. The first is based on scrollbars for moving the viewport and zoom buttons to change the scale of the information space. The second is based on dragging the displayed portion of information space to pan and using a zoom slider to change the scale of the information space. The third technique is an extension and adaptation to mobile screens of *Overview&Detail* approaches, which are based on displaying an overview of the information space together with a detail view of a portion of that space. In these approaches, the overview is usually a small-scale thumbnail of the whole information space that includes a properly positioned graphical highlight of the portion of space displayed in the detail view (see Fig. 2 for an example). The highlighted portion (hereinafter, *viewfinder*) can usually be directly moved within the overview to pan the information space. Our technique aims at improving this solution, allowing users to zoom by directly manipulating the size of the viewfinder. Moreover, since screen space on mobile devices is at a premium, we display only the outline of the overview, overlaid on the detail view. Unlike previous studies that investigated other panning and zooming techniques for small-screen devices using emulators on desktop systems (Jones et al., 2005), we carried out our evaluation on a real PDA. To this end, we took care of efficiently implementing the studied techniques so that they could run interactively on the intended platform.

In Section 2, we describe previous research that dealt with the problem of displaying and navigating large information spaces on small-screen devices and report on prior evaluations in this area. In Section 3, we describe in detail the techniques we considered and provide a technical comparison of their features. In Section 4, we present the user study we carried out to assess how these techniques supported users in navigating two different types of information spaces, namely maps and web pages, on a PDA. Finally, we discuss the obtained results and the application of these techniques to small-screen user interfaces.

2 Related work

Dealing with information spaces that do not fit a single screen is a well-known problem in HCI research, and a considerable effort has been devoted to the study of different representations and navigation techniques, especially for large documents and 2D data spaces. Solutions to this problem can be organized into five general classes: restructuring of the information space, traditional scrolling/panning and zooming techniques, Overview&Detail approaches, Focus&Context approaches, and off-screen objects visualization.

2.1 Restructuring the information space

To view large information spaces on small screens, researchers have often looked at restructuring the information space itself, especially in the case of web pages (Chen et al., 2003; Trevor et al., 2001). For example, a basic approach consists in manually designing web pages specifically for each target device (Jacobs et al., 2003). When this is not feasible, a possible solution is based on automatically reformatting pages (Björk et al., 1999; Buyukkokten et al., 2000). Most commercially available web browsers for mobile devices are able to reformat web pages by concatenating all columns, thus providing a single-column viewing mode. However, these reformatting techniques significantly affect the layout of pages, thus making it difficult for users to leverage their experience with desktop web browsing. To solve this problem, researchers have proposed to display web pages as thumbnails, i.e., scaled down versions of pages that fit the width of the small screen and are sometimes restructured to improve user recognition of their different parts. In this way, users can start viewing a web page in thumbnail mode to identify an area of interest, and then zoom into that area for reading. For example, Summary Thumbnails (Lam and Baudisch, 2005) are thumbnail views that preserve the original page layout that allows users to recognize the overall page structure, but also contain readable summaries of the textual areas, so that users can disambiguate the desired information from similar looking areas. The MiniMap method (Roto et al., 2006) changes the size of the text relative to the rest of the page contents and limits the maximum width of the text paragraphs to the width of the browser viewport. An overview of the web page with an indication of the current viewport is then overlaid transparently on top of the browser viewport, thus providing users with a navigation aid and helping them to locate information inside the page.

Techniques for restructuring an information space to fit a small screen are valuable in situations where restructuring is possible. However, when it is necessary to maintain the layout of the original information space (e.g., when using maps to assess distances between locations or when viewing photographs), it is only possible to rely on navigation approaches. A basic approach is to provide users with functionalities that allow them to select the portion of space to visualize, leaving the remaining space off-screen. This is achieved by moving the viewport over the information space. Conventional approaches rely on *scrolling* by means of scrollbars that provide separate horizontal and vertical viewport control or *panning* by directly dragging the information space in any direction. In general, panning has been found to work well for relatively small information spaces (Plaisant et al., 1995), but to be rather tedious for larger ones (Kaptelinin, 1995). A number of systems have thus exploited *zooming*, which changes the scale of the information space and can be used to obtain multiple perspectives on it (Gutwin and Fedak, 2004). Systems where zooming is used to provide an overview of the space have been shown to perform better than systems based on scrolling only (Kaptelinin, 1995). In Zoomable User Interfaces such as Pad++ (Bederson and Hollan, 1994), panning and zooming are coupled with the *semantic zooming* concept: the representation of an object depends on the scale of the information space and objects can change size, shape, details or they can appear/disappear from the visualization when zoomed. Typically, users must interact with different controls for scrolling/panning and for zooming (e.g., scrollbars and zoom buttons). To simplify interaction, solutions such as *control menus* (Pook et al., 2000) have been proposed for desktop systems to provide unified and rapid selection of operations (e.g., pan and zoom) in a way that is similar to pie menus and allowing control of the chosen operation in the same gesture. On mobile devices, various interaction techniques have been proposed to simplify scrolling, panning and zooming. For example, ZoneZoom (Robbins et al., 2005) is an input technique that lets users easily explore large images on Smartphones: each image is partitioned into nine cells, each one mapped into a number of the phone keypad, and pressing a key produces an automated pan and zoom on the associated cell (which can then be recursively partitioned into nine more cells). Rosenbaum and Schumann (2005) propose an adaptation of the ZoneZoom technique to PDAs to pan and zoom on images by interacting with a grid overlaid on the currently displayed image portion. The grid size is proportional to the size of the whole image and each grid cell can be tapped to zoom on the corresponding portion of the image. Cells can also be merged or split to provide users with different zoom levels. Jones et al. (2005) take the Speed-Dependent Automatic Zooming (SDAZ) technique proposed by Igarashi and Hinckley (2000) for navigating documents and adapt it to mobile devices. SDAZ combines scrolling and zooming into a single operation, where the zoom level decreases as scroll speed increases, and has been shown to outperform standard scroll, pan and zoom methods in document and map navigation tasks (Cockburn and Savage, 2003). In the SDAZ version by Jones et al. (2005), two concentric circles are drawn when users tap on the information space with the pointer. If the pointer remains within the inner circle, the user is free to pan in any direction and the panning rate increases as the pointer moves away from the starting position. When the pointer moves beyond the inner circle, both zooming and panning operations take place. The information space progressively zooms out as the user moves closer to the outer circle and the panning speed changes to maintain a consistent visual flow. When the pointer reaches the outer circle, no further zooming occurs, while panning remains active.

2.3 Overview&Detail approaches

Although scrolling, panning and zooming techniques allow users to explore an information space at different levels of detail, it is often useful to display more than one level of detail simultaneously, as done by *Overview&Detail* approaches. These approaches provide one or multiple overviews of the space (usually at a reduced scale), together with a detail view of a specific portion of space (Plaisant et al., 1995). For example, the Large Focus-Display (Karstens et al., 2004) provides two separate views of the information space, one for context and one for detail. The context view displays a downscaled version of the information space and highlights the portion of space displayed in the detail view with a rectangular viewfinder. Users can drag the viewfinder to navigate the information space. By examining the size and position of the viewfinder in the context view, users are also able to derive useful information for navigation, such as the scale ratio between the displayed portion and the whole information space. Although Overview&Detail approaches have been found to be useful in desktop interfaces (Hornbaek and Frokjaer, 2001), they are problematic on mobile devices, because the screen space that can be assigned to visualize overviews is typically insufficient to allow the user to easily relate them to the detail view (Chittaro, 2006). For example, Buring et al. (2006*a*) report the results of a user study in which participants performed search tasks on scatterplots by using two applications on a PDA, one displaying a detail view and an overview and the other displaying only the detail view. While there was no significant difference in user preference between the interfaces, participants solved search tasks faster without the overview. This indicates that, on small screens, a larger detail view can outweigh the benefits gained from an overview window. An alternative solution, such as displaying the overview on-demand, can save screen space but makes it impossible for the user to see the two views at the same time, requiring users to switch attention from one view to the other and to remember the contents of the unshown view.

2.4 *Focus&Context approaches*

An alternative to Overview&Detail approaches is to display the information space at different levels of detail simultaneously, without separating the different views. Techniques based on this *Focus&Context* approach usually display one or multiple focus areas with undistorted content embedded in surrounding context areas that are distorted to fit into the available screen space. *Fisheye views* (Furnas, 1986) are one way of integrating context and focus into a single view. In the Rectangular FishEye View (Rauschenbach et al., 2001), a rectangular focus is surrounded by one or more context belts, appropriately scaled to save screen space. Different schemes are used to choose the scaling factor for each context belt, in such a way that less detail is displayed as the distance from the focus increases. A fisheye view technique coupled with compact overviews is used in the DateLens calendar interface for PDAs (Bederson et al., 2004). The basic operation of DateLens is to start with an overview of a large time period using a graphical representation of each day's activities. Tapping on any day expands the area representing that day, displaying details about appointments while keeping an overview of the other days. Two user studies have been carried out to compare DateLens with traditional calendar visualizations. In both studies, Datelens allowed users to perform complex tasks significantly faster with respect to a default PDA calendar, even if users familiar with the default calendar strongly preferred its daily view and behaviors. *Variable-scale maps* (Harrie et al., 2002) apply the same principle used in the Rectangular FishEye-View but show in full detail a circular area surrounding a specific point (not necessarily the center of the viewed area) while using a smaller scale and applying generalization and distortion operations to fit the remaining part of the information space on the screen. Unlike variable-scale maps, *focus maps* (Zipf and Richter, 2002) are not based on distortion but on subdividing the information space into different regions of interest and displaying each region with a different amount of detail according to its degree of interest. In a map-based example, regions of interest may comprise the region a user is currently in and, if the current task involves movement, the regions she is about to encounter. In this way, user attention is directly drawn towards those regions that are currently most relevant, but the other regions can still be used, for example, to help the user locate and orient herself. The disadvantage of Focus&Context techniques is that the different scales and the introduced distortions make it more difficult for users to integrate all information into a single mental model and interfere with tasks that require precise geometric assessments (Baudisch et al., 2002). For example, an evaluation carried out on desktop systems by Nekrasovski et al. (2006) has shown that a traditional panning and zooming technique enables users to be significantly faster and to require less mental effort in navigating an information space than a Rubber Sheet technique, i.e., a Focus&Context technique that allows users to explore areas of an information space at multiple level of details by stretching or squeezing rectilinear focus areas (Sarkar et al., 1993). However, a study that compared a semantic zooming technique and a fisheye view technique to support users in interacting with scatterplots on small screens found no significant difference in task-completion times and a higher level of user preference for the latter technique (Buring et al., 2006*b*).

2.5 Off-screen object visualization

While Overview&Detail and Focus&Context techniques can facilitate the exploration of large information spaces, they introduce additional interaction and cognitive costs that make them unsuitable for users who need large undistorted content to perform spatial tasks, such as first responders who need to identify locations of potential hazards in a building or view the real-time location of other team members on a map. To better support these users, one can enable them to locate relevant objects on the information space even when they are outside the area displayed in the viewport. This is the approach followed by Mackinlay et al. (2003), who proposed CityLights, i.e., compact graphical representations such as points, lines or arcs which are placed along the borders of a window to provide awareness about off-screen objects located in their direction. In a desktop scenario, CityLights lines have been used to inform users about the presence and size of hidden windows in a spatial hypertext system. In mobile scenarios, a variation of CityLights, called Halo (Baudisch and Rosenholtz, 2003), shows off-screen object locations by surrounding them with circles that are just large enough to reach into the border region of the viewport. By looking at the position and curvature of the portion of circle visualized on-screen, users can derive the off-screen location of the object located in the circle center. A user study has shown that Halo enables users to complete map-based route planning tasks faster than a technique based on displaying arrows coupled with labels for distance indication, while a comparison of error rates between the two techniques did not find significant differences. In a recent study (Burigat et al., 2006), we have compared Halo with two different approaches based on arrows. In the two arrow-based approaches, arrows displayed at the border of the viewport pointed at off-screen objects and their size and body length, respectively, informed about the distance of objects. We found that arrows allowed users to order off-screen objects faster and more accurately according to their distance, while Halo allowed users to better identify the correct location of off-screen objects. Our study also investigated the effectiveness of the three techniques with respect to the number of off-screen objects. Our findings suggest that when the cognitive demand on the user is higher, arrow-based visualizations can outperform Halo, especially in the case of cluttered configurations where several off-screen objects must be taken into account. However, Halo is a more suitable solution when precise geometric assessments are needed. While approaches such as Halo can provide awareness of off-screen objects, getting to those objects requires considerable navigation effort from the user when only standard navigation techniques such as zooming, panning and scrolling are available. To solve this problem, Irani et al. (2006) have proposed an interaction technique, *hopping*, that combines the Halo approach with a mechanism that enables users to quickly access off-screen objects. To save space and reduce the amount of overlapping along the edges of the screen, Halos are drawn using ellipses for objects that are directly north, east, south, and west of the viewport and circles for any other location. The second component of the hopping technique is a laser beam that appears when the user taps on the information space and drags the pointing device toward an edge. The laser beam is drawn from the tapped position up to the edge of the screen. The user then moves the pointing device radially, and the laser beam moves until it intersects a halo. For each intersection, a proxy (i.e., a temporary duplicate of the off-screen object) is created and placed in the area between the initial and the currently tapped point, allowing users to interact with the remote object. Proxies remain opaque for one second, and then begin to fade away. If users select a proxy, they are teleported to the corresponding off-screen object using an animated transition. An experimental study carried out on a desktop computer showed that users are significantly faster at selecting off-screen targets with hopping rather than standard zooming or panning techniques.

3 The considered techniques

In this section, we describe in detail each of the three navigation techniques we considered in our study, then we technically compare them from different points of view.

3.1 DoubleScrollbar

The first technique (hereinafter, *DoubleScrollbar*) is very frequently seen in desktop as well as mobile user interfaces. It allows users to perform scrolling operations by using separate vertical and horizontal scrollbars (Fig. 3), and zooming operations by choosing a specific zoom level among a predefined set. The current level is indicated by a percentage and users can change it either through a menu (Fig. 4) or by tapping on two specific icons depicting a magnifying glass with a plus or minus sign. The plus (minus) icon is also grayed out when the maximum (minimum) zoom level is reached. As users perform panning or zooming operations, the position and length of both scrollbars thumbs (i.e., the draggable sections of scrollbars) change dynamically to highlight which portion of information space is currently displayed on screen.

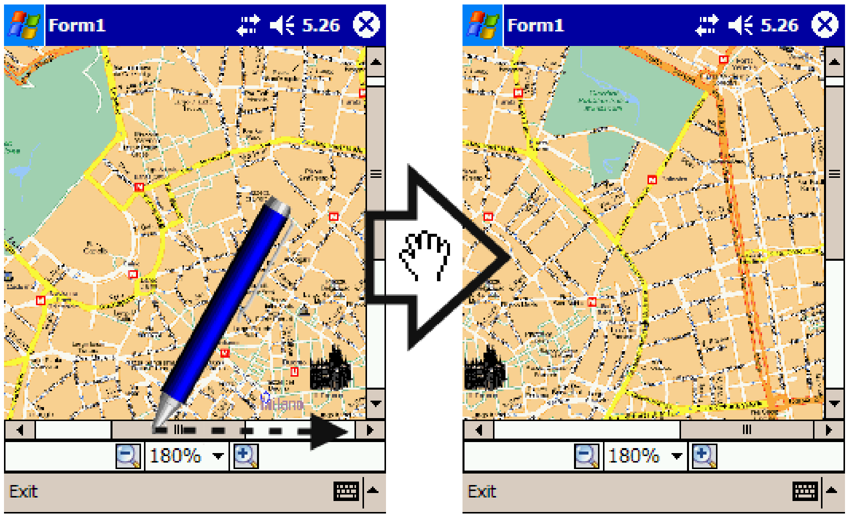


Fig. 3. Panning with DoubleScrollbar is carried out by using separate vertical and horizontal scrollbars. (The figure shows how the horizontal scrollbar is operated).

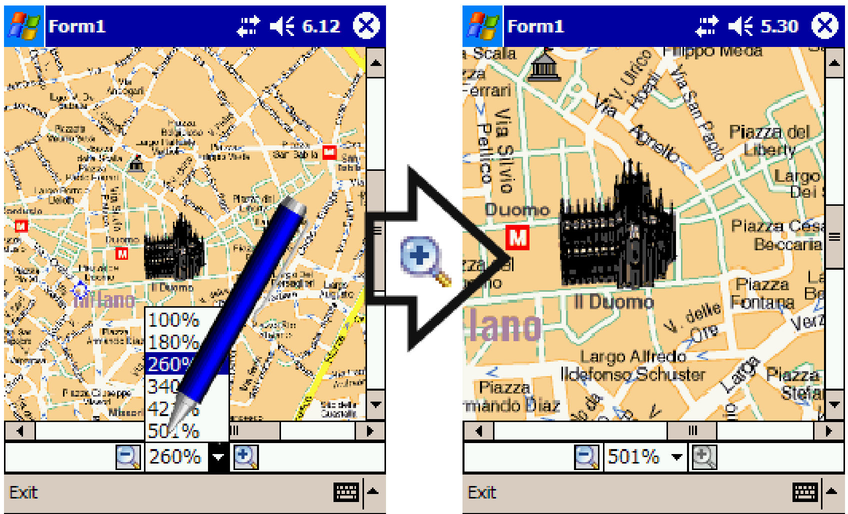


Fig. 4. Zooming with DoubleScrollbar is carried out by choosing a specific zoom level among a predefined set, either through a menu or by tapping on the two icons depicting a magnifying glass.

3.2 Grab&Drag

Like DoubleScrollbar, the second technique (hereinafter, *Grab&Drag*) is widely used in desktop and mobile user interfaces. Grab&Drag allows users to perform panning by dragging the portion of information space displayed in the viewport (Fig. 5). On a PDA, dragging is carried out by moving the stylus in any direction while keeping it in contact with the screen. Zooming is performed by operating a slider (Fig. 6). The zoom level is incremented (decremented) by dragging the slider thumb towards the plus (minus) sign and the currently selected zoom level is displayed as a percentage.

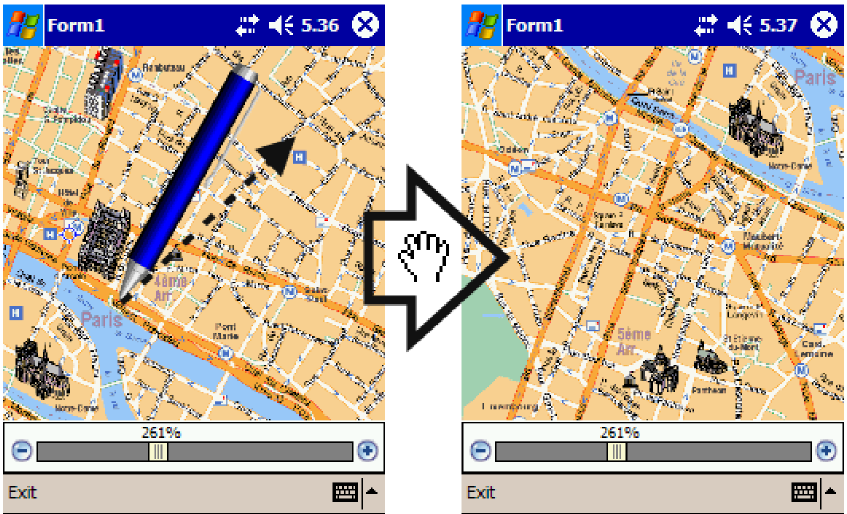


Fig. 5. Panning with Grab&Drag is carried out by grabbing and dragging the portion of information space displayed in the viewport.

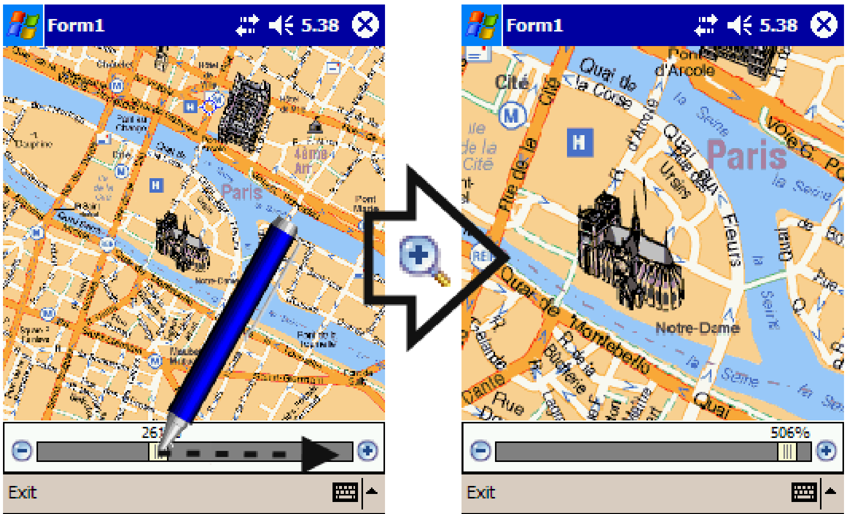


Fig. 6. Zooming with Grab&Drag is carried out by operating the slider at the bottom of the interface.

3.3 Zoom-Enhanced Navigator

The third technique, called *Zoom-Enhanced Navigator* (ZEN), is an extension and adaptation of Overview&Detail approaches to mobile devices. As previously mentioned, Overview&Detail approaches are frequently employed in desktop interfaces but difficult to port to (and rarely employed in) mobile applications. Our idea was to show users only an outline of the overview, thus saving screen space, and to let users change the scale of the information space by directly changing the size of the viewfinder, thus integrating panning and zooming in the same interaction mechanism.

ZEN is composed of the graphical elements illustrated in Fig. 7. The black rectangular outline provides information about the proportions of the information space; the red rectangular outline within the overview is the viewfinder and can be manipulated to navigate the information space; the gray circle inside the viewfinder highlights the area the user can tap to pan the information space (Fig. 8). Panning is performed by dragging the viewfinder in the desired direction within the black rectangular outline, thus changing the portion of information space displayed in the detail view. If the user taps instead the viewfinder area between the circle and the red outline, she becomes able to

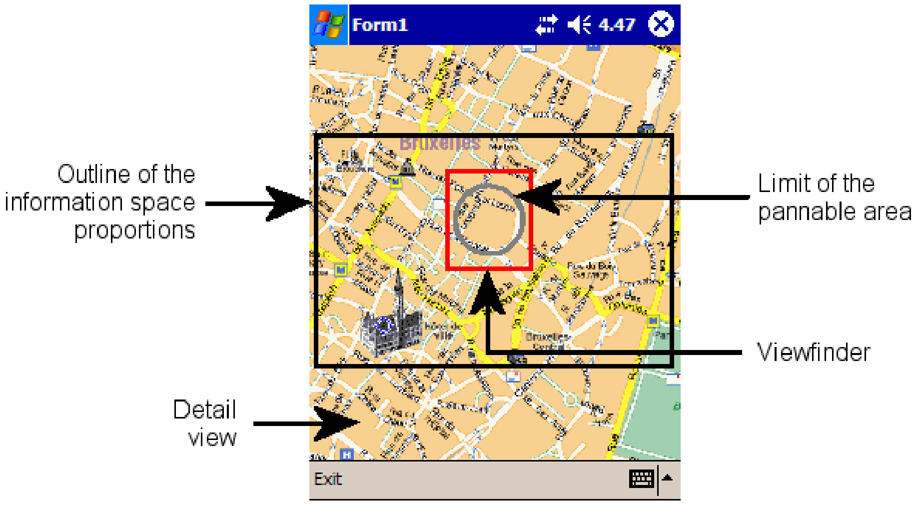


Fig. 7. ZEN is composed of three graphical elements: a black rectangular outline that provides information about the proportions of the information space, a red rectangular outline as viewfinder and a gray circle that defines the limit of the panning area. The elements are graphically emphasized in this figure for illustration purposes.

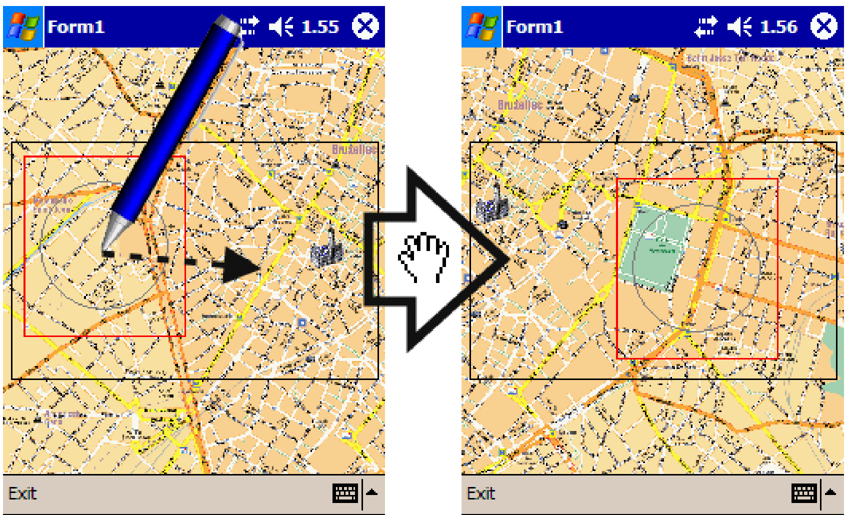


Fig. 8. Panning with ZEN is carried out by tapping the area inside the gray circle in the viewfinder and dragging it in the desired direction.

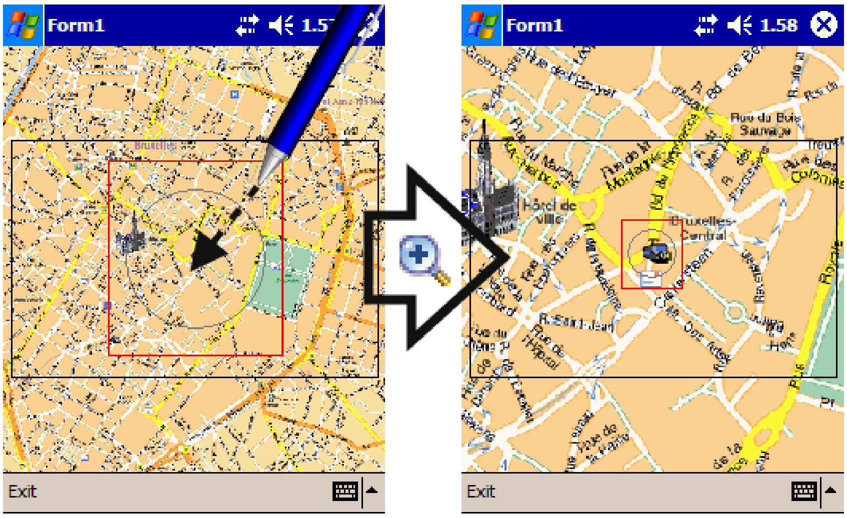


Fig. 9. Zooming with ZEN is carried out by tapping the area between the viewfinder border and the inner gray circle border and dragging it inwards (to zoom in) or outward (to zoom out).

change the size of the viewfinder: dragging outwards increases the size of the viewfinder, and thus decreases the scale of the information space, while dragging inwards decreases the size of the viewfinder and increases the scale of the information space (Fig. 9). The viewfinder is thus similar to a tracking menu (Fitzmaurice et al., 2003), a graphical user interface widget that provides access to different functionalities and that, unlike traditional menus, always stays under the cursor when it is moved.

3.4 Technical comparison

In this section, we compare the three techniques in terms of a set of criteria inspired by those proposed by Harrower and Sheesley (2005) for analyzing approaches to panning and zooming in 2D interactive maps. The criteria we consider are useful to point out strengths and weaknesses of different navigation techniques. The first four criteria (interactive update, sequential versus non-sequential navigation, navigation parameters, orientation cues) are aimed at analyzing the features of a specific technique (functional criteria), while the remaining two criteria (user workload, information-to-controls ratio) are aimed at analyzing how well a technique supports navigation (efficiency criteria).

3.4.1 Interactive update

Interactive update provides visual and temporal correspondence between manipulation of the controls and changes in the information space visualization. When a navigation technique supports interactive update, the visualization changes as pan and zoom controls are used, otherwise the visualization changes only after pan and zoom commands are completed. All three techniques we compared are based on interactively updated panning, which is particularly important to support search tasks within an information space. In the case of zooming, DoubleScrollbar makes no distinction between interactively and non-interactively updated zooming since the user has necessarily to choose a zoom level directly on the menu or with an icon click (and the information space is immediately updated after any of these commands). Technically, it would have been possible to provide interactively updated zooming with Grab&Drag and ZEN, thus providing users with feedback to select the appropriate zoom level during navigation. However, after an initial testing of this feature which did not give good results due to the processing power limitations of the PDA (which is among the most powerful currently available), we decided to actually zoom the considered information space only after users stopped manipulating zoom controls (i.e., when they stopped dragging the slider thumb with Grab&Drag and when they stopped stretching or shrinking the viewfinder with ZEN).

3.4.2 Sequential versus non-sequential navigation

Techniques that allow users to jump instantly to a new zoom level or a new position within an information space support a non-sequential form of navigation, while those that force the user to go through intermediate positions before reaching their desired zoom level or position determine a sequential form of navigation. Usually, if the user knows the structure and content of a specific information space, non-sequential navigation is faster than sequential navigation. However, since knowledge of an information space usually improves only after its extensive examination, both forms of navigation should be provided to users. The three techniques we considered provide sequential panning, since users must navigate through all intermediate points to reach a target destination from a starting position. This is actually desirable in search tasks where the target position is not known a priori. DoubleScrollbar supports non-sequential zooming because users can select a zoom level (on the menu) without passing through all intermediate levels. Grab&Drag and ZEN support instead only sequential zooming since users cannot directly select a new zoom level without moving the zoom control through all intermediate levels.

3.4.3 Navigation parameters

Navigation operations are influenced by various parameters such as the panning rate or the range of available zoom levels. All three techniques we considered employ the same horizontal and vertical panning rates, which determine how much the viewport moves (horizontally and vertically, respectively) as a function of how much the user drags a scrollbar thumb (DoubleScrollbar), the viewport (Grab&Drag) or the viewfinder (ZEN). DoubleScrollbar separates horizontal and vertical scrolling while Grab&Drag and ZEN provide continuous panning in any direction. However, since DoubleScrollbar is based on scrollbars, it actually provides two different scrolling rates: (i) dragging the scrollbar thumb or tapping on the scrollbar arrows enable scrolling at the predefined rate, and (ii) tapping on the scrollbar areas above or below the thumb allows to scroll a larger amount of space (up to the height or width of the viewport for the vertical and horizontal scrollbars respectively). This feature provides users with additional flexibility in navigating an information space. On the other hand, DoubleScrollbar provides only a discrete number of predefined zoom levels while both Grab&Drag and ZEN provide a wider selection of levels, although with the same minimum and maximum values. The minimum level is reached when the whole information space can be displayed in the viewport along its height or width (usually the shortest of the two). The maximum level depends instead on the highest level of detail that can be reached for each map. Finally, DoubleScrollbar and Grab&Drag display the current zoom level as a percentage while ZEN does not provide any explicit indication.

3.4.4 Orientation cues

Orientation cues allow users to understand what portion of an information space is currently displayed in the viewport. In the information visualization literature, the importance of orientation cues to support user navigation of any information space has been widely recognized (Card et al., 1999). Among the three techniques we considered, the level of support to user orientation varies greatly: Grab&Drag does not provide additional orientation cues other than those contained in the information space (e.g., landmarks on maps) while both DoubleScrollbar and ZEN further support user orientation through the position and length of scrollbar thumbs and through the position and size of the viewfinder, respectively.

3.4.5 User workload

User workload concerns the effort required to users by different navigation techniques. While it is difficult to determine workload differences among the considered three techniques without a user study, we can here note that ZEN combines zoom and pan in the same control, possibly requiring less stylus movements compared to the other two techniques that provide separate controls for panning and zooming. Moreover, DoubleScrollbar separates horizontal and vertical scrolling controls, possibly increasing the amount of actions needed to navigate the information space. It is also worth noting that it is slightly more difficult to start panning with DoubleScrollbar and ZEN, which require users to tap on relatively small areas (i.e., scrollbars thumbs and the viewfinder, respectively), than with Grab&Drag, which allows users to tap on any point of the viewport.

3.4.6 Information-to-controls ratio

Information-to-controls ratio, derived from Tufte's data-ink ratio (Tufte, 1983), takes into consideration how much space in the interface is devoted to the controls for panning and zooming compared to the space devoted to display actual content. Since screen space on mobile devices is limited, it is particularly important to employ navigation techniques that maximize the amount of space available for content. Of the three techniques we considered, DoubleScrollbar needs space to display scrollbars and zoom controls. However, since it does not require users to click or drag the actual map, clicking or dragging on the map can be used for other activities, such as selection or editing. Grab&Drag needs space to display the zoom control but is the most unobtrusive for panning. ZEN needs less space than the others in terms of the total number of pixels devoted to the controls but superimposes the controls over the content itself, which might have a possible negative effect on the visibility of the information space.

4 Experimental evaluation

To compare the techniques described in Section 3, we carried out an experimental evaluation that required users to navigate two different types of information space, namely large maps and web pages, on a small screen. In particular, we aimed at testing how the differences in the way the three techniques provide control over zooming and panning (the integration of panning and zooming controls in ZEN, the direct interaction with the information space in Grab&Drag, the familiarity with controls in DoubleScrollbar) would affect user performance in terms of time and number of interface actions in a number of navigation tasks that: a) required users to search for specific targets in two of the most commonly used information spaces on mobile devices (i.e., maps and web pages), b) were realistic and representative of typical navigation activities with those information spaces, c) differed in the amount of navigation effort they required. We also wanted to test if the additional orientation cues provided by DoubleScrollbar and ZEN would enable a better acquisition of spatial knowledge in exploring large maps.

4.1 Participants

The evaluation involved a sample of 20 users (12 M, 8 F). They were undergraduate students or graduates at our university from diverse backgrounds (10 Computer Science, 2 Mathematics, 6 Business Administration, 2 Humanities) in the 21-39 age range (average 27). Most of them (19 out of 20) had never or rarely (1-2 times overall) used a PDA before, while 18 out of 20 were familiar with navigating both maps and web pages on desktop PCs (they used the web regularly as a support to their learning activities and for planning travels). Two users had occasionally used maps on PDAs (1 to 3 times overall). Users were volunteers recruited by direct or email contact.

4.2 Tasks

Each participant was assigned three different types of tasks (15 tasks in total): navigation of web pages (WebTasks), navigation of maps (MapTasks), and spatial memory acquisition (SpatialMemoryTask). The tasks are described in detail in the following:

- WebTasks: navigation of a web page to either identify 3 occurrences of specific words highlighted in the text (WebTask1) or a specific link in the web page (WebTask2). An example of WebTask1 is: "Find 3 occurrences of the words *Mobile Devices* (as consecutives and contained in a same line of text) in the web page and tap on each occurrence". Little navigation effort was required to complete this task, since the occurrences of target words in the web page (they were 5) exceeded the number requested, all occurrences were highlighted using color (as in the snapshots of web pages provided by Google cache) and most of them could be found in the main body text of the web page. An example of WebTask2 is: "Find the link *Pocket PC* in the web page and tap on it". For this task, no highlighting of target words was provided in the web page, thus requiring more navigation effort (with respect to the previous task) to find the target link. Users were asked to perform a WebTask1 and a WebTask2 for each of the techniques considered (6 WebTasks in total).
- MapTasks: navigation of a large city center map to either identify a specific location, such as a street (MapTask1) or the shortest path (in terms of distance to be travelled) between two specific underground stations on the map (MapTask2). An example of MapTask1 is: "You have to go to Rue du Chemin in Paris. Find out where it is and tap on it as soon as you locate it". Every MapTask1 required the user to perform a free exploration of the city center map to identify the target location, which made this task more demanding in terms of navigation than WebTasks. In particular, in

WebTask1 users were facilitated in their search for the target link by the structure of the webpage (since links sections are typically located at the borders of the page) and by the horizontal alignment of text. An example of MapTask2 is: “You are currently located at Rennes underground station in Paris. Indicate the shortest path to reach Parmentier station”. Every MapTask2 required the user to locate the targets on the city center map by looking at the symbols and names of underground stations and then identify the shortest path between the start and destination stations. Users indicated the path by tapping the sequence of underground stations located in the subway line between the two targets. Since in this case the targets to be identified were two and there was also a path to be indicated, this task was more demanding in terms of navigation effort than MapTask1. Users performed a MapTask1 and a MapTask2 for each of the techniques (6 MapTasks in total).

- SpatialMemoryTask: indication of the exact location of two targets previously searched for (one in MapTask1 and one in MapTask2), by tapping their location directly on a map displayed on the PDA. An example of this type of task is: “Point out where Rue du Chemin and Parmentier underground station are located in the city center map of Paris displayed on screen, by tapping on each target”. To perform SpatialMemoryTask, participants could only rely on spatial knowledge previously acquired during map navigation, since no zooming or panning was allowed during this task to read details on the map. Users performed a SpatialMemoryTask after performing the two MapTasks with each of the techniques considered (3 SpatialMemoryTasks in total). The aim of the task was to assess which technique enabled users to develop a better “mental map” of the information space, thus improving performance and speeding up target revisitation (as required for example in MapTask2). More specific revisitation tasks were not included in the evaluation since they are typically used to test effects of factors such as information space distortion (Skopik and Gutwin, 2005), but these effects did not apply to the techniques we considered.

4.3 Materials

During the experiment, participants were provided with a 624Mhz PocketPC with a 3.5” display and QVGA (320x240) resolution. Figure 10 shows samples of the two information spaces used in the experiment: city maps and web pages. Both follow familiar graphic conventions aimed at supporting navigation: for instance, the map uses symbols and color coding to help identify stations, hospitals, monuments and other types of locations; the web page has headings and links sections visually emphasized in the text. For both information spaces, most of these graphic elements were visible at all zoom levels although finer details such as location names, street names or words were not readable at

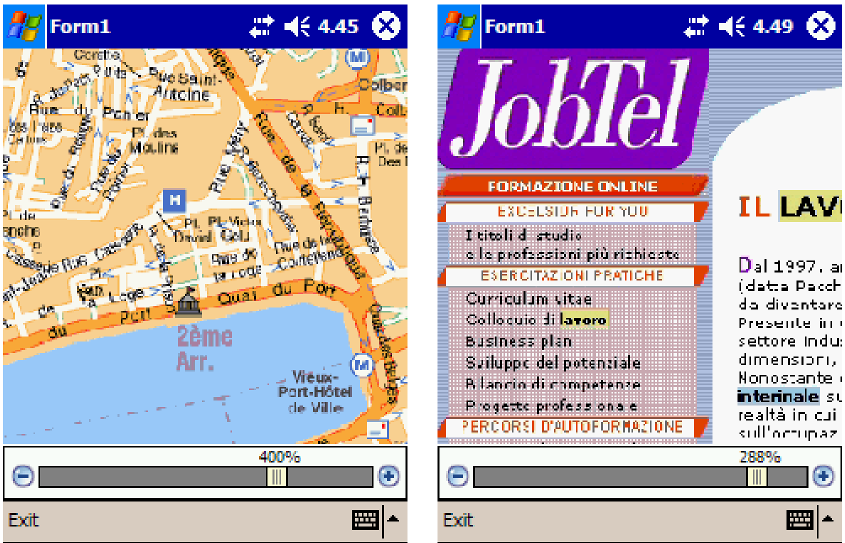


Fig. 10. Samples of the two information spaces considered in our evaluation: maps (left) and web pages (right).

the coarsest levels. The size of all city maps and web pages was very large (32 times the viewport size for maps and 23 times for web pages). The employed maps concerned cities with an underground transportation system located in a country different from the users' one (to avoid familiarity). The language of maps (French) was not the users' mother language (Italian), but is not difficult to read for Italians. We also took care of selecting web pages that were similarly complex in terms of the location of targets and their number of occurrences.

4.4 Procedure

The experimental design was within-subjects. Participants were initially briefed by the experimenter about the nature of the experiment. Then, they were provided with an introduction and demonstration of one of the techniques. This was followed by the presentation of two training tasks (WebTask1 and WebTask2, or MapTask1 and MapTask2) to perform on a training web page or map for familiarizing with the technique on the considered information space. While carrying out the training tasks, participants were allowed to ask any question to the experimenter to clarify possible doubts concerning the technique or the tasks to be carried out. Subsequently, the two experimental tasks were carried out. The same procedure was then followed for the other information space. For each technique, after completing the experimental tasks on maps, users performed the SpatialMemoryTask.

Printed task sheets (containing the description of each task) were presented by the experimenter to participants one at a time, to provide clear instructions for each task and easy reference to target names during task execution. For each task, participants were asked first to read the task description on the printed sheet, then (when ready) to tap on a “Start Task” button that was initially displayed by the PDA, perform the task, and tap the “Exit” command (displayed on the bottom left corner of the PDA screen, as shown in Fig. 10) upon completing it. The order of presentation of the three navigation techniques, as well as their association with tasks and information space (maps and web pages) were counterbalanced to avoid any order effect. At the end of the experiment, participants were briefly interviewed to collect their comments and were asked to tell which visualization they preferred in WebTasks, in MapTasks, and overall.

4.5 Data logging

Logging code automatically recorded the following data for each task carried out by each participant:

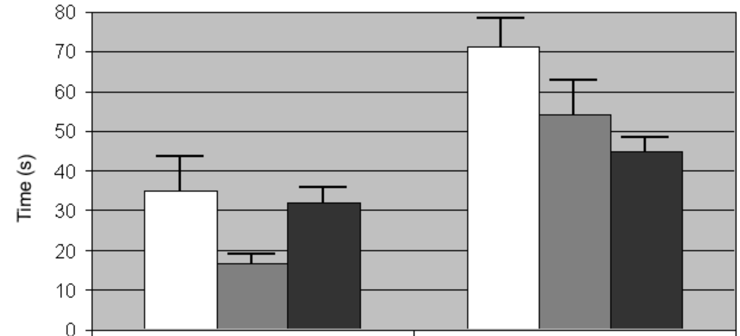
- Task completion time: the time taken to complete the task, defined as the time elapsed between the tap on the “Start Task” and “Exit” buttons.
- User interface actions: the number of distinct user interface actions to carry out the task. A pan action was recorded when users operated the scrollbars in DoubleScrollbar, dragged the stylus on the information space in Grab&Drag, or moved the viewfinder in ZEN. A zoom action was recorded when users selected a zoom level in the menu or operated the zoom buttons in DoubleScrollbar, moved the zoom slider thumb in Grab&Drag or changed the size of the viewfinder in ZEN. A target selection action was recorded when users tapped on targets on the screen.
- Action timings: the start time, end time and duration of each user interface action.
- Accuracy: the distance, in pixels, of the actual target location from the point tapped by the user in the SpatialMemoryTask.

5 Results

Overall, most of the statistically significant differences found by our study among the three techniques were revealed by WebTask1 and MapTask2. These were also the tasks which differed most in the amount of navigation effort required to the user to explore the information space (WebTask1 requiring the least, MapTask2 requiring the most). Specifically, Grab&Drag significantly improved user performance in the task that was less demanding (WebTask1), while DoubleScrollbar and ZEN led to better performance and user orientation in the task requiring a larger amount of navigation of the information space (MapTask2). In the following sections, we provide a detailed description of our findings.

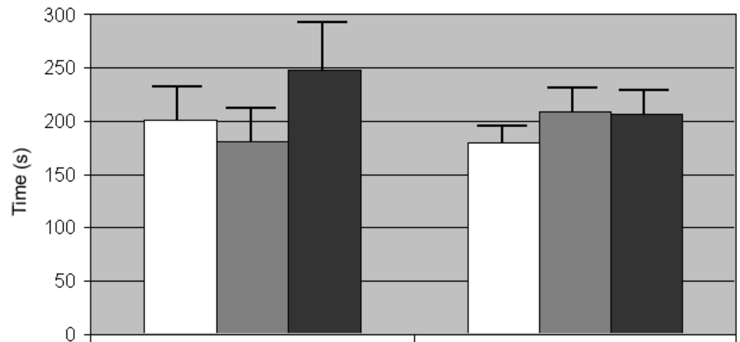
5.1 Task completion times

The time needed by users to complete WebTasks and MapTasks was subjected to the Kolmogorov-Smirnov test of normality prior to further analysis. The test revealed a moderate degree of non-normality due to the presence of outliers. Since these outliers were legitimate values and were not the result of some kind of mistake or mishap, we performed a logarithmic transformation



	WebTask1	WebTask2
DoubleScrollbar	35.09	71.31
Grab&Drag	16.71	53.96
ZEN	31.89	44.9

Fig. 11. Mean completion times for WebTasks. Error bars indicate standard error of the mean.



	MapTask1	MapTask2
DoubleScrollbar	200.2	179.9
Grab&Drag	181.2	208.9
ZEN	247.8	206.1

Fig. 12. Mean completion times for MapTasks. Error bars indicate standard error of the mean.

of the data to reduce their impact and make the distribution more symmetric, as recommended in (Cohen, 2000). Figures 11 and 12 show the mean completion times for WebTasks and MapTasks, respectively. A one-way analysis of variance (ANOVA) was then employed on the log-transformed times. The within-subjects factor was the type of navigation technique with three levels: DoubleScrollbar, Grab&Drag, ZEN. The ANOVA revealed a significant effect only for WebTask1 ($F(2, 57) = 4.92, p < 0.05$). Therefore, we employed the Tukey post-hoc test for comparison among pairs of means, which showed that users spent significantly less time to search for targets with Grab&Drag than they did with ZEN ($q = 4.18, p < 0.05$).

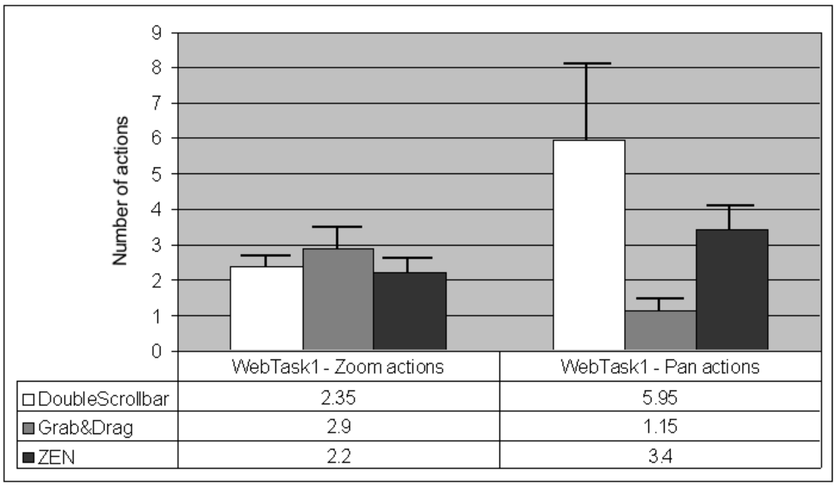


Fig. 13. Mean number of zooming and panning actions for WebTask1. Error bars indicate standard error of the mean.

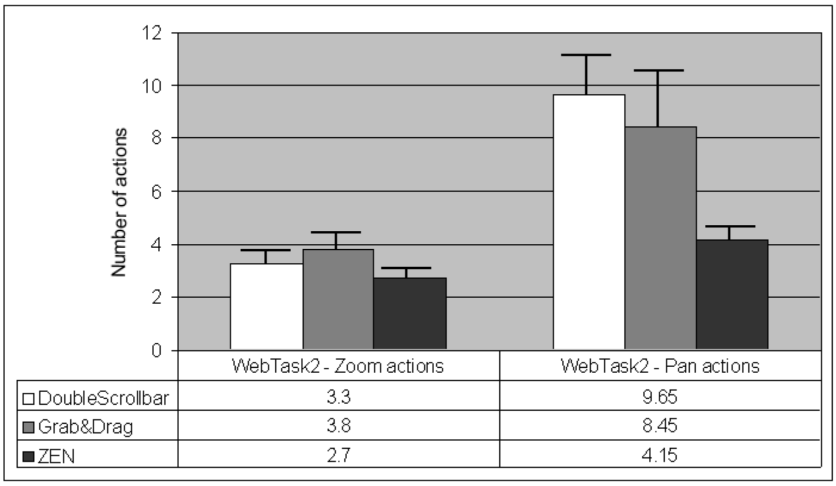


Fig. 14. Mean number of zooming and panning actions for WebTask2. Error bars indicate standard error of the mean.

5.2 User interface actions

We employed Friedman's test to analyze the number of zooming and panning actions users performed to complete each task. Means are shown in Fig. 13 through 16. Most of the differences found in the number of zoom actions were small and none of them statistically significant, although Grab&Drag required more actions in all tasks. On the contrary, differences in the number of pan actions tended to be larger and their analysis revealed a significant effect in WebTask1 ($T = 11.36, p < 0.001$), WebTask2 ($T = 8.95, p < 0.05$) and MapTask2 ($T = 8.33, p < 0.05$). Dunn's Multiple Comparison post-hoc test was then used to compare pairs of means. A statistically significant difference was found for WebTask1 in the number of pan actions between DoubleScroll-

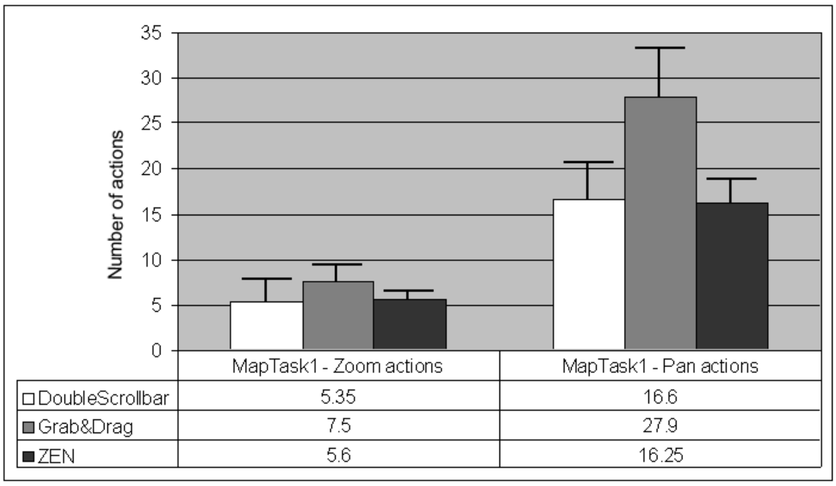


Fig. 15. Mean number of zooming and panning actions for MapTask1. Error bars indicate standard error of the mean.

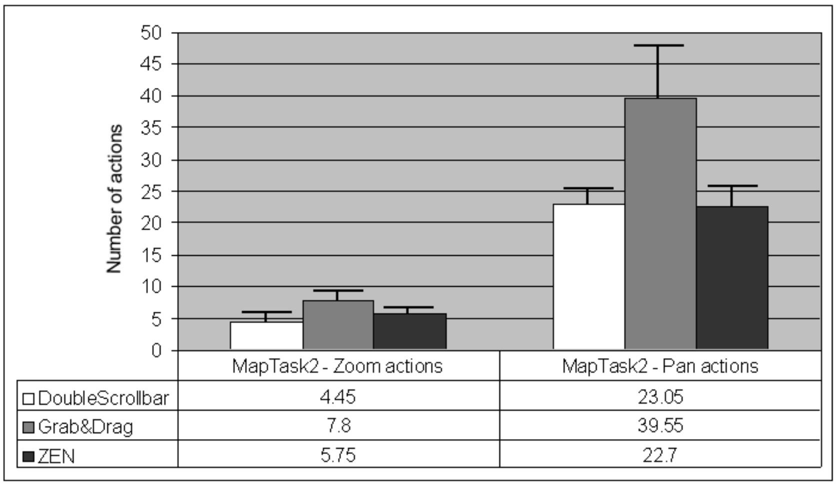


Fig. 16. Mean number of zooming and panning actions for MapTask2. Error bars indicate standard error of the mean.

Bar and Grab&Drag ($p < 0.01$), as well as between Grab&Drag and ZEN ($p < 0.05$) with Grab&Drag requiring less pan actions. However, in the remaining tasks it was ZEN to require less pan actions, and Dunn's post-hoc test showed a significantly lower number of pan actions for ZEN with respect to DoubleScrollBar in WebTask2 ($p < 0.05$) and with respect to Grab&Drag in MapTask2 ($p < 0.05$).

5.3 Action timings

To better understand user's navigation behavior, we also analyzed the mean time required to carry out a zooming or panning action. Zoom action times for

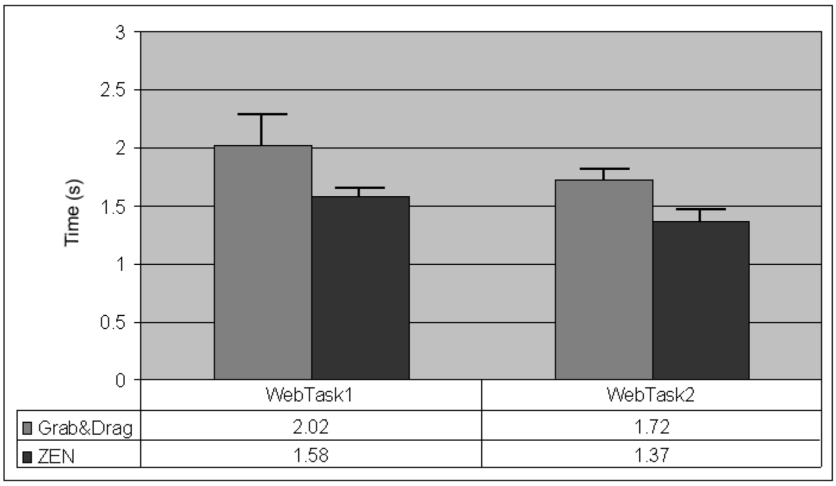


Fig. 17. Mean zoom action times for WebTasks. Error bars indicate standard error of the mean.

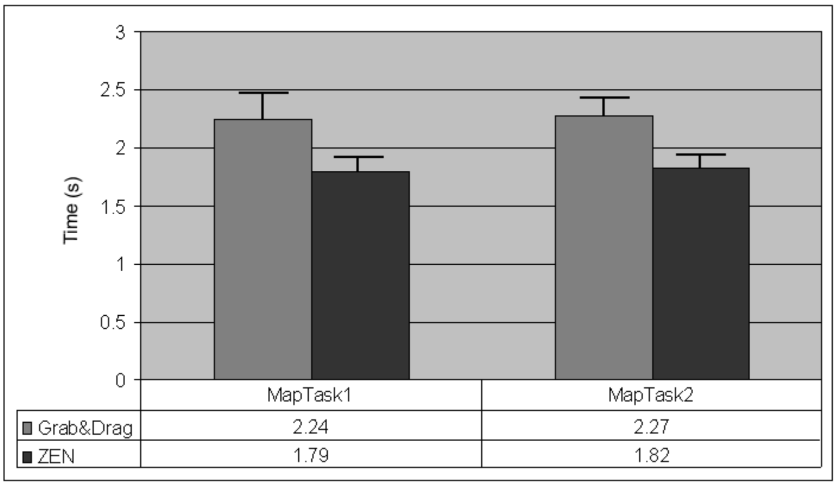


Fig. 18. Mean zoom action times for MapTasks. Error bars indicate standard error of the mean.

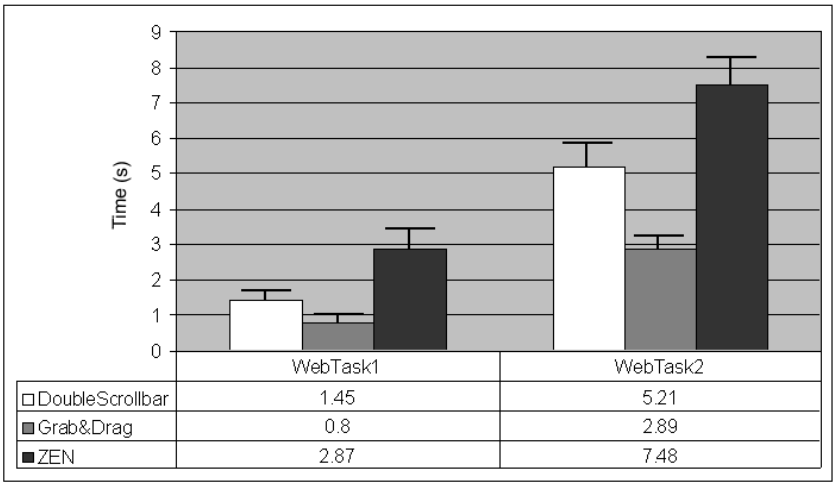


Fig. 19. Mean pan action times for WebTasks. Error bars indicate standard error of the mean.

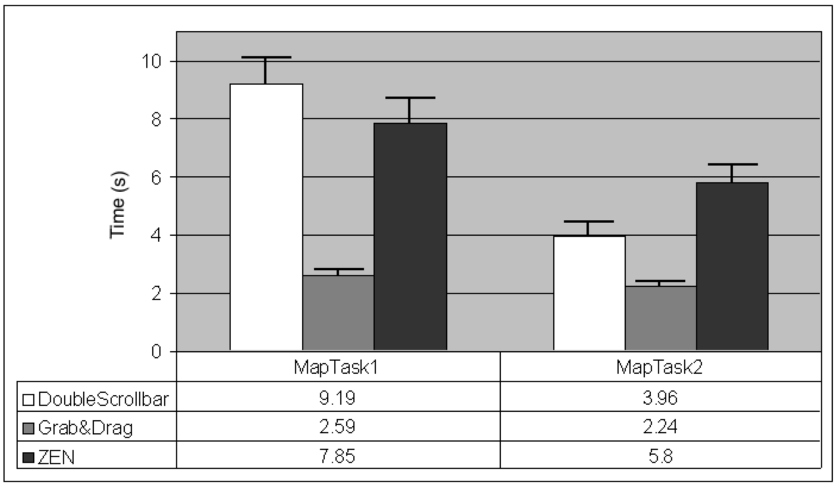


Fig. 20. Mean pan action times for MapTasks. Error bars indicate standard error of the mean.

DoubleScrollBar have been excluded because they had a negligible duration. Means are shown in Fig. 17 through 20.

As with task completion times, the Kolmogorov-Smirnov test revealed a moderate degree of non-normality and we thus performed a logarithmic transformation of the data, as described in section 5.1. T-tests were used to analyze zoom times and ANOVA to analyze pan times. The analysis of zoom times revealed a significant difference between Grab&Drag and ZEN in WebTask2 ($T(2, 19) = 3.19, p < 0.005$) and MapTask2 ($T(2, 19) = 2.65, p < 0.05$), with Grab&Drag requiring longer zoom action times in both cases. For pan action times, the ANOVA pointed out a significant effect in WebTask1 ($F(2, 57) = 7.08, p < 0.01$), WebTask2 ($F(2, 57) = 12.58, p < 0.001$), MapTask1 ($F(2, 57) = 21.34, p < 0.0001$) and MapTask2 ($F(2, 57) = 13.88, p <$

0.001). Tukey's post-hoc test revealed a statistically significant difference between pan action times of Grab&Drag and ZEN in WebTask1 ($q = 5.32, p < 0.01$) and WebTask2 ($q = 7.03, p < 0.001$), as well as between pan action times of Grab&Drag and DoubleScrollBar in WebTask2 ($q = 4.31, p < 0.05$), with Grab&Drag times being shorter. These results combined with those on the number of pan actions are indicative of a navigation strategy generally adopted by users when performing MapTasks with Grab&Drag, that is, a larger number of pan actions of shorter duration, probably required to cope with the lack of additional orientation support provided by the technique. Post-hoc tests also show a statistically significant difference between Grab&Drag and the other techniques in both MapTasks. In particular, pan action times with Grab&Drag were significantly shorter than pan action times with DoubleScrollBar in MapTask1 ($q = 8.88, p < 0.001$) and MapTask2 ($q = 4.80, p < 0.01$), as well as being significantly shorter than pan action times with ZEN in MapTask1 ($q = 6.63, p < 0.001$) and MapTask2 ($q = 7.33, p < 0.001$).

5.4 Errors in the SpatialMemoryTask

Friedman's test was used to analyze the number of errors made by users in the SpatialMemoryTask (Fig. 21), where an error was counted when the Euclidean distance between the target location estimated by the user, and the actual location of the target on the map was higher than a predefined threshold (30 pixels). This threshold marked a sensible area of approximation for an accurate answer, since beyond this area it was very likely to find other (wrong) targets on the map. The analysis pointed out a significant effect ($T = 16.14, p < 0.001$) and the subsequent post-hoc test (Dunn's Multiple Comparisons) revealed a statistically significant difference between Grab&Drag and ZEN ($p < 0.01$)

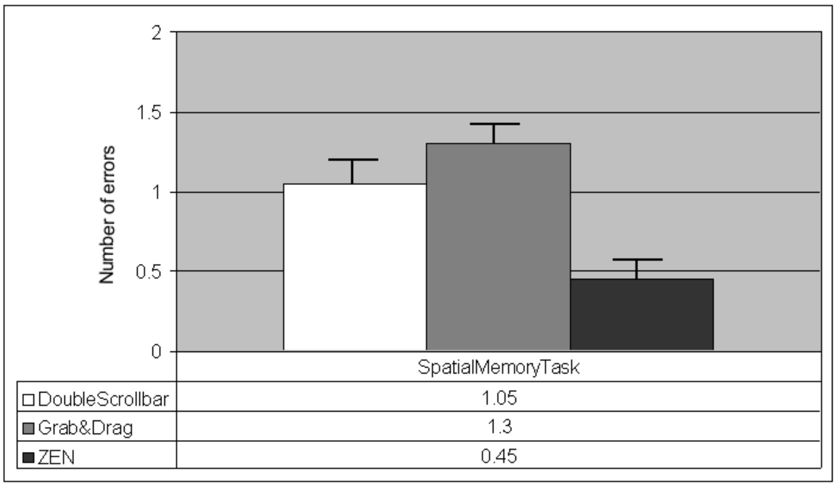
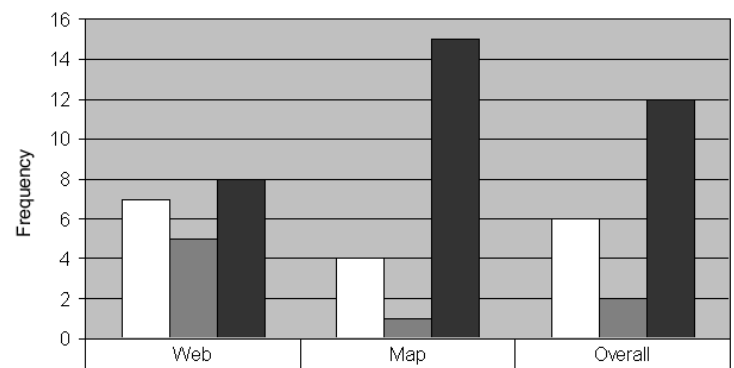


Fig. 21. Mean number of errors in the SpatialMemoryTask. Error bars indicate standard error of the mean.

with participants making less errors with ZEN. The results obtained with the SpatialMemoryTask are consistent with those on user performance in showing that Grab&Drag made it challenging for users to orient themselves within the information space. In addition, they indicate users' difficulties in acquiring accurate spatial knowledge about target locations on maps with Grab&Drag.

5.5 Subjective preference

Figure 22 presents a frequency table of user preference for the three techniques in WebTasks, MapTasks, and overall. For WebTasks, although differences among techniques were small, ZEN and DoubleScrollBar got more preferences than Grab&Drag. According to user comments, the major strength of DoubleScrollBar for WebTasks was the usability and familiarity of scrollbars, which allowed them also to explore the information space without occluding it with the hand or the stylus. The major strength of the ZEN technique consisted in providing clear and constant indication of the area explored within the whole information space, thus improving one's orientation. This aspect was particularly important and appreciated by users when performing MapTasks, where the preference for ZEN was statistically significant ($\chi^2 = 16.29, p < 0.005$). Many participants remarked that exploring map areas by dragging the viewfinder was very intuitive and enabled them to constantly keep track of which portion of the map was currently displayed. Panning by dragging the map without additional orientation support, as in Grab&Drag, turned out to be very challenging for most users, who reported the feeling of being completely lost in the information space during navigation (this is also confirmed by the fact that only 1 user out of 20 expressed preference for Grab&Drag in MapTasks).



	Web	Map	Overall
DoubleScrollbar	7	4	6
Grab&Drag	5	1	2
ZEN	8	15	12

Fig. 22. Frequency table of user preferred technique in WebTasks, MapTasks and overall.

The overall assessment of the three techniques shows a significant preference for ZEN ($\chi^2 = 7.60, p < 0.05$). Users recommended improvements to the layout of ZEN to increase clarity of its zoom controls and visibility of the viewfinder, to better distinguish it from the underlying information space.

6 Discussion and conclusions

In the following, we discuss in more detail our findings following the organization of the technical analysis presented in Section 3.4.

- *Interactive update*. By observing user execution of tasks and through user comments, we found that the lack of interactively updated zooming made ZEN zoom controls less intuitive for users. Indeed, while users found it initially difficult to manipulate the ZEN viewfinder to select the desired zoom level, they took advantage of the textual indication (as a percentage) of the zoom level and benefited from the familiar zoom controls provided by Grab&Drag and DoubleScrollbar. Although this issue did not significantly affect the amount of zoom actions performed by users with the three techniques, it shows that the lack of interactive update may have an effect on the simplicity and usability of techniques with which users are less acquainted.
- *Sequential vs. non-sequential navigation*. The evaluation clarified that although all three techniques provided sequential panning, ZEN (and to a lesser degree DoubleScrollbar) supported the development of a better mental map of the considered information space, as shown by the results of the SpatialMemoryTask. This finding underlines that combining sequential panning with orientation cues has the potential of better supporting spatial knowledge acquisition (as we discuss below). We found that DoubleScrollbar support of non-sequential zooming did not have significant effects on user performance. Moreover, we did not observe any preference for selecting zoom levels by using the zoom menu (non-sequential zooming) versus the zoom in/out icons (sequential zooming) during the tasks.
- *Navigation parameters*. With DoubleScrollbar, users scrolled by dragging the scrollbar thumb or by tapping on the scrollbar arrows and avoided interaction with the scrollbar areas above or below the thumb (that allow to scroll a larger amount of space, as reported in Section 3.4.3). This is probably due to users feeling more comfortable with scrolling an unfamiliar information space at a slow/regular rate. Moreover, the availability of the two different scrolling rates was not particularly prominent at the interface level and users could have simply forgot about it during task execution. Further studies are needed to investigate if this feature can provide benefits as a form of accelerator for expert users. Comments from our participants indicated that the two techniques displaying the zoom level as a percentage (i.e., DoubleScrollbar and Grab&Drag) initially supported an easier under-

standing of their zoom in/out controls, which might suggest the opportunity of introducing this feature also in ZEN.

- *Orientation cues*. Our study highlights that orientation cues are important to effectively navigate an information space. The expectation that the additional cues provided by DoubleScrollbar and ZEN with respect to Grab&Drag would support a better acquisition of spatial knowledge was indeed confirmed. In particular, ZEN was the technique that provided the best user accuracy in the SpatialMemoryTask, probably as a consequence of providing continuous indication of user position in the information space and emphasizing spatial awareness during navigation. The lack of orientation cues provides also a possible explanation for the panning behavior of users with Grab&Drag, characterized by a sequence of many short panning actions to make it easier to keep track of their current position and to maintain orientation within the information space. On the contrary, ZEN and DoubleScrollbar made users feel more confident in navigating longer distances with a single action. This partly contributes to confirm our expectation that orientation would have not been problematic with these two techniques, especially with ZEN (where users could refer to the viewfinder to derive their current position). Moreover, the benefits brought by the additional orientation support provided by ZEN explain the large preference of users for ZEN in MapTasks.
- *User workload*. Our findings show that having the pan control located directly on the information space (as in Grab&Drag and ZEN) allows users to be faster in tasks requiring a limited amount of navigation (such as WebTasks, and in particular WebTask1). This is also consistent with the fact that Grab&Drag outperformed DoubleScrollbar in requiring less panning actions in WebTask1, while ZEN required less panning actions than DoubleScrollbar in WebTask2. However, the availability of the pan control on the information space is not sufficient to ensure efficient performance in tasks (as MapTasks) requiring more navigation effort: in this case, it is the support provided by orientation cues that makes the difference. Indeed, in MapTask2 users had to perform more panning actions with Grab&Drag than with ZEN to find the targets.

The experimental evaluation pointed out that the integration of zoom and pan controls in ZEN did not provide clear benefits in terms of user workload. Moreover, it was observed that users did have difficulties in starting pan actions with ZEN due to the need of tapping on precise areas to control the viewfinder. Participants also commented that reading map details such as street or underground station names was at times more difficult with ZEN because the user's hand dragging the viewfinder reduced visibility of the areas inspected. Moreover, text on maps was rarely aligned horizontally, which made reading rather challenging on a small screen with all three techniques. We think this issue deserves further investigation, e.g. by testing refined versions of pan-zoom controls for ZEN, as well as by extending the range of information spaces and navigation tasks considered.

As reported in Section 5.2, the comparative analysis of the number of interface actions did not reveal significant differences in the number of zoom actions performed with the three techniques. Users did not manipulate the zoom level much, generally preferring to navigate the information space with several panning actions. This is probably related to the nature of the tasks studied and the large size of the employed information spaces, which led users to typically adopt the following navigation behavior: a) initially get an overview of the space to navigate, b) select an appropriate zoom level to make text readable, c) go through a series of panning actions seldom interleaved with zoom level changes, until targets are reached. The limited use of zoom controls is also reflected by the mean time of zoom actions, which was very similar across all tasks. However, it is interesting to note that the mean time of zoom actions with Grab&Drag was higher than the mean time with ZEN regardless of the task, with two tasks out of four reaching a significant difference. Such a result can be partially explained by noting that operating the zoom slider in Grab&Drag usually required users to move the slider thumb for a longer distance than that required to change the size of the viewfinder, given the same initial and final zoom levels.

- *Information-to-control ratio*. Between the two techniques that presented the best information-to-control ratio (ZEN and Grab&Drag), users largely preferred ZEN, although we observed that ZEN viewfinder controls hampered visibility of the information space, especially while users were getting acquainted with the new technique. However, future refinements of the viewfinder layout and increased user practice with ZEN controls might alleviate this problem, as well as show stronger evidence of the benefits produced by increasing the space for content on small screens.

To further improve our understanding of user behavior and clarify the causes of navigation problems, we are currently working at refining a tool (Burigat et al., 2008) that provides analysts with a set of visualizations aimed at highlighting different aspects of how users navigate an information space. The visualizations are aimed at highlighting the spatio-temporal evolution of specific navigation parameters (e.g., the zoom level or panning speed) for each individual navigation session.

In future studies, we will also focus on the calibration of important parameters of navigation techniques, such as pan/scroll rates and zoom levels. Indeed, a change of these settings may impact upon user performance and produce different results from the ones we have obtained. In particular, we think it would be worth investigating if the choice of different zoom levels according to the features of the information space (e.g, size, complexity), may produce larger differences among techniques in the amount of zoom actions performed by users. We will also study different designs of ZEN controls to find a solution that is intuitive for users and that is less sensible to scalability issues (since the size of the viewfinder shrinks as the zoom level increases). Moreover, since

ZEN is a novel technique, it would be interesting to carry out a longitudinal study to determine if a long-term use of the technique could result in better user performance with respect to commonly used navigation techniques.

7 Acknowledgments

This work has been partially supported by the Italian Ministry of Education, University and Research (MIUR) under the PRIN 2005 Project "Adaptive, Context-aware, Multimedia Guides on Mobile Devices".

References

Baudisch, P., Good, N., Bellotti, V. and Schraedley, P. (2002), Keeping Things in Context: a Comparative Evaluation of Focus Plus Context Screens, Overviews, and Zooming, *in* 'Proc. Conference on Human Factors in Computing Systems (CHI 2002)', ACM Press, pp. 259–266.

Baudisch, P. and Rosenholtz, R. (2003), Halo: a Technique for Visualizing Off-Screen Locations, *in* 'Proc. Conference on Human Factors in Computing Systems (CHI 2003)', ACM Press, pp. 481–488.

Bederson, B., Clamage, A., Czerwinski, M. and Robertson, G. (2004), 'DateLens: a Fisheye Calendar Interface for PDAs', *ACM Transactions on Computer-Human Interaction* **11**(1), 90–119.

Bederson, B. and Hollan, J. (1994), Pad++: a Zooming Graphical Interface for Exploring Alternate Interface Physics, *in* 'Proc. Conference on User Interface Software and Technology (UIST 1994)', ACM Press, pp. 17–26.

Björk, S., Holmquist, L., Redström, J., Bretan, I., Danielsson, R., Karlgren, J. and Franzèn, K. (1999), WEST: a Web Browser for Small Terminals, *in* 'Proc. Conference on User Interface Software and Technology (UIST 1999)', ACM Press, pp. 187–196.

Burigat, S., Chittaro, L. and Gabrielli, S. (2006), Visualizing Locations of Off-Screen Objects on Mobile Devices: a Comparative Evaluation of Three Approaches, *in* 'Proc. Conference on Human-Computer Interaction with Mobile Devices and Services (Mobile HCI 2006)', ACM Press, pp. 239–246.

Burigat, S., Chittaro, L. and Ieronutti, L. (2008), 'Exploring Mobile Browsing through Interactive Visualizations of User Behavior', *IEEE Computer Graphics and Applications*, to appear.

Buring, T., Gerken, J. and Reiterer, H. (2006*a*), Usability of Overview-supported Zooming on Small Screens with Regard to Individual Differences in Spatial

Ability, *in* 'Proc. Conference on Advanced Visual Interfaces (AVI 2006)', ACM Press, pp. 233–240.

Buring, T., Gerken, J. and Reiterer, H. (2006*b*), 'User Interaction with Scatterplots on Small Screens - A Comparative Evaluation of Geometric-Semantic Zoom and Fisheye Distortion', *IEEE Transactions on Visualization and Computer Graphics* **12**(5), 829–836.

Buyukkokten, O., Garcia-Molina, H., Paepcke, A. and Winograd, T. (2000), Power Browser: Efficient Web Browsing for PDAs, *in* 'Proc. Conference on Human Factors in Computing Systems (CHI 2000)', ACM Press, pp. 430–437.

Card, S., Mackinlay, J. and Shneiderman, B. (1999), *Readings in Information Visualization*, Morgan Kaufmann.

Chen, Y., Ma, W. and Zhang, H. (2003), Detecting Web Page Structure for Adaptive Viewing on Small Form Factor Devices, *in* 'Proc. World Wide Web Conference (WWW 2003)', ACM Press, pp. 225–233.

Chittaro, L. (2006), 'Visualizing Information on Mobile Devices', *IEEE Computer* **39**(3), 40–45.

Cockburn, A. and Savage, J. (2003), Comparing Speed-dependent Automatic Zooming with Traditional Scroll, Pan and Zoom Methods, *in* 'Proc. British Conference on Human Computer Interaction (HCI 2003)', pp. 87–102.

Cohen, B. (2000), *Explaining Psychological Statistics*, Wiley.

Fitzmaurice, G., Khan, A., Pieke, R., Buxton, B. and Kurtenbach, G. (2003), Tracking Menus, *in* 'Proc. Conference on User Interface Software and Technology (UIST 2003)', ACM Press, pp. 71–79.

Furnas, G. W. (1986), Generalized Fisheye Views, *in* 'Proc. Conference on Human Factors in Computing Systems (CHI 86)', ACM Press, pp. 16–23.

Gutwin, C. and Fedak, C. (2004), Interacting with Big Interfaces on Small Screens: a Comparison of Fisheye, Zoom, and Panning Techniques, *in* 'Proc. Conference on Graphics Interface (GI 2004)', Canadian Human-Computer Communications Society, pp. 145–152.

Harrie, L., Sarjakoski, L. T. and Lehto, L. (2002), 'A Mapping Function for Variable-Scale Maps in Small-Display Cartography', *Journal of Geospatial Engineering* **2**(3), 111–123.

Harrower, M. and Sheesley, B. (2005), 'Designing Better Map Interfaces: a Framework for Panning and Zooming', *Transactions in GIS* **9**(2), 77–89.

Hornbaek, K. and Frokjaer, E. (2001), Reading of Electronic Documents: The Usability of Linear, Fisheye, and Overview+Detail Interfaces, *in* 'Proc. Conference on Human Factors in Computing Systems (CHI 2001)', ACM Press, pp. 293–300.

Igarashi, T. and Hinckley, K. (2000), Speed-dependent Automatic Zooming for Browsing Large Documents, *in* 'Proc. Conference on User Interface Software and Technology (UIST 2000)', ACM Press, pp. 139–148.

Irani, P., Gutwin, C. and Yang, X. (2006), Improving Selection of Off-screen Targets with Hopping, *in* 'Proc. Conference on Human Factors in Computing Systems (CHI 2006)', ACM Press, pp. 299–308.

Jacobs, C., Li, W., Schrier, E., Bargeron, D. and Salesin, D. (2003), Adaptive Grid-Based Document Layout, *in* 'Proc. Conference on Computer Graphics and Interactive Techniques (SIGGRAPH 2003)', ACM Press, pp. 838–847.

Jones, S., Jones, M., Marsden, G., Patel, D. and Cockburn, A. (2005), 'An Evaluation of Integrated Zooming and Scrolling on Small Screens', *International Journal of Human-Computer Studies* **63**(3), 271–303.

Kaptelinin, V. (1995), A Comparison of Four Navigation Techniques in a 2D Browsing Task, *in* 'Proc. Conference on Human Factors in Computing Systems (CHI 1995)', ACM Press, pp. 282–283.

Karstens, B., Rosenbaum, R. and Schumann, H. (2004), Presenting Large and Complex Information sets on Mobile Handhelds, *in* 'E-Commerce and M-Commerce Technologies', IRM Press, Hershey, London, pp. 32–56.

Lam, H. and Baudisch, P. (2005), Summary Thumbnails: Readable Overviews for Small Screen Web Browsers, *in* 'Proc. Conference on Human Factors in Computing Systems (CHI 2005)', ACM Press, pp. 681–690.

Mackinlay, J., Good, L., Zellweger, P., Stefik, M. and Baudisch, P. (2003), City Lights: Contextual Views in Minimal Space, *in* 'Proc. ACM CHI Conference on Human Factors in Computing Systems (CHI '03)', ACM Press, pp. 838–839.

Nekrasovski, D., Bodnar, A., McGrenere, J., Guimbretiere, F. and Munzner, T. (2006), An Evaluation of Pan&zoom and Rubber Sheet Navigation with and without an Overview, *in* 'Proc. Conference on Human Factors in Computing Systems (CHI 2006)', ACM Press, pp. 11–20.

Plaisant, C., Carr, D. and Shneiderman, B. (1995), 'Image-Browser Taxonomy and Guidelines for Designers', *IEEE Software* **12**(2), 21–32.

Pook, S., Lecolinet, E., Vaysseix, G. and Barillot, E. (2000), Context and Interaction in Zoomable User Interfaces, *in* 'Proc. International Conference on Advanced Visual Interfaces (AVI '00)', ACM Press, pp. 227–231.

Rauschenbach, U., Jeschke, S. and Schumann, H. (2001), 'General Rectangular FishEye Views for 2D Graphics', *Computers and Graphics* **25**(4), 609–617.

Robbins, D., Cutrell, E., Sarin, R. and Horvitz, E. (2005), ZoneZoom: Map Navigation for Smartphones with Recursive View Segmentation, *in* 'Proc. International Conference on Advanced visual interfaces (AVI 2004)', ACM Press, pp. 231–234.

Rosenbaum, R. and Schumann, H. (2005), Grid-based Interaction for Effective Image Browsing on Mobile Devices, *in* 'Proc. of the SPIE, Volume 5684', SPIE - The International Society for Optical Engineering, pp. 170–180.

Roto, V., Popescu, A., Koivisto, A. and Vartiainen, E. (2006), Minimap: a Web Page Visualization Method for Mobile Phones, *in* 'Proc. Conference on Human Factors in Computing Systems (CHI 2006)', ACM Press, pp. 35–44.

Sarkar, M., Snibbe, S., Tversky, O. and Reiss, S. (1993), Stretching the Rubber Sheet: a Metaphor for Viewing Large Layouts on Small Screens, *in* 'Proc. Conference on User Interface Software and Technology (UIST 93)', ACM Press, pp. 81–91.

Skopik, A. and Gutwin, C. (2005), Improving Revisitation in Fisheye Views with Visit Wear, *in* 'Proc. Conference on Human Factors in Computing Systems (CHI 2005)', ACM Press, pp. 771–780.

Trevor, J., Hilbert, D. M., Schilit, B. N. and Koh, T. (2001), From Desktop to Phonetop: a UI for Web Interaction on Very Small Devices, *in* 'Proc. Conference on User Interface Software and Technology (UIST 2001)', ACM Press, pp. 121–130.

Tufte, E. (1983), *The Visual Display of Quantitative Information*, Graphics Press.

Zipf, A. and Richter, K.-F. (2002), 'Using Focus Maps to Ease Map Reading: Developing Smart Applications for Mobile Devices', *Kunstliche Intelligenz* **4**, 35–37.