Dynamic Visualization of Large Numbers of Off-screen Objects on Mobile Devices: an Experimental Comparison of Wedge and Overview+Detail

Stefano Burigat
stefano.burigat@uniud.it

Luca Chittaro
luca.chittaro@uniud.it

Andrea Vianello
andrea.vianello@uniud.it

HCI Lab
Department of Mathematics and Computer Science
University of Udine
Via delle Scienze 206, 33100, Udine, Italy

ABSTRACT

Overview+Detail [25] and Wedge [16] have been proposed in the literature as effective approaches to resolve the off-screen objects problem on mobile devices. However, they have been studied with a small number of off-screen objects and (in most studies) with static scenarios, in which users did not have to perform any navigation activity. In this paper, we propose improvements to Wedge and Overview+Detail which are specifically aimed at simplifying their use in dynamic scenarios that involve large numbers of off-screen objects. We compare the effectiveness of the two approaches in the considered scenario with a user study, whose results show that Overview+Detail allows users to be faster in searching for off-screen objects and more accurate in estimating their location.

Author Keywords

Visualization; mobile devices; map navigation; peripheral awareness; off-screen objects; Overview+Detail; Wedge

ACM Classification Keywords

H.5.2 Information Interfaces and Presentation: User Interfaces—*Evaluation/methodology, Graphical user interfaces (GUI)*

INTRODUCTION

People increasingly use mobile devices to display large information spaces such as web pages or maps. Unfortunately, the small screen of mobile devices greatly increases the complexity of exploring these information spaces at an adequate level of detail, forcing users to carry out multiple zooming and panning actions [5]. This could be problematic, for example, when the user of a mobile map application (e.g. Google Maps) wants to know the location of objects of interest around her, such as banks, restaurants or hotels. Due to the limited size of displays, the zooming and panning actions needed to look for this information could make users lose the global context when examining the details of the visualization and vice versa [12]. As a consequence, the user may not be aware of the presence of relevant objects located off-screen (the *off-screen objects problem*) and thus may not be able to reach them.

Overview+Detail (hereinafter, O+D) and Contextual Cues Techniques are the two approaches that have been explored most in the literature to mitigate the off-screen objects problem. O+D provides both detail and context information by typically displaying two separate views simultaneously, one for the context and one for the detail [25]. With this approach, the overview can be used to highlight all objects of interest that are outside the detail view. Contextual Cues Techniques, such as Halo [3] or Wedge [16], augment the detail view with abstract shapes (or *proxies*), displayed near the border of the screen, to make users aware of relevant objects that are outside the view area. In particular, Wedge exploits isosceles triangles (called *wedges*) to point at off-screen objects: for each off-screen object, the tip of a triangle is made to coincide with the object, while the base as well as part of the two legs of the triangle are visible on the screen. By mentally extending the visible part of the legs, users should be able to estimate the position of the corresponding off-screen object (see Figure 1).

Several user studies have been carried out to evaluate the effectiveness of O+D and Contextual Cues Techniques as solutions to the off-screen objects problem. These studies revealed that both approaches are effective in helping users to carry out spatial tasks with off-screen objects. In particular, the effectiveness of Wedge stands out among Contextual Cues Techniques. However, these studies had limitations that make it difficult to generalize their results to typical real-world usage scenarios. Most studies involved spatial tasks, such as finding the closest off-screen object, that required users to only look at a static configuration of off-screen objects, thus neglecting the possible effect of panning and zooming actions on user's ability to make sense of the visualization. Moreover, all studies involved a small number of off-screen objects (typically less than 10), while real-world scenarios often involve larger numbers. Since there is evidence that even a small increase in the number of off-screen objects negatively affects user performance in spatial tasks [7], the

MobileHCI'12, September 21–24, 2012, San Francisco, CA, USA.

effectiveness of the two approaches with larger numbers of objects is dubious.

The aim of our work is to propose improvements to O+D and to the most effective Contextual Cues Technique (Wedge) to simplify their use in dynamic scenarios with large numbers of off-screen objects whose relative position to the border of the screen changes as a consequence of user's navigation actions. With a user study, we investigate whether the two modified approaches are as effective in a complex and dynamic scenario as they are in the static scenarios used in past studies.

The paper is organized as follows. Section 2 surveys related work. Section 3 discusses in more detail the motivations for our work. Section 4 describes the two approaches we compared in the study. Section 5 describes the experimental evaluation and reports results, which are then discussed in Section 6.

RELATED WORK

In this section, we focus on O+D and Contextual Cues Techniques, illustrating how these approaches aim to solve the off-screen objects problem and highlighting their strengths and weaknesses.

Overview+Detail

O+D typically provides two simultaneous views of an information space: an overview of the whole space and a detail view of a specific portion of the space. In the overview, a graphical highlight (the *viewfinder*) shows which is the portion of space displayed by the detail view [25]. The two views can support zooming and panning and they can be *tightly coupled*, if the manipulation of one view is immediately reflected in the other view, or *loosely coupled* if the manipulation of one view results in an update of the other view only when users complete their panning or zooming actions. Studies in the desktop HCI literature show that the overview allows users to be faster in performing search tasks in an information space compared to a scrolling interface [2, 4, 24] and can help users to acquire information during navigation [21]. However, O+D seems to be worse than zooming and Focus+Context interfaces in terms of task completion times [1, 18, 20, 21]. Nonetheless, O+D was almost always preferred by users, probably because it helps users to build a better mental model of the explored information space [23].

On mobile devices, O+D can be problematic due to the small size of displays: information in the overview might be difficult to read and the overview may cover a sizable part of the detail view. Moreover, it could be complex for users to relate the two views [12]. The reduced size of the overview can also negatively affect interaction, making it more difficult for users to manipulate the viewfinder. Additionally, a small panning action on the overview can result in large jumps in the detail view.

Few empirical studies have been carried out on mobile O+D. In their work on web page visualization, Roto et al. [26] proposed a browser that automatically reformats page content and, during scrolling, displays a small overview containing a thumbnail of the whole page. Users found the proposed browser more usable compared to a traditional mobile browser. In [10], O+D was compared with a detail-only zooming interface, in a study where users had to perform search tasks in a scatterplot using a PDA. Users with high spatial ability were faster when using the zooming interface, suggesting that the use of a single larger detail view on mobile devices can be preferable to an O+D solution. Burigat et al. [9] compared two different implementations of O+D and a zooming interface, in a study where users had to navigate maps, diagrams and web pages. The two O+D solutions differed in terms of the information included in the overview: one showed only the outline of overview and viewfinder while the other showed a thumbnail of the considered information space. Results reveal that a small thumbnail of the information space in the overview helps users to be faster in carrying out search tasks on maps. Moreover, O+D helps users to recall the position of objects in the considered information space. In a subsequent study [6], Burigat and Chittaro compared different implementations of O+D, to better understand the effects of highlighting objects of interest in the overview and the possibility to directly manipulate the viewfinder. Results show that both factors are beneficial to users in map search tasks, with manipulability providing the highest performance improvement.

Contextual Cues Techniques

Contextual Cues Techniques augment the detail view with abstract shapes placed near the border of the screen to provide awareness of off-screen objects.

City Lights [28], one of the first Contextual Cues Techniques proposed in the literature, encodes off-screen objects with lines displayed at the border of the screen. The lines correspond to the orthogonal projection of objects on the edges of the display. Moreover, line color encodes the distance of objects. EdgeRadar [17] represents off-screen objects by using points, placed into narrow areas all along the edges of the screen. Points which are closer to the border represent farther off-screen objects. Scaled Arrows and Stretched Arrows are two other Contextual Cues Techniques proposed in [8]. In Scaled Arrows, the orientation of arrows placed along the border encodes the direction of off-screen objects while the size of arrows is inversely proportional to the distance of the objects from the edges of the screen. Stretched Arrows differ from Scaled Arrows by encoding distance with the length of arrows. In both techniques, larger arrows correspond to closer objects.

Unlike the above mentioned solutions, Halo [3] is a Contextual Cues Technique that surrounds every off-screen object with a circle large enough to enter the display area, exploiting the human ability to visually complete a circle by looking at only a portion of it, to derive position and distance of off-screen objects. Baudisch et al. [3] found that users were faster in carrying out spatial tasks on a map with Halo, compared to a technique that encodes off-screen objects with static arrows and labels for distance indication. In [8], Halo was compared with Scaled Arrows and Stretched

Arrows, in a study where users had to carry out spatial tasks on a map. Halo scored better in distance estimation, while Scaled Arrows scored better in user preference and in the task that required the relative distance order of off-screen objects. Moreover, users were faster and more accurate with the two arrow approaches when the number of off-screen objects increased. The comparison between Halo and EdgeRadar in [17] revealed that EdgeRadar is preferable to help users keep track of moving off-screen objects. In [22], the authors propose an approach based on Halo, which uses oval halos in order to reduce overlap among off-screen proxies. However, the distortion negatively affected distance estimation and user accuracy in locating off-screen objects. Another Halo variant, HaloDot [15], adds a small dot at the intersection between a circle and the intrusion border, i.e. the inner limit of the area where halos are visible, to improve direction awareness of off-screen objects. Moreover, the authors use color and transparency to encode the relevance of off-screen objects and use an aggregation approach to mitigate clutter among proxies. The results of a comparison with Halo pointed out that HaloDot allows users to search faster for relevant objects of interest and that the aggregation is useful.

As mentioned above, Wedge [16] is a Contextual Cues Technique that exploits triangles (*wedges*) to make users aware of off-screen objects. With Wedge, users were more accurate in identifying the location of off-screen objects compared to using Halo, especially when the number of objects was larger, likely because wedges take up less space on the border of the screen compared to Halo. A study described in [7] compared Wedge with Scaled Arrows and O+D. In the O+D condition, off-screen objects were represented by points in the overview. Results pointed out that O+D was more effective when users had to reason in terms of the spatial configuration of off-screen objects, such as finding the pair of objects which are closest to each other. However, Wedge was more effective when only distance information of the off-screen objects was important. When users had to identify the correct location of off-screen objects, no significant differences were found among the three approaches. Finally, results showed that with a slightly larger number of objects user performance was negatively affected with all approaches.

MOTIVATIONS OF THE STUDY

In general, the previously summarized studies pointed out that Wedge and O+D are more effective than other approaches in helping users to carry out spatial tasks in large information spaces on mobile devices. However, most of these studies had important limitations in terms of the employed scenarios. More specifically:

- Most tasks involved a relatively small number of off-screen objects. In [3], Halo was compared with the arrow-based technique employing 5 off-screen objects. In the comparison among Halo, Scaled Arrows and Stretched Arrows described in [8] the number of objects ranged from 5 to 8. The same number of objects was used in [7], where Wedge was compared with Scaled Arrows and O+D. In [16], the comparison between Wedge and Halo involved 5 objects even if the density of these objects on the screen was set to simulate more cluttered conditions.
- Almost all of the tasks involved only static scenarios, i.e., users were presented with a predefined configuration of off-screen objects and never had to perform any navigation activity in the considered information space.
- Some of the studies (e.g., [3, 16]) were carried out only on mobile device emulators running on a desktop computer with a mouse rather than on actual mobile devices.

As a consequence of these limitations, results of studies on the off-screen objects problem cannot be easily generalized to more complex and dynamic scenarios. As an example of dynamic scenario, consider a user who needs to know the position of underground stations or bus stops, located around her, in order to reach a destination. In the center of a big city, the number of these objects of interest is likely to be very high and in this case, by looking at the map at the zoom level required to see map details, a lot of these objects of interest might be outside the screen. Thus, the user has to perform panning actions to see where the desired objects of interest are located. This might have a significant impact on Contextual Cues Techniques since the relative position of all off-screen objects with respect to the border of the screen changes. Finally, with a large number of off-screen objects, which is a realistic scenario when dealing with e.g. underground stations or bus stops, the user could find it difficult to make sense of the visualization and identify the most relevant objects of interest for her purposes.

Our work was thus aimed at studying Wedge and O+D in dynamic scenarios with large numbers of off-screen objects, proposing possible improvements to the two approaches aimed at making them more effective in such scenarios and, eventually, understanding whether the effectiveness of Wedge and O+D in static scenarios with low numbers of off-screen objects would transfer to more complex scenarios.

THE CONSIDERED APPROACHES

In this section, we describe the two approaches we compared in our study, illustrating the changes we made to Wedge and O+D to improve their use in dynamic scenarios with large numbers of objects.

As discussed in previous sections, Wedge is based on local models of amodal perception which suggest that the human visual system completes the occluded part of an object by connecting the extension of visible contours [27], which, in this case, are the visible part of the legs of wedges associated to off-screen objects (Fig. 1). The base of each wedge is a key graphical element that helps users to discriminate among legs belonging to different triangles. A key feature of Wedge is that triangles have three degrees of freedom since it is possible to change rotation, aperture and intrusion of each triangle while keeping it pointed at the same location. These features are important to improve location accuracy and to display wedges at the corners of the screen. In Wedge, the degrees of freedom are also used by an iterative overlap resolution algorithm.

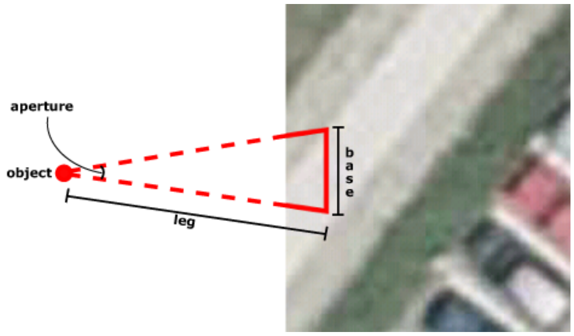


Figure 1. Each wedge consists of two legs and a base. Part of the two legs and the base are visible on the screen while the tip of the wedge coincides with the off-screen object.

After implementing Wedge on mobile devices, we examined the effect of increasing the number of off-screen objects. We started with a configuration of 20 off-screen objects randomly positioned around the visualized portion of a map. Figure 2 shows the resulting visualization with (Fig. 2a) and without (Fig. 2b) activation of the overlap resolution algorithm. Even with such a relatively low number of off-screen objects, making sense of the visualization in a dynamic scenario might be difficult, especially in the areas where the density of wedges is higher. It must also be noted that the effect of the overlap resolution algorithm, while noticeable, does not sensibly improve the visualization, mainly because several overlaps cannot be resolved. Setting the number of off-screen objects to 30, we obtained visualizations such as those in Fig. 2c (with overlap resolution) and Fig. 2d (without overlap resolution). As it can be seen, clutter greatly increases with respect to the 20 objects configuration and the overlap resolution algorithm does not improve the visualization. For this reason, we did not use the (computationally expensive) overlap resolution algorithm during the user study.

From these examples, it is clear that the largest triangles have a significant impact on the readability of the visualization since they take more screen space along the border of the screen and create several overlaps with smaller wedges. Larger wedges are associated to off-screen objects which are farther from the visualized area and in many real-world scenarios are less important for users. For this reason, we extended Wedge with two mechanisms that reduce the visibility of the largest wedges to simplify the visualization and reduce clutter.

The first mechanism acts on the transparency of wedges, changing it according to the distance of the corresponding off-screen objects. The farther the off-screen object is from the displayed area, the higher the transparency level of the corresponding wedge. The mapping between distance and transparency is based on two distance thresholds. All wedges associated to off-screen objects which are farther than a maximum threshold will use a predefined high level of transparency. Wedges associated to objects which are closer than a minimum threshold will be displayed with 100% opacity. All other wedges will be displayed with a level of transparency directly proportional to the distance of the

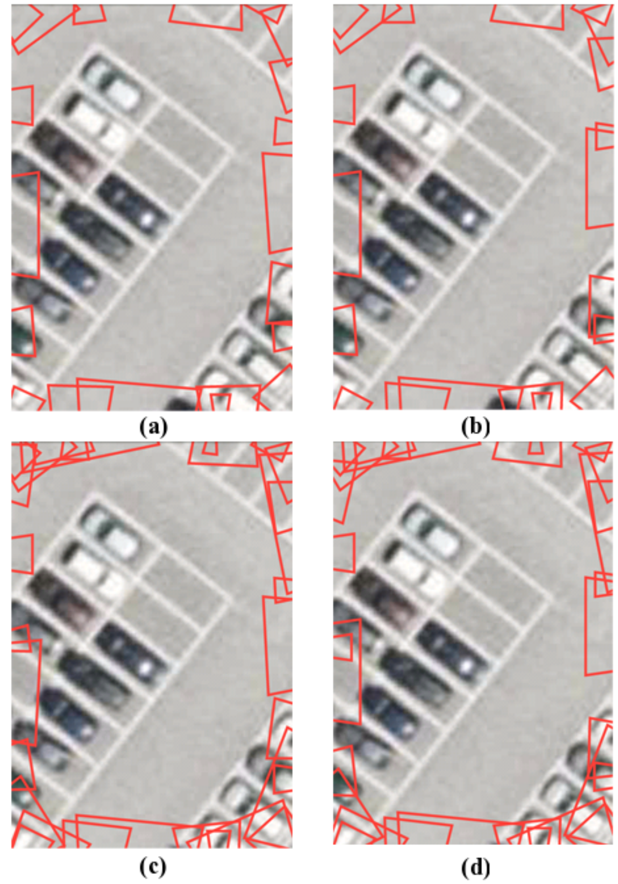


Figure 2. Examples of configurations with 20 off-screen objects (a)-(b) and 30 off-screen objects (c)-(d). In (b) and (d), the algorithm to resolve overlaps was not used.

corresponding off-screen object. This mechanism makes it easier for the user to identify the closest off-screen objects while keeping awareness of all off-screen objects. Figure 3b shows an example of the application of this mechanism to the configuration of 40 off-screen objects displayed in Fig. 3a.

The second mechanism we added to Wedge displays only those wedges corresponding to off-screen objects whose distance from the border of the screen is lower than a predefined threshold. This mechanism can reduce clutter in the case of a large number of distant objects, making wedges associated to closer objects more visible. The drawback is that it does not keep users aware of the global configuration of off-screen objects. Figure 3c shows an example of the application of this mechanism.

Activating the two mechanisms at the same time further declutters the visualization, as Fig. 3d shows.

Our study compared Wedge with a classic O+D approach that shows an overview of the information space as a small-scale thumbnail at the bottom right corner of the screen. A semi-transparent viewfinder in the overview highlights the area of the information space displayed in the detail view. Following the suggestions about overview size given in

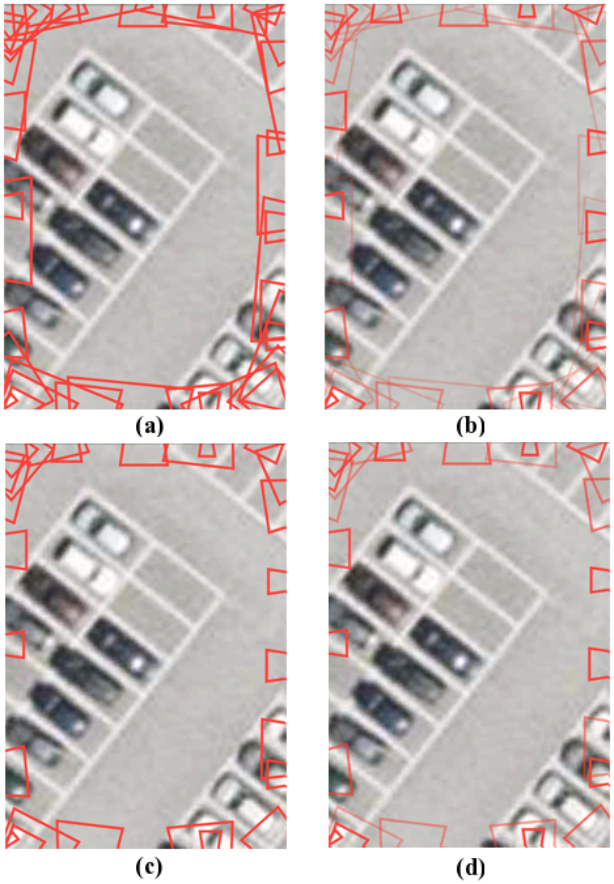


Figure 3. (a) Example of configuration with 40 off-screen objects. (b) Example of application of the transparency mechanism. (c) Example of application of the distance mechanism. (d) Example of application of both transparency and distance mechanisms.

[20], our overview covers about 15% of the detail view, a compromise between legibility of its content and occlusion of the detail view. In the overview, objects of interest were displayed as colored dots (Figure 4). We extended the visualization with the same distance mechanism used in Wedge, based on displaying only those dots associated to objects which are closer than a predefined distance threshold, to reduce clutter due to a high density of dots. However, we did not add the transparency mechanism because we found that it negatively affected readability of the visualization, due to the small size of dots in the overview. Moreover, dots did not overlap, thus making transparency much less useful as a way to reduce clutter in the visualization.

USER STUDY

To evaluate the effectiveness of the two approaches based on Wedge and O+D in a dynamic situation with a large number of off-screen objects, we defined a complex scenario in which users had to navigate a map containing two categories of objects: target objects that users had to find and dangerous objects that users had to avoid. Target objects were positioned as a trail while dangerous objects framed the path, constraining user navigation. Users had to follow the sequence of target objects, tapping on each of them, while keeping dangerous objects outside of the display area during navigation. The scenario thus combines two tasks, i.e. Closest ([3, 7, 8, 16]) and Avoid ([3, 16]), that have been used in the literature to compare techniques for off-screen objects visualization. Such scenario reflects possible complex mobile phone usage scenarios, such as following a tour of points of interest while avoiding traffic jams or other obstacles.

Figure 4. Close-up of the overview in the O+D visualization. The overview displays objects of interest as colored dots.

The scenario was designed as a game to increase user involvement. In the game, users had to navigate the map (by dragging it) following a trail of coins (target objects) while avoiding monsters (dangerous objects). Users could collect coins by tapping on them on the screen. Coins where positioned in such a way that at most one could be on screen at any time while the others were all off-screen. Each collected coin gave users a 10 points bonus. Each time a monster entered the screen, the user would get a penalty of 10 points. Moreover, the user had 10 seconds to collect the next coin before expiration of a timer and an additional 10 points penalty. The timer was included to make the game more challenging but the 10 seconds time interval was large enough to let users easily complete the task. Indeed, during the study, the timer expired only 3 times. The timer had also the function of recreating the situation in which a user has a short period of time to look at her mobile phone screen, and thus to evaluate which is the closest object of interest or which are the objects to keep out of the screen. The goal of the game was to obtain the maximum possible score, collecting all coins.

Figure 5 shows the interface of the game with Wedge (Fig. 5a) and O+D (Fig. 5b), respectively. In the Wedge condition, wedges associated to coins were yellow, while wedges associated to monsters were red. In the O+D condition, the overview displayed coins as yellow points and monsters as red points. In both conditions, the distance threshold was set to 300 pixels. In the Wedge condition, the minimum and maximum transparency thresholds were set respectively to 100 and 300 pixels. We chose these values because they guarantee maximum visibility to the closest objects, reducing at the same the clutter caused by the biggest wedges. The thresholds in the two conditions assured that, if the same

(a) (b)

Figure 5. The game interface with Wedge (a) and O+D (b). The status bar at the top of the screen shows total score, total time and the remaining time before expiration of the timer. Both examples in the figure show a coin, a monster and a number of off-screen objects.

portion of the map were showed in the detail view, the number of points visible in the overview was the same as the number of wedges placed along the border of the screen. In both interfaces, the status bar located at the top of the screen showed global score, total time and the progress bar displaying remaining time before expiration of the timer (top of Fig. 5a and 5b).

We also took into consideration the effect of the complexity of the configuration of objects on user performance. We thus prepared 4 configurations of objects, two of which were of lower difficulty and two of higher difficulty. In high difficulty configurations, the trail of coins followed a more winding path. Each configuration contained 20 coins and 20 monsters.

Hypotheses

Considering the results of the only study that compared Wedge and O+D in a static scenario [7], one would be led to expect that Wedge could be more effective than O+D in our game scenario. Indeed, that study revealed that users were better at estimating distance of multiple off-screen objects with Wedge. This ability could be useful to monitor multiple monsters in the game, allowing users to tune their panning actions and make less errors. Wedge might also be preferred by users in a game scenario, due to its more attractive graphical aspect. However, since the static scenarios with a few off-screen objects used in the literature are very different from the dynamic scenario with larger numbers of off-screen objects in our study, there is no guarantee that past results would transfer to the new context. Moreover, considering how we prepared the configurations of objects, we would expect that users take more time and make more errors while playing with the high difficulty configurations than with the low difficulty ones, regardless of the visualization.

Participants

The study involved a sample of 16 users (8 male and 8 female) recruited among university students. Their age ranged from 21 to 28 ($M = 24.5, SD = 1.79$). Four users were left-handed, even if one of them played with the right hand, and 12 were right-handed. All users had normal or corrected-to-normal vision. On a self-reported scale ranging from 1 (low familiarity) to 7 (high familiarity), users had a high level of familiarity with mobile devices ($M = 6.13, SD = 1.02$) while they where slightly less familiar with mobile touchscreen devices ($M = 5.50, SD = 1.63$). Almost half of the users played less than once a month with video games for console or computer and with video games for mobile devices.

Materials

The study was carried out on a HTC Desire mobile phone with Android 2.2, featuring a 1GHz processor and a 3.7 inches touch screen with a resolution of 320x533 pixels. During the evaluation, a 320x463 area displayed the considered map while two menu bars were displayed at the top of the screen: a 25-pixels tall Android notification bar and the 45-pixels tall status bar containing elapsed time, total score and the progress bar, as shown at the top of Fig. 5a and 5b. In the O+D condition, the overview was located at the bottom right corner of the screen and covered a maximum area of 140x160 pixels (15% of the detail view). In both conditions, users could drag the displayed map in the desired direction using their fingers. While using O+D, users could not move the viewfinder in the overview to carry out panning actions. During the evaluation, the device was in portrait mode and placed on a table over a mat to avoid sliding. Figure 5 shows examples of the two considered approaches on the mobile phone.

The following questionnaires were administered to users in different phases of the study. A *demographic questionnaire* was used to collect data about users (age, sex, occupation, dominant hand and sight defects) and their familiarity with mobile devices and touchscreen mobile devices. Familiarity was measured on a 7-levels Likert scale where 1 corresponded to low familiarity and 7 to high familiarity. Moreover, the questionnaire contained two items asking how often users played with video games for console or computer and with video games for mobile devices. The possible answers were: never, almost once a month, several times a month, several times a week, every day less than 1 hour, every day about 1-3 hours, every day more than 3 hours.

The *NASA Task Load Index* (TLX) [19] was used to measure subjective mental workload experienced by users during game sessions. NASA-TLX is a subjective workload assessment technique that derives an overall score based on a weighted average of ratings on six sub-scales: mental demand, physical demand, temporal demand, performance, effort, and frustration level. We administered to users a computerized version of the NASA-TLX (installed on a mobile device) [11] at the end of each game session. NASA-TLX scores can range in the 1-100 interval.

A modified version of the *Player Enjoyment Scale* (PES) was used to evaluate the enjoyment perceived and the experience of flow in a gaming activity [13]. PES is based on the EGameFlow scale [14] and includes 13 bipolar statements, both positive and negative, with scores from 1 (strongly disagree) to 7 (strongly agree). We removed the 7th and 12th items ("The difficulty of challenges in the game increased as my skills improved" and "I experienced an altered sense of time while playing the game") because no increase of difficulty was planned (participants played a single level of the game) and the gaming session was short (five minutes). We had to slightly change the 8th item ("The game provided new challenges with an appropriate pacing") into "The game proceeded with an appropriate pacing", because the challenges of the game did not change during each session. A total enjoyment score is obtained by reversing the negative item scores and summing them to positive item scores. Since we removed the two above mentioned items of the PES, our modified version of the scale (for brevity, mPES) returns a score that can range from 11 to 77.

We also asked participants to rate three additional statements ("I found it difficult to understand the information about off-screen objects provided by wedges/overview", "I found the information about off-screen objects provided by wedges/overview useful", "I found the game difficult") to better evaluate user perception of the two considered approaches.

Experimental design and procedure

The study was based on a 2x2 within subjects design with Approach (Wedge and O+D) and Difficulty of objects configuration (low difficulty and high difficulty) as independent variables. We used a Latin square design to guarantee that the order of presentation of experimental conditions as well as their association with objects configurations were counterbalanced to mitigate order effects.

Participants carried out the evaluation in a seated position. They were individually briefed about the nature of the experiment and were asked to fill in the demographic questionnaire. Afterwards, the experimenter described in detail the two approaches and presented training tasks to users, to let them familiarize with the interfaces and clarify possible doubts concerning interfaces or tasks. The evaluation consisted of four game sessions. Users played two times with each approach, with two different configurations of objects (low difficulty and high difficulty). To start a task, users had to tap on a "Start" button that was initially displayed on the screen. Each task ended when users collected the last coin. After two sessions with an approach, users were asked to fill in the NASA-TLX and the mPES questionnaires. After the test, users were also asked to express their preference on the considered approaches. Tests had an average length of 30 minutes.

For each user, we automatically recorded time spent to complete each session, total score and total number of penalties, i.e. the appearance of a monster on the screen and the expiration of the timer.

Results

Task completion time

Figure 6 shows the mean completion times for all four possible combinations of Approach and Difficulty of objects configuration.

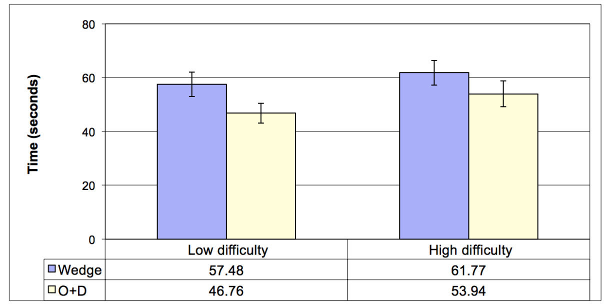


Figure 6. Mean completion times (capped bars indicate $\pm 1SE$) with the two approaches, for the two levels of difficulty.

Task completion times were subjected to the Kolmogorov-Smirnov test of normality prior to further analysis. The test revealed no significant deviation from the normal distribution. A two-way repeated measures ANOVA was then carried out, revealing no significant interaction between Approach and Difficulty $(F(1,15) = 0.30, p = 0.59)$. A significant main effect of Approach was detected, $(F(1,15) = 9.18, p < 0.01, \eta^2 = 0.38)$: users spent less time to carry out game sessions using O+D. A significant main effect of Difficulty was also detected, $(F(1,15) = 6.15, p < 0.05, \eta^2 = 0.29)$: users took more time to complete the task when playing with a high difficulty configuration of objects.

Errors

Figure 7 shows the mean number of errors for all four possible combinations between Approach and Difficulty of objects configuration. We only analyzed errors due to monsters entering the screen because the number of timer expiration errors was negligible (only 3 over the whole experiment).

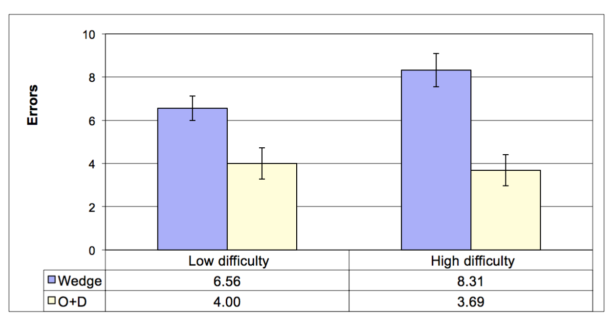


Figure 7. Mean number of errors (capped bars indicate $\pm 1SE$) with the two approaches, for the two levels of difficulty.

The Kolmogorov-Smirnov test of normality revealed no significant deviation from the normal distribution. A two-way repeated measures ANOVA was then carried out, which revealed no significant interaction between Approach and Difficulty, ($F(1, 15) = 3.63, p = 0.07$), and no significant main effect of Difficulty ($F(1, 15) = 1.11, p = 0.31$). However, a significant main effect of Approach was detected, ($F(1, 15) = 29.92, p < 0.0001, \eta^2 = 0.67$): users made less errors using O+D.

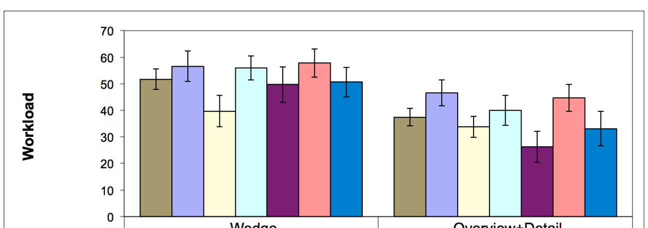
Subjective mental workload

Figure 8 shows means of the subjective mental workload we measured with the NASA-TLX. We report means for both the overall score and the six individual sub-scale ratings: mental demand (MD), physical demand (PD), temporal demand (TD), performance (PE), effort (EF), and frustration level (FL). A Wilcoxon signed-rank test on the overall scores pointed out a significant effect ($W = 15, p < 0.01, r = 0.48$) suggesting that users experienced less mental workload while using O+D. The Wilcoxon signed-rank test on sub-scale ratings revealed a significant effect for temporal demand ($W = 22, p < 0.05, r = 0.38$), performance ($W = 19.5, p < 0.05, r = 0.44$) and effort ($W = 18, p < 0.05, r = 0.39$).

Game enjoyment

Figure 9 shows the mean overall mPES scores. The data was analyzed with a Wilcoxon signed-rank test, which revealed a significant difference between the means ($W = 6.50, p < 0.01, r = 0.54$), with users perceiving O+D as more enjoyable than Wedge.

The Wilcoxon signed-rank test on each individual mPES statement revealed a significant effect for items "I perceived a sense of control on the game" ($W = 3.5, p < 0.01, r = 0.48$) (Fig. 10) and "While playing I always knew what was the next step" ($W = 3, p < 0.01, r = 0.58$) (Fig. 11) with O+D getting better scores. The Wilcoxon signed-rank test on the additional statements we asked users revealed a significant effect for item "I found the game difficult" ($W = 4, p < 0.05, r = 0.39$) (Fig. 12) with O+D getting a better score.



	Wedge	Overview+Detail
Global	51.73	37.41
MD	56.56	46.56
PD	39.69	33.75
TD	55.94	40.00
PE	49.69	26.25
EF	57.81	44.69
LF	50.63	33.13

Figure 8. Mean overall and sub scale workload ratings (capped bars indicate $\pm 1SE$) for the two approaches.

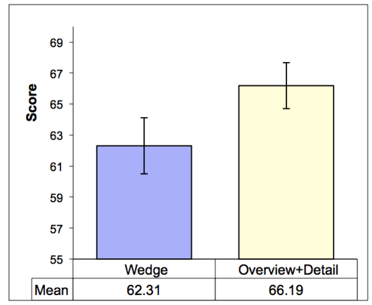


Figure 9. Mean overall scores of mPES (capped bars indicate $\pm 1SE$).

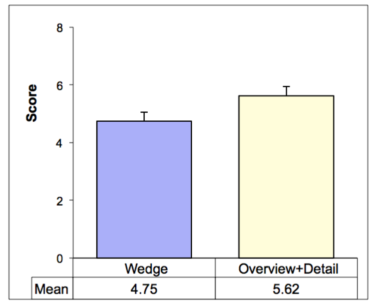


Figure 10. Mean overall scores (capped bars indicate $\pm 1SE$) for item "I perceived a sense of control on the game".

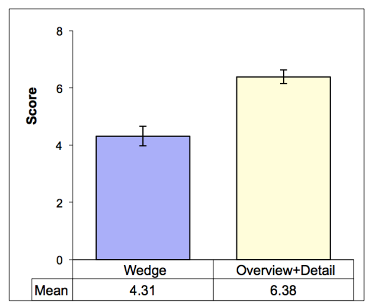


Figure 11. Mean overall scores (capped bars indicate $\pm 1SE$) for item "While playing I always knew what was the next step".

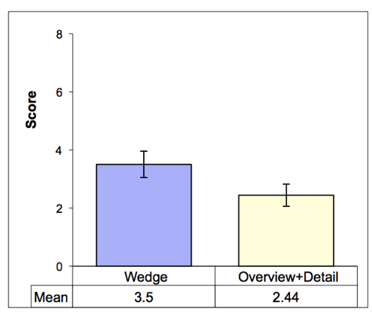


Figure 12. Mean overall scores (capped bars indicate $\pm 1SE$) for item "I found the game difficult".

Subjective preference

Four users preferred Wedge while the other 12 preferred O+D. A chi-square test revealed a significant difference between the scores ($\chi^2(1, N = 16) = 4, p < 0.05$) with O+D being the preferred approach.

DISCUSSION AND CONCLUSIONS

Overall, our evaluation found that the better results that Wedge obtains over O+D in static scenarios with a few off-screen objects do not transfer to a dynamic scenario with larger numbers of objects. In the latter context, O+D seems to be a more effective approach to off-screen objects visualization.

Users were faster while using O+D in terms of the time spent to carry out tasks. A possible explanation of this result could be that each panning action with Wedge changes the position and size of proxy shapes, while the position of off-screen objects in the overview does not change during navigation. Thus, users might spend more time with Wedge because they need to partially re-evaluate the visualization after every navigation action. Moreover, the presence of the viewfinder in the overview probably acts as a navigation aid that let users better orient and navigate in the information space. Note that this result is different from what was obtained in the study described in [7], where no difference was found between Wedge and O+D for the time needed by users to find the off-screen object nearest to the screen, probably because the scenario considered in that study was static.

Users made more errors using Wedge than using O+D. Again, the presence of static information in the overview might allow users to more easily avoid dangerous objects compared to the need to constantly re-evaluate wedges in Wedge. Users commented that the information provided by wedges changed too rapidly during navigation to allow them to easily spot monsters that could enter the screen. As for task completion times, this result differs from what was obtained in [7], where no significant differences were found between Wedge and O+D in the Closest task (in which users had to identify the off-screen object closest to the border of the screen) and in the Locate task (in which users had to estimate the exact location of every off-screen object).

Considering the configuration of objects, users took more time to complete the high difficulty configurations compared to the low difficulty ones, regardless of the type of approach used. However, contrary to our hypothesis, they did not make more errors when playing with high difficulty configurations. A possible explanation could be that users spent more time to keep dangerous objects out of the screen.

In NASA-TLX results, Wedge scored worse in all ratings. In particular, the differences were significant for the overall score and for the temporal demand, performance and effort sub-scale factors. These results suggest that the information provided by Wedge requires more mental workload in a dynamic context. As in previous results, this could be explained by the fact that the global context needs to be re-evaluated after every minimal panning action, increasing perceived effort. The increased temporal demand perceived by users could be due to the additional time needed to evaluate the situation compared to O+D. Finally, the low rating for performance could be due to the high number of errors made by users.

The mPES scores showed that users enjoyed playing with O+D more than playing with Wedge. The significant difference for the item “While playing I always knew what was the next step” could be a consequence of the fact that, when using O+D, users had a better sense of the global situation, similarly to what emerged in [7]. The significant difference we found for the item “I found the game difficult” could be due to the higher number of errors made by users using Wedge and the higher difficulty users had to understand the information provided by wedges. Globally, these factors could have negatively influenced the perceived enjoyment.

Users’ subjective preference reflects performance. In general, user comments show that the information provided by O+D was more useful to understand the global position of off-screen objects and allowed users to more easily navigate the map. The few negative comments about O+D mentioned that the overview occluded part of the detail view while the use of wedges did not occlude a sizable portion of the map, even if it also did not allow users to easily determine the overall configuration of off-screen objects.

In conclusion, the results of the study point out that O+D seems to be more effective than Wedge as a solution to visualize the position of off-screen objects in the considered context, making it more appropriate for real-world usage scenarios.

ACKNOWLEDGMENTS

Andrea Vianello acknowledges the financial support of the PRIN 2008 project “Descriptive Geometry and Digital Representation: intelligent user interfaces to support modeling and navigation in 3D graphics application for architecture and design”.
Stefano Burigat acknowledges partial financial support of the Italian Ministry of Education, University and Research (MIUR) under a law 17/1999 project.

REFERENCES

1. Baudisch, P., Good, N., Bellotti, V., and Schraedley, P. Keeping things in context: a comparative evaluation of focus plus context screens, overviews, and zooming. In *Proc. Conference on Human Factors in Computing Systems*, ACM Press (2002), 259–266.

2. Baudisch, P., Lee, B., and Hanna, L. Fishnet, a fisheye web browser with search term popouts: a comparative evaluation with overview and linear view. In *Proc. Conference on Advanced Visual Interfaces*, ACM Press (2004), 133–140.

3. Baudisch, P., and Rosenholtz, R. Halo: a technique for visualizing off-screen locations. In *Proc. Conference on Human Factors in Computing Systems*, ACM Press (2003), 481–488.

4. Beard, D., and Walker, J. Navigational techniques to improve the display of large two-dimensional spaces.

Behaviour and Information Technology 9, 6 (1990), 451–466.

5. Bederson, B., and Hollan, J. Pad++: a zooming graphical interface for exploring alternate interface physics. In *Proc. Conference on User Interface Software and Technology*, ACM Press (1994), 17–26.
6. Burigat, S., and Chittaro, L. On the effectiveness of overview+ detail visualization on mobile devices. *Personal and Ubiquitous Computing* (2011), doi: 10.1007/s00779–011–0500–3.
7. Burigat, S., and Chittaro, L. Visualizing references to off-screen content on mobile devices: A comparison of arrows, wedge, and overview+ detail. *Interacting with Computers 23*, 2 (2011), 156–166.
8. Burigat, S., Chittaro, L., and Gabrielli, S. 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*, ACM Press (2006), 239–246.
9. Burigat, S., Chittaro, L., and Parlato, E. Map, diagram, and web page navigation on mobile devices: the effectiveness of zoomable user interfaces with overviews. In *Proc. Conference on Human-Computer Interaction with Mobile Devices and Services*, ACM Press (2008), 147–156.
10. Büring, T., Gerken, J., and Reiterer, H. Usability of overview-supported zooming on small screens with regard to individual differences in spatial ability. In *Proc. Conference on Advanced Visual Interfaces*, ACM Press (2006), 233–240.
11. Cao, A., Chintamani, K., Pandya, A., and Ellis, R. Nasa tlx: Software for assessing subjective mental workload. *Behavior research methods 41*, 1 (2009), 113–117.
12. Chittaro, L. Visualizing information on mobile devices. *IEEE Computer 39*, 3 (2006), 40–45.
13. Chu Yew Yee, S., Duh, H., and Quek, F. Investigating narrative in mobile games for seniors. In *Proc. Conference on Human Factors in Computing Systems*, ACM Press (2010), 669–672.
14. Fu, F., Su, R., and Yu, S. Egameflow: A scale to measure learners' enjoyment of e-learning games. *Computers & Education 52*, 1 (2009), 101–112.
15. Gonçalves, T., Afonso, A., Carmo, M., and Pombinho, P. Evaluation of halodot: visualization of relevance of off-screen objects with over cluttering prevention on mobile devices. In *Proc. IFIP Conference on Human-computer interaction*, Springer-Verlag (2011), 300–308.
16. Gustafson, S., Baudisch, P., Gutwin, C., and Irani, P. Wedge: clutter-free visualization of off-screen locations. In *Proc. Conference on Human Factors in Computing Systems*, ACM Press (2008), 787–796.
17. Gustafson, S., and Irani, P. Comparing visualizations for tracking off-screen moving targets. In *Proc. Conference on Human Factors in Computing Systems*, ACM Press (2007), 2399–2404.
18. Gutwin, C., and Skopik, A. Fisheyes are good for large steering tasks. In *Proc. Conference on Human Factors in Computing Systems*, ACM Press (2003), 201–208.
19. Hart, S., and Staveland, L. Development of NASA-TLX (task load index): Results of empirical and theoretical research. *Human mental workload 1* (1988), 139–183.
20. Hornbæk, K., Bederson, B., and Plaisant, C. Navigation patterns and usability of zoomable user interfaces with and without an overview. *ACM Transactions on Computer-Human Interaction 9*, 4 (2002), 362–389.
21. Hornbæk, K., and Frøkjær, E. Reading patterns and usability in visualizations of electronic documents. *ACM Transactions on Computer-Human Interaction 10*, 2 (2003), 119–149.
22. Irani, P., Gutwin, C., and Yang, X. Improving selection of off-screen targets with hopping. In *Proc. Conference on Human Factors in Computing Systems*, ACM Press (2006), 299–308.
23. Nekrasovski, D., Bodnar, A., McGrenere, J., Guimbretière, F., and Munzner, T. An evaluation of pan & zoom and rubber sheet navigation with and without an overview. In *Proc. Conference on Human Factors in Computing Systems*, ACM Press (2006), 11–20.
24. North, C., and Shneiderman, B. Snap-together visualization: can users construct and operate coordinated visualizations? *International Journal of Human-Computer Studies 53*, 5 (2000), 715–739.
25. Plaisant, C., Carr, D., and Shneiderman, B. Image-browser taxonomy and guidelines for designers. *IEEE Software 12*, 2 (1995), 21–32.
26. Roto, V., Popescu, A., Koivisto, A., and Vartiainen, E. Minimap: a web page visualization method for mobile phones. In *Proc. Conference on Human Factors in Computing Systems*, ACM Press (2006), 35–44.
27. Sekuler, A., Palmer, S., and Flynn, C. Local and global processes in visual completion. *Psychological Science 5*, 5 (1994), 260–267.
28. Zellweger, P., Mackinlay, J., Good, L., Stefik, M., and Baudisch, P. City lights: contextual views in minimal space. In *Proc. Conference on Human Factors in Computing Systems*, ACM Press (2003), 838–839.