Visualizing Locations of Off-Screen Objects on Mobile Devices: A Comparative Evaluation of Three Approaches

Stefano Burigat
HCI Lab
Dept. of Math and Computer Science
University of Udine
Via delle Scienze, 206
33100 Udine, Italy
+39 0432 558457
burigat@dimi.uniud.it

Luca Chittaro
HCI Lab
Dept. of Math and Computer Science
University of Udine
Via delle Scienze, 206
33100 Udine, Italy
+39 0432 558450
chittaro@dimi.uniud.it

Silvia Gabrielli
HCI Lab
Dept. of Math and Computer Science
University of Udine
Via delle Scienze, 206
33100 Udine, Italy
+39 0432 558457
gabrielli@dimi.uniud.it

ABSTRACT
Browsing large information spaces such as maps on the limited screen of mobile devices often requires people to perform panning and zooming operations that move relevant display content off-screen. This makes it difficult to perform spatial tasks such as finding the location of Points Of Interest (POIs) in a city. Visualizing the location of off-screen objects can mitigate this problem: in this paper, we present a user study comparing the Halo [2] approach with two other techniques based on arrows. Halo surrounds off-screen objects with circles that reach the display window, so that users can derive the location and distance of objects by observing the visible portion of the corresponding circles. In the two arrow-based techniques, arrows point at objects and their size and body length, respectively, inform about the distance of objects. Our study involved four tasks requiring users to identify and compare off-screen objects locations, and also investigated the effectiveness of the three techniques with respect to the number of off-screen objects. 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. Implications of these results for mobile map-based applications are also discussed.

Categories and Subject Descriptors
H.5.2 [**Information Interfaces and presentation**]: Evaluation, screen design, Graphical user interfaces (GUI); I.3.6. [**Computer** Graphics]: Interaction techniques

General Terms
Design, Experimentation, Human Factors

Keywords
Visualization, mobile devices, off-screen locations, maps.

1. INTRODUCTION
Limited screen size in mobile devices, such as PDAs and Smartphones, makes it difficult to display large information spaces (e.g., maps, photographs, web pages, etc.). This requires users to perform panning and zooming operations, a process which is cognitively complex, as well as disorienting and tedious. Indeed, displaying an information space in its entirety may provide only an overview without sufficient detail, while a zoomed-in view provides details but makes relevant content disappear off-screen. This is a serious problem for users who need to perform spatial tasks, such as tourists who look for suitable points of interest (e.g., restaurants, monuments, gas stations, etc.) on a map, or first responders who need to identify locations of potential hazards in a building or view the real-time location of other team members [3]. To mitigate this problem, one can provide users with information that enables them to locate relevant objects even when they are off-screen [7].

In this paper, we experimentally evaluate three different approaches for visualizing locations of off-screen objects on small screens, comparing their effectiveness in supporting some spatial cognition tasks. The paper is organized as follows. Section 2 surveys related work. Section 3 discusses possible visual encodings to visualize the location of off-screen objects. Section 4 presents the three approaches compared in our study. Section 5 describes the experimental evaluation and reports its results. Finally, Section 6 discusses implications of the results for mobile map-based applications and points out future research directions.

2. RELATED WORK
Several techniques have been proposed in the literature to display large information spaces on the limited screens of mobile devices. When possible, content analysis is used to restructure the information space into areas of related content that fit on the display screen [4][5][6][21]. However, these techniques are meant for web pages, and they are usually unsuitable for images and maps.

In the case of images and maps, a basic approach is to provide users with panning and zooming capabilities that allow them to select the portion of space to visualize, while omitting the remaining space. Panning repositions information on the screen and has been found to work well for relatively small images [16], but to be rather tedious for larger ones [13]. Zooming changes the scale of the information space and can be used to obtain different views on it [11][9]. Typically, only sliders and menus are available to the user for panning and zooming, but alternative interaction techniques have been developed to simplify these operations on mobile devices. For example, ZoneZoom [18] 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 on the phone keypad, and pressing a key produces an automated pan and zoom on the associated cell. Jones et al. [12] present a technique that reduces the physical workload of users by combining zooming and panning into a single operation, dependent on how much users drag the pointing device on the screen with respect to the starting position.

Other techniques to display maps and images on small screens are based on combining panning and zooming with compression or distortion operations on the information space and can be classified into Overview&Detail and Focus&Context techniques.

Overview&Detail techniques provide one or multiple overviews of the whole space (usually at a reduced scale), simultaneously with a detailed view of a specific portion of space [16]. For example, in the Large Focus-Display [14], the overview is a downscaled version of the information space that highlights the currently displayed portion as a rectangular viewfinder. Users can drag and resize the viewfinder to perform panning and zooming operations. By examining the size and position of the viewfinder, users are also able to derive useful information for the browsing process, such as the scale ratio between the displayed portion and the whole information space. Unfortunately, the overview covers part of the detailed view, thus hiding useful information. As a possible solution, Rosenbaum and Schuman [20] propose an adaptation of the ZoneZoom technique to perform panning and zooming operations on images by interacting with a grid overlaid on the currently displayed image portion. The grid is proportional to the whole image and each grid cell can be tapped to display the corresponding portion of the image. Cells can also be merged or splitted to provide users with different zoom levels.

Focus&Context techniques display the information space at different levels of detail simultaneously, without separating the different views. A focus area displaying undistorted content is embedded in surrounding context areas distorted to fit into the available screen space. A typical example of these techniques is provided by the Rectangular FishEye-View [17], where a rectangular focus is surrounded by one or more context belts (composed of grid rectangles), 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 with increasing distance from the focus. The disadvantage of Focus&Context techniques is that the different scales and the distortions introduced make it more difficult for users to integrate all information into a single mental model and interfere with tasks that require precise geometric assessments [1].

Despite their capability to provide overviews of an information space, both Overview&Detail and Focus&Context techniques are not always the suitable solution to the off-screen objects issue because of their previously mentioned disadvantages. Techniques that are explicitly aimed at providing information about the location of off-screen objects are thus often necessary. Examples of these techniques can be found in videogames (e.g., [19]) and in virtual environments (e.g., [8]), where arrows are used as indications to help users find specific objects or places. CityLights [15] are 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 [2], shows off-screen objects locations by surrounding them with circles that are just large enough to reach into the border region of the display window. From the portion of the 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.

3. POSSIBLE VISUAL ENCODINGS

In the initial phase of our work, we focused on identifying alternative visual encodings to convey information about the direction and distance of off-screen objects.

Table 1 lists the main options available, not including those encodings that were redundant for a given technique. For example, direction and distance in Halo are intrinsically encoded through arc position and arc distance and any other encoding (e.g., using color to convey distance information) would not be an alternative. It must be noted that by CityLights we really mean the variant of CityLights that uses lines along the screen border .

Subsequently, we developed mockups to better compare benefits and drawbacks of all visual encodings. Fig. 1 shows some of the

Table 1: Possible visual encodings

Type of Technique	Encoding of Direction	Encoding of Distance
Halo	Arc position	Arc curvature
Arrows	Arrow orientation	Arrow length Arrow size Arrow color Arrow shape Label
CityLights	Line position	Line thickness Line color Label

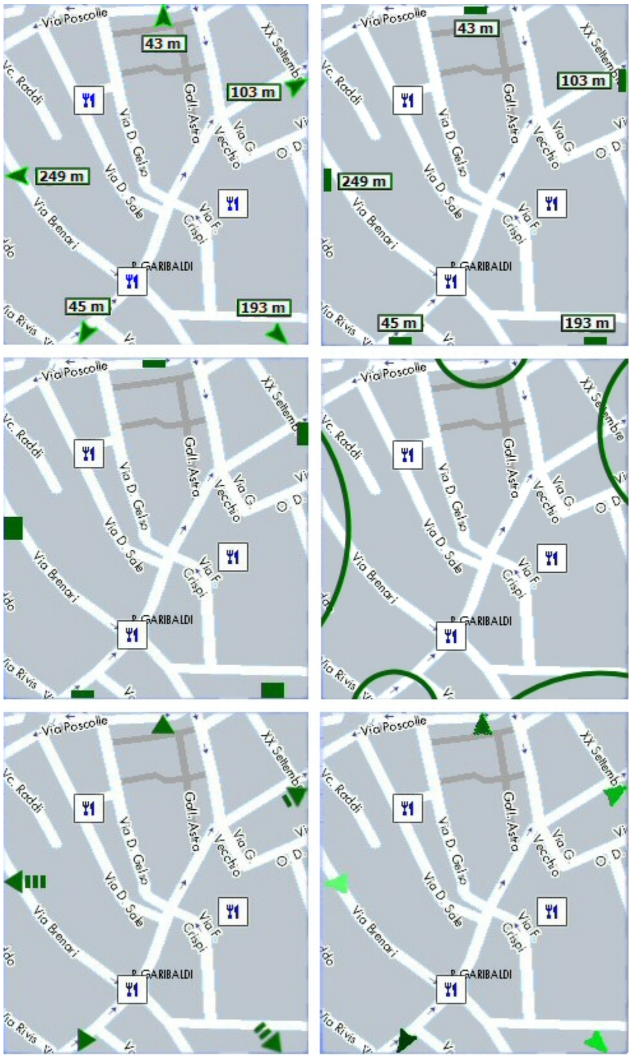

Fig. 1: Example of mockups developed to analyze possible visual encodings

mockups for a sample configuration involving 5 objects: the upper two images show the use of labels for indication of distance associated to arrows and CityLights lines respectively; in the middle image on the left, the thickness of CityLights lines is used to convey information about objects distance (the thicker the line, the farther the object is located); the middle image on the right shows an example of Halo with arcs enabling users to derive the location of objects; in the lower image on the left, the length of arrow bodies is proportional to the distance of objects and, finally, the lower image on the right shows the use of color shades in arrows to provide information about the distance of objects (the lighter the shade, the farther the object).

The comparison raised a number of research questions that we decided to investigate further through the user study described in Section 5. In particular, two important aspects must be carefully considered when designing solutions for off-screen objects visualization: i) Is it easy for a user to correctly interpret a certain visual encoding? For example, is it intuitive to associate the size of an arrow pointing to an off-screen object to the distance of the object from the screen border? ii) Is a particular visual encoding scalable with respect to the number of off-screen objects that need to be displayed on a map? In other words, is it more difficult for a user to derive information about off-screen objects from an encoding when the number of objects increases?

We decided to focus our experimental study only on solutions that were fully comparable in terms of direction and distance indication. We excluded the basic CityLights approach, because, unlike Halo and arrow-based approaches, it does not provide an accurate indication of object direction, being based on the use of lines on the border of the screen. Among the different visual encodings available for arrows, we focused on variations of body length and size, which made a comparative evaluation with Halo more interesting, while variations of color, as well as use of labels, were not considered since they can be similarly applied to any of the approaches analyzed.

4. THE CONSIDERED APPROACHES

Figure 2 illustrates the three approaches for off-screen object visualization that we compared in our study.

The upper image shows an example of the Halo [2] approach: the three arcs in the screen space belong to circles whose centers are off-screen and indicate the exact locations of 3 different objects. Users derive the off-screen location of objects from the curvature and relative size of the arcs. One can notice that the portion of screen border occupied by an arc grows as the location of the corresponding object is further away, so that more distant objects are associated to wider arcs. It is also worth noting that arcs may get cropped at the corners of the display window, making it more difficult for users to correctly derive information from them.

The middle image in Fig. 2 exemplifies the first one of the two arrow-based solutions we considered, called Scaled-Arrows. Arrow orientation is used to encode off-screen objects direction while arrow size is inverse-linearly proportional to the object distance from the screen. The larger the size (e.g., the arrow on the top border), the closer to the screen the object is located. Unlike Halo, this visualization associates off-screen objects that are located closer to the displayed area, which are usually more interesting for the user, to graphical elements that occupy more space on screen, thus being more easily noticeable.

The lower image in Fig. 2 exemplifies the second arrow-based solution, called Stretched-Arrows. In this case, the length of the arrow body is inverse-linearly proportional to the object distance from the screen. The longer the arrow body, the closer to the screen the object is located. Like Scaled-Arrows, this visualization associates objects that are located closer to the displayed area to graphical elements that occupy more space on screen.

5. USER STUDY

Our study compares Halo, Scaled-Arrows and Stretched-Arrows. Its overall goals were to assess if the three different visualizations were able to support users in accurately and quickly identifying off-screen objects, as well as to collect users' subjective preferences about the approaches presented. To this aim, users were asked to perform four different types of tasks, that involved finding the exact location of off-screen objects on a map or comparing their distances. Each task was performed twice: once

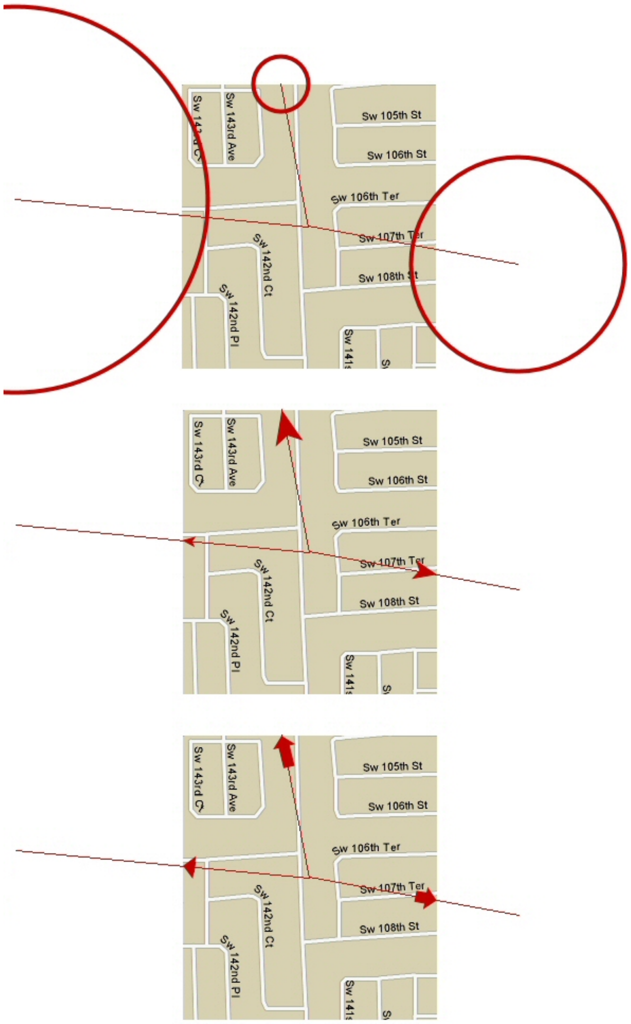

Fig. 2: The three considered approaches to visualize the location of off-screen objects: Halo (upper image), Scaled-Arrows (middle image), Stretched-Arrows (lower image)

with a map configuration containing 5 off-screen objects and once with one containing 8 objects.

5.1 Tasks

The tasks we considered were partially inspired and adapted from a related study [2]. Attention was paid to make tasks as realistic and meaningful as possible for the user. Tasks were presented through a scenario where the participant was asked to play the role of a tourist who had to identify the location of off-screen Points of Interest (POIs) in a city map displayed on the PDA. In the following, we list the names we gave to the tasks, together with their descriptions:

- *Closest*: indicate which is the closest POI among the ones visualized on the map (assuming you are located in the map center). With respect to the considered scenario, this task was explained as a situation in which the user wants to reach the closest location to visit. The user had to provide her answer by tapping on (or near) the corresponding arc/ arrow displayed on screen.

Fig. 3: Examples of 5 and 8 off-screen objects configurations

- *Estimate*: point out the pair of POIs which are closest to each other. In a tourist scenario, this task represents the situation of a tourist who wants to find two locations to visit during a short tour while limiting, as much as possible, the time needed to move from one location to the other. To execute the task, the user had to tap on or near the two arcs/arrows selected.

- *Order*: order the POIs in increasing distance from the map center. This task was explained as the situation in which the user needs to plan a tour including as many locations to visit as possible, starting from the closest and stopping at the farthest. To execute the task the user had to tap on or near each arc/arrow visualized on screen, in the correct sequence.

- *Locate*: indicate the exact location of each POI, by marking it with a pen on paper. This represents a situation in which a tourist wants to mark on paper the different locations to visit during a tour. To execute the task we provided the user with a sheet of paper reproducing at its center a printout of the map to be analyzed (displayed also on the PDA); the sheet of paper was empty in the area where the user had to mark the location of each POI.

We also studied user's performance in each task with different numbers of off-screen objects. To this end, we prepared configurations with 5 and 8 off-screen objects for each map (some sample configurations are shown in Fig. 3), so that we could test how users dealt with an increasing level of clutter in the visualization.

5.2 Apparatus and Procedure

A 624Mhz PocketPC with a 3.5" display and QVGA (320x240) resolution was employed. Logging code automatically recorded time taken and answers tapped by users. The maps used during the study depicted an area unknown to all participants.

A sample of 17 users was involved in the evaluation. Most of them were undergraduates or postgraduates at our university; we also took care to recruit a balanced group of users in terms of gender (M = 7, F = 10) and background (9 from scientific disciplines, 8 from Humanities). Sixteen out of 17 users had never or rarely used a PDA before. Eleven out of 17 users had used maps occasionally to navigate city environments.

The experimental design was within-subjects. To avoid any sequence or learning effects, tasks and conditions order, as well as off-screen objects configurations, were counterbalanced. Before starting the test, participants were individually briefed about the 3 visualizations, they were asked to fill in a short demographic questionnaire and verbally instructed about the tasks to be performed. They also tried four training maps for each visualization to familiarize with the visualizations and tasks assigned.

Participants were also interviewed at the end of the session to rate their preference for a particular visualization during each task and after the overall session was completed. The average duration of the test was approximately 30 minutes.

5.3 Hypotheses

Considering the features of the 3 visualizations and the specific tasks presented to users, our hypotheses in the study were the following:

- There would not be significant differences in the time taken by users to complete the Closest and Estimate tasks, as well as in the number of errors made, with all 3 visualizations.
- Arrow-based visualizations would allow users to complete the Order task significantly faster and with less errors than Halo, due to the lower amount of clutter they generate on the maps.
- Halo would outperform arrow-based visualizations in the Locate task due to the more accurate information on the exact location of off-screen objects.
- Users' preferences would mirror their performance results.

5.4 Results

5.4.1 Time to complete tasks

Two-way repeated measures ANOVA (Analysis of Variance) was employed on the times needed by users to complete the Closest, Order and Estimate tasks, with "Number of Objects" and "Type of Visualization" as within-subjects factors. The values of means are reported in Fig. 4, for tasks involving 5 and 8 off-screen objects.

For the Closest task, the main effect of "Number of Objects" did attain significance ($F(1, 96) = 6.54$, $p = 0.021$), with more time taken to complete the task when using 8 objects configurations, while the main effect of "Type of Visualization" did not. The interaction between the two factors was not significant. For the Estimate task, the main effect of "Number of Objects" did not attain significance, as well as the main effect of "Type of Visualization" and the interaction between the two factors. This confirms our first hypothesis. For more straightforward tasks participants were equally fast with all three types of visualizations.

For the Order task, a significant main effect was found for "Number of Objects" ($F(1, 96) = 62.029$, $p < 0.001$) and "Type of Visualization" ($F(2, 96) = 8.542$, $p < 0.001$). The interaction between the two factors did not reach significance ($F(2, 69) = 3.272$, $p = 0.051$). The main effect of "Type of Visualization" was

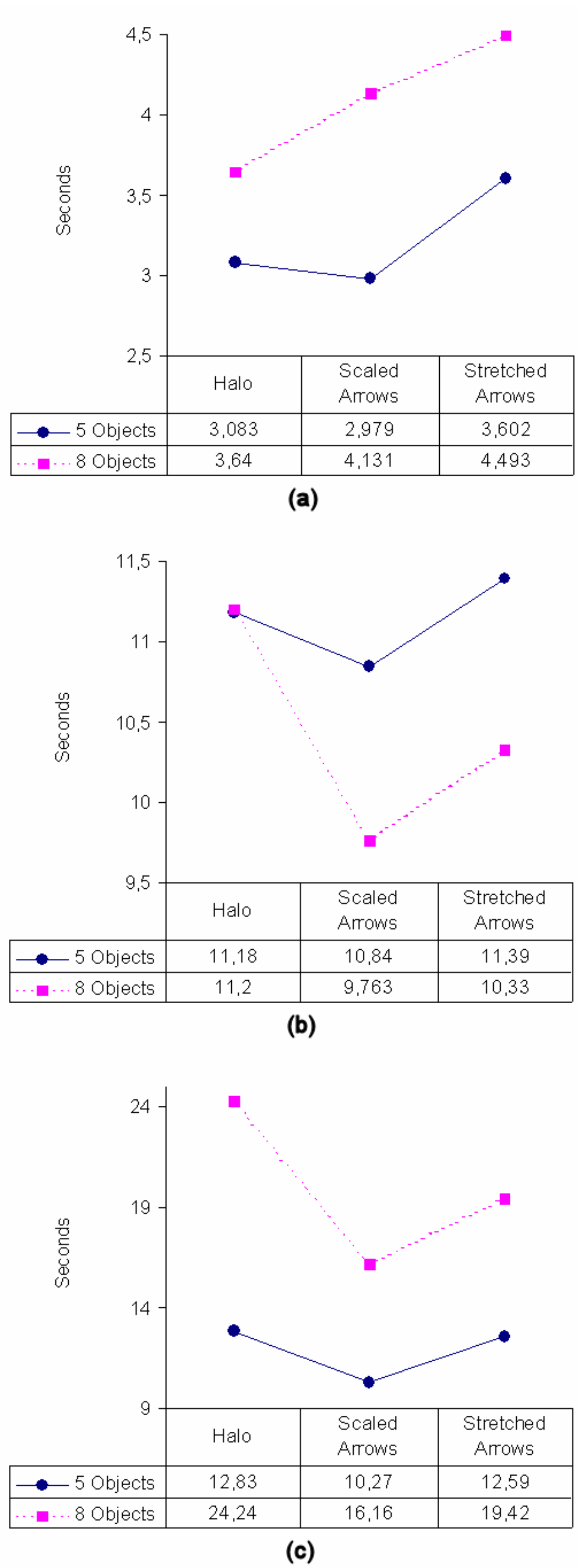

Fig. 4: Mean time to complete tasks Closest (a), Estimate (b) and Order (c)

Table 2: Mean number of errors in the Closest, Order and Estimate tasks for 5 and 8 objects configurations

Task	Halo	Scaled-Arrows	Stretched-Arrows
Closest (5 objects)	0	0,177	0,059
Closest (8 objects)	0	0	0,059
Estimate (5 objects)	0,412	0,353	0,294
Estimate (8 objects)	0,294	0,353	0,529
Order (5 objects)	0,765	0	0,118
Order (8 objects)	1,294	0,529	0,706

further investigated by comparing each possible pair of visualizations using one-way repeated measures ANOVA. A statistically significant difference was found between Halo and Scaled-Arrows ($F(1, 15) = 22.851$, $p < 0.001$) and Halo and Stretched-Arrows ($F(1, 15) = 5.578$, $p = 0.031$), while there was no statistically significant difference between Scaled-Arrows and Stretched-Arrows.

As expected, participants were faster in the Scaled-Arrows and Stretched-Arrows conditions if compared to Halo, and this effect was more evident in the case of 8 objects configurations. This result contributes, in part, to confirm our second hypothesis.

5.4.2 Errors

Two-way repeated measures ANOVA was carried out on the mean number of errors made in each of the four tasks, with "Number of Objects" and "Type of Visualization" as within-subjects factors. For the Closest, Estimate and Order tasks, we measured the number of errors made by users (i.e., selection of a wrong arc or arrow instead of the correct one, for each task). For the Locate task, we measured both the Distance Error (i.e., the Euclidean distance between the subject's location estimate and the actual location of the off-screen object) and the Angular Error (i.e., the distance in degrees between the radial direction of the user's location estimate and the actual radial direction of the off-screen object with respect to the screen center). Mean number of errors for the Closest, Estimate and Order tasks are reported in Table 2 for tasks involving 5 and 8 off-screen objects configurations. Mean distance errors for the Locate task are reported in Table 3 while mean angular errors are reported in Table 4.

For both the Closest and Estimate tasks, the ANOVA did not find significant main effects of "Number of Objects" and "Type of Visualization" on error, or a significant interaction effect between the two factors. These results are consistent with our first hypothesis.

For the Order task, a significant main effect was found for "Number of Objects" ($F(1, 96) = 11.551$, $p < 0.01$) and "Type of Visualization" ($F(2, 96) = 14.759$, $p < 0.001$), with an higher

Table 3: Mean distance error in the Locate task for 5 and 8 objects configurations

Task	Halo	Scaled-Arrows	Stretched-Arrows
Locate (5 objects)	1,312	2,565	2,718
Locate (8 objects)	2,053	4,376	4,3

Table 4: Mean angular error in the Locate task for 5 and 8 objects configurations

Task	Halo	Scaled-Arrows	Stretched-Arrows
Locate (5 objects)	4	5,118	4,941
Locate (8 objects)	11,35	9,941	9

mean number of errors made when using Halo. The interaction between the two factors did not reach significance ($F(2, 69) = 0.025$, $p > 0.05$). The main effect of "Type of Visualization" was further investigated by performing comparisons between each possible pair of visualizations using one-way repeated measures ANOVA. A statistically significant difference was found between the mean number of errors with Halo and Scaled-Arrows ($F(1, 15) = 13.364$, $p < 0.01$) and with Halo and Stretched-Arrows ($F(1, 15) = 21.043$, $p < 0.001$), while there was no statistically significant difference between Scaled-Arrows and Stretched-Arrows.

These findings further support our second hypothesis in showing participants' increased accuracy when using the arrow-based visualizations for this task.

For the Locate task, the ANOVA found a significant main effect of "Number of Objects" on the angular error ($F(1, 96) = 56.053$, $p < 0.001$), but no significant main effect of "Type of Visualization" ($F(2, 96) = 0.250$, $p > 0.05$) and no interaction effect ($F(2, 96) = 1.587$, $p > 0.05$). A significant main effect of both factors was also found on the distance error, (F(1, 96 = 18.523, $p < 0.001$) for "Number of Objects" and (F(2, 96) = 13.363, $p < 0.001$) for "Type of Visualization", but the interaction did not attain significance (F(2, 96 = 1.446, $p > 0.05$). The main effect of "Type of Visualization" was further investigated using one-way repeated measures ANOVA. Results show that subjects made significantly less errors with Halo than with Scaled-Arrows ($F(1, 15) = 14.977$, $p < 0.001$) and Stretched-Arrows ($F(1, 15) = 24.393$, $p < 0.001$).

These results confirm our third hypothesis, showing users' higher accuracy when using Halo in the Locate task. However, it is worth noting that while there were significant differences among the 3 visualizations for the distance error, no significant differences were found for the angular error. We speculate that the worse performance of users with Halo in configurations with 8 off-screen objects, as reported in Table 4, is partly due to the more cluttered visualization created by this approach when many off-screen objects are displayed on a map. Moreover, the

Table 5: Preferences expressed by users for the three visualizations in each task

Task	Halo	Scaled-Arrows	Stretched-Arrows
Closest	7	8	5
Order	3	10	5
Estimate	10	10	3
Locate	12	3	3

difficulty of the Locate task is increased by the chance of finding cropped arcs located close to the display window corners, which makes it harder for users to perform comparisons among arcs. By contrast, the less accurate performance of users with the arrow-based visualizations can be explained by considering that both Scaled-Arrows and Stretched-Arrows are characterized by graphical elements that allow users to *qualitatively* estimate distances, rather than provide precise information about them, as it is done by Halo.

5.4.3 Users' preferences

Table 5 reports users' preferences for each visualization and task assigned. Users were allowed to express their preference for more than one type of visualization.

In the Closest and Estimate tasks, both Halo and Scaled-Arrows received approximately the same number of preferences, as it would have been expected from users' performance in the same tasks. In the Order task, a larger number of users expressed their preference for the Scaled-Arrows visualization, while in the Locate task the majority of users preferred Halo.

Comments collected during the experiment and the post-test interviews contributed to better understand participants' impressions about the 3 visualizations presented. With respect to the intuitiveness of visualizations, we realized that although a few participants during the training phase took some time to familiarize with the 3 approaches, the majority found them straightforward from the beginning of the experiment.

Most users expressed appreciation for Halo when the task required to precisely estimate distance and object position. We observed during the Locate task, for instance, that users tried to draw a ring from the arc displayed on the map to derive the exact location of the object off-screen. By contrast, when the tasks required to compare distances and positions of many off-screen locations, users expressed appreciation for the arrow-based visualizations that were better able to provide a view '*at a glance*' of the correct answer to the task. The problem with Halo, according to a user, was that *"… too many arcs (say, more than 4) are difficult to take into account at once, so they provide too little support for the task…"*.

When asked which was their preferred visualization, at the end of the whole experiment, 7 users answered Scaled-Arrows, 3 chose Halo and 3 chose Stretched-Arrows.

6. CONCLUSION AND FUTURE WORK

The user study in this paper shows that Halo and arrow-based visualizations do not differ significantly in supporting users to perform simple spatial tasks such as finding the closest off-screen object. However, it is worth observing that these tasks do not necessarily entail the need for the user to be spatially aware of the information space explored, since an automatic support of the task execution could also be implemented and provided by the mobile application itself (e.g., a navigation system).

It is more interesting to look at the differences found among visualizations for complex tasks, such as Order, where the cognitive burden put on the user for identifying direction and distance of the off-screen locations was also increased by the need to keep in mind the items (arcs/arrows) already tapped to form the correct sequence requested. The higher complexity of the task presents some analogies with more challenging conditions of use of the mobile application, such as those in which user's attention is distributed among multiple cognitive processes (often carried out on the move). This is the case, for example, of first responders' task conditions, where it can become important for the user to grasp information at a glance from the displayed map, on which basing quick decision-making or coordination activities (e.g., in the context of an emergency). Our findings suggest that when the cognitive demand on the user is higher (as when search and memory operations need to be run in parallel), the type of visualization employed can make a (significant) difference for what concerns the support provided to the user: according to our results, arrow-based visualizations outperform Halo, and this is particularly evident in the case of several off-screen objects to be taken into account (cluttered configurations).

As for the better support in terms of precision in the Locate task, we cannot derive the same conclusions on cognitive load, since the task execution was less demanding in terms of the user's working memory (e.g., there was no need to keep in mind the off-screen objects' locations already identified, since they were marked on paper).

So far, the contribution provided by the results of this study is particularly relevant to the design of mobile applications that are meant to support activities where, for different reasons (ranging from safety to system flexibility or context unpredictability) it is crucial for the user to acquire *spatial awareness* of the information space explored, so as to better exert direct control and decision-making over it.

At least two directions for future research can be envisaged. The first concerns a more extended and precise investigation on possible variations of one or more of the off-screen objects encodings (as reported in Table 1). This would enable designers to understand and establish how to optimize them for presentation on small screen devices, according to specific user and application requirements. Our user study constitutes a first step in this direction, as far as arrow-based visualizations are concerned, but more experimentation is currently needed.

A second research direction we intend to pursue, concerns the study of the implications of our results for other existing mobile systems interfaces that support everyday activities such as navigation, decision-making and coordination. For example, inspiration for further design solutions and user testing could be derived from other approaches recently proposed, like the approach exploited by Geocaching [10] to describe distance and direction of caches disseminated by users in a certain geographic area. With reference to the adoption of mobile applications to support tasks in stressful conditions, it would be interesting to assess if visualization of off-screen objects would enable users to exert higher control and get a more flexible exploitation of location-aware systems' features, such as the retrieval of location-specific content [3].

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".

8. 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. *Proc. ACM CHI Conference on Human Factors in Computing Systems (CHI '02)*, ACM Press, New York, 2002, 259-266.

[2] Baudisch, P., and Rosenholtz, R. Halo: A Technique for Visualizing Off-Screen Locations*. Proc. ACM CHI Conference on Human Factors in Computing Systems (CHI '03)*, ACM Press, New York, 2003, 481-488.

[3] Betts, B., Mah, R., Papasin, R., Del Mundo, R., McIntosh, D.M., and Jorgensen, C. *Improving Situational Awareness for First Responders via Mobile Computing*. Technical Report NASA/TM-2005-213470, Smart Systems Research Laboratory, NASA Ames Research Center, October 2005.

[4] Björk, S., Holmquist, L.E., Redström, J., Bretan, I., Danielsson, R., Karlgren, J. and Franzèn, K. WEST: A Web Browser for Small Terminals. *Proc. ACM Conference on User Interface Software and Technology (UIST '99)*, ACM Press, New York, 1999, 187-196.

[5] Buyukkokten, O., Garcia-Molina, H. et al. Power Browser: Efficient Web Browsing for PDAs. *Proc. ACM CHI Conference on Human Factors in Computing Systems (CHI '00)*, ACM Press, New York, 2000, 430-437.

[6] Chen, Y., Ma, W.Y., and Zhang, H.J. Detecting Web Page Structure for Adaptive Viewing on Small Form Factor Devices*. Proc. International Conference on World Wide Web (WWW '03)*, ACM Press, New York, 2003, 225-233.

[7] Chittaro, L. Visualizing information on mobile devices. *IEEE Computer*, *39*, 3 (2006), 40-45.

[8] Chittaro, L., and Burigat, S. 3D Location-pointing as a Navigation Aid for Virtual Environments. *Proc. International Conference on Advanced Visual Interfaces (AVI '04)*, ACM Press, New York, 2004, 267-274.

[9] Combs, T.T.A., and Bederson, B.B. Does zooming improve image browsing? *Proc. ACM Conference on Digital Libraries*, ACM Press, New York, 1999, 130-137.

[10] Geocaching Official Site, http://www.geocaching.com, 2006.

[11] Gutwin, C., and Fedak, C. Interacting with big interfaces on small screens: a comparison of Fisheye, zoom, and panning techniques. *Proc. Conference on Graphics Interface*, Canadian Human-Computer Communications Society, 2004, 145-152.

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

[13] Kaptelinin, V. A comparison of four navigation techniques in a 2d browsing task. *Conference companion on Human factors in computing systems,* ACM Press, New York, 1995, 282-283.

[14] Karstens, B., Rosenbaum, R., and Schumann, H. Presenting Large and Complex Information sets on Mobile Handhelds. *E-Commerce and M-Commerce Technologies*, 32–56. IRM Press, Hershey, London, 2004.

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

[16] Plaisant, C., Carr, D., and Shneiderman, B. Image-browser taxonomy and guidelines for designers. *IEEE Softw*are, *12*, 2 (1995), 21-32.

[17] Rauschenbach, U., Jeschke, S., and Schumann, H.. General rectangular fisheye views for 2D graphics. *Computers & Graphics*, *25*, 4 (2001), 609-617.

[18] Robbins, D.C., Cutrell, E., Sarin, R., and Horvitz, E. ZoneZoom: Map navigation for smartphones with recursive view segmentation. *Proc. International Conference on Advanced visual interfaces (AVI '04)*, ACM Press, New York, 2004, 231-234.

[19] Rockstar Games, *Midnight Club II*, www.rockstargames.com, 2003.

[20] Rosenbaum, R.U., and Schumann, H. Grid-based interaction for effective image browsing on mobile devices. *Proc. SPIE Int. Soc. Opt. Eng*. 5684, 2005, 170-180.

[21] Trevor, J., Hilbert, D. M., Schilit, B. N., and Koh, T.K. From Desktop to Phonetop: A UI for Web Interaction on Very Small Devices. *Proc. ACM Conference on User Interface Software and Technology (UIST '01)*, ACM Press, New York, 2001, 121-130.