

Spatial Interactive Visualization on Small Screen

Tero Hakala
Nokia Research Center
P.O. Box 100
33721 Tampere, FINLAND
+358 718035411
tero.hakala@nokia.com

Juha Lehtikainen
Nokia Research Center
P.O. Box 100
33721 Tampere, FINLAND
+358 718035450
juha.lehtikainen@nokia.com

Antti Aaltonen
Nokia Research Center
P.O. Box 100
33721 Tampere, FINLAND
+358 718035636
antti.aaltonen@nokia.com

ABSTRACT

The amount of data stored in personal digital devices increases rapidly as their memory capacities increase. These devices are usually equipped with relatively small displays, which makes presenting the information a challenge. We set out to explore the spatial design space for small screen user interfaces by incorporating additional dimensions into the visual representation, and investigate techniques that may be used to display more information at once. We focus on interactive visualization, with a document manager as a target application. We present the design factors and a simulated application running on a desktop computer. We also report a formative usability study with promising results.

Categories and Subject Descriptors

H 5.2 [User Interfaces]: Graphical user interfaces, interaction styles, screen design. D.4.3 [File Systems Management]: Distributed file systems. D.4.8 [Performance]: Modeling and prediction; simulation.

General Terms

Design, performance and measurement

Keywords: Mobile devices, document management, interactive visualization, depth cues, 3D user interfaces

1. INTRODUCTION

One of the most frequent tasks with the desktop computers is document (or file) management, but still it may be one of the most difficult things to learn for the novice users. Similarly, as the memory capacity of mobile devices increases the need for better file management tools for them emerges. However, the characteristics of the mobile devices, such as small screen, poor interaction devices and limited computational power [16], makes the design and implementation of these kinds of tools even more demanding task than with desktop computers.

We set out to study the issues related to applying spatial cues to a document management application on small screen devices, including visualization, graphical design, and interaction. Therefore, the purpose of this paper is not to compare spatial design with alternative solutions for file management, but rather to explore the design space.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MobileHCI'05, September 19–22, 2005, Salzburg, Austria.
Copyright 2005 ACM 1-59593-089-2/05/0009...\$5.00.

The rest of the paper is organized as follows. First, we review the relevant related work, followed by a presentation of our approach and design, including detailed description of the graphical design.

We also consider the possibilities and limitations of small screen graphics. We then describe a preliminary usability study, followed by discussion and conclusions.

2. RELATED RESEARCH

Our approach to the design problem was three-folded. First, we wanted to review the previous document management methods that use either two or three-dimensional visualizations. Then, the experiences of the earlier information visualization techniques that are suitable for small screens were studied. Third, we wanted to learn the results of the earlier studies of the effect of 3D graphics and spatial cognition.

2.1 Document Management

Suitable ways for presenting file structure and information in desktop computing environment have been studied for decades and different kinds of visualization techniques have been presented. Perhaps the most common way for document management is to use tree visualization, such as the visualization of a file structure with Microsoft Windows Explorer. In addition, other two-dimensional visualizations that rely on more graphical presentation of the data exist, such as the tree map [11] and the hyperbolic browser [13]. Three-dimensional tree structures, such as cone tree [17], have also been presented.

Data Mountain [18] is a file management technique that uses more than two dimensions in order to take advantage on spatial memory. Data Mountain allows users to arrange ‘thumbnail’ images of documents on an inclined 3D plane.

An approach very similar to ours is File System Navigator, or FSN [21]. It provides a view to the files as a 2D tree layout projected on a 3D ground plane, and uses a spotlight metaphor for selection. It runs on desktop computers, though.

Although MAPA [7] was designed for displaying the structure of large Web sites (and thus, improving the navigation) and not directly for document managing, it is an example of displaying a tree structure in a 3D space on inclined plane, where the pages are displayed as squares. A page that has a focus is placed at the front and its child pages form a row behind it. Then each child page and its children behind it form a single column. This provides a layout mechanism with minimal visual clutter.

2.2 Information Visualization

Basically there are several techniques for presenting large information spaces in a compressed form. Noik [15] has classified these as follows: implicit (use of perspective), filtered (removal of

objects with low degree of interest), distorted (size, shape and position), and adorned (changing attributes such as color).

Many of the current visualization methods aimed at small screens rely on distorting the view; *i.e.* the viewpoint information is manipulated in a way that enables seeing important object in detail, while preserving the broader context in which the object belongs. For example, the rubber sheet [20] is a view distortion technique that allows the user to choose areas on the screen to be enlarged. Therefore, the whole information space can be displayed at once with very low amount of detail. Should the user want to see some areas in more detail, he or she stretches the rubber sheet on the particular screen location, effectively zooming into the information on that area. In electronic document reading, displaying the overview and the detail at the same time is also found more beneficial than the traditional linear format [10].

Zooming and zoomable user interfaces (ZUI), such as PAD++ [1], are another way of presenting large information spaces even on a small screen. The information is presented on a 2D space, and the user can pan the view as well as zoom in and out of any part of that space. The view transition is animated to maintain the broader context of the local detailed information. Combs and Bederson [6] studied image browsers and found out that their system, based on a ZUI method (as well as 2D thumbnail grid), outperformed 3D browsers in terms of retrieval time and error rate.

Kamba *et al.* [12] present way of saving screen space by using pop-up type of interface components (the controls are hidden until needed), and movable interaction elements (the elements can be arranged on the screen so that maximum work area is retained).

Fitzmaurice *et al.* [8] present probably the first study towards 3D (virtual reality in this case) in small terminals. They simulated a handheld portable display by attaching a 4" display to a Silicon Graphics workstation. The graphics were displayed on the workstation screen. A video camera captured the large screen and displayed the equivalent image on the small display. Hence, they were able to simulate a portable display with computational capabilities of a powerful workstation.

Gutwin and Fedak [9] have studied different techniques for accessing information on a small screen: panning, zoming, and fish-eye (distortion). They found out that for web navigation, the fisheye-view was the most efficient.

2.3 3D and spatial cognition

Since the real world we live in is three-dimensional in nature, it may be easier for us to think in terms of three dimensions than in two with regard to the computers as well. Further, humans seem to have a very strong spatial sense that helps us in orienting and navigating in the real world and when 2.5D graphics are applied to computing, these same skills could help us to adopt to using the computer (see *e.g.* [8]). Robertson *et al.* [18] make an important notion that the benefit that the spatial cognition provides has been used in many two and three-dimensional document management techniques. These kinds of presentations aid the user's ability to recognize and understand spatial relationships of the objects and especially the 3D interfaces enable displaying more data without causing additional cognitive load, since the changes of the objects size that is caused by perspective indicate spatial relations at a distance.

The performance of three-dimensional visualizations has been studied, but some of these results discourage to exploit the use of the 3D since the performance does not either differ or improve

from the 2D (see *e.g.* [3]). However, contradictory results have also been reported, *e.g.*, by Ware and Franck [22].

In the case of two- or three-dimensional visualization of document management, the results that Cockburn and McKenzie [4] found are discouraging in terms of performance. They noticed that people performed (although not significantly) better with a 2D interface. However, the subjective measures showed that the users felt the 3D interface more natural and more preferable. In addition, the authors provide more evidence in their latter study [5] that the inclined plane (they called that 2.5D condition) performs better than the 3D in document management.

Incorporating 3D into small screens requires some thinking. The design space is rather limited in terms of both information visualization and interaction. The small number of available pixels alone makes it hard to use many of the effects required to create a 3D scene. Further, since the physical size of the screen is also small, the resulting images are either very small, or only a few can be displayed simultaneously. As a result, many researchers have concluded that it is not feasible to have any 3D graphics at all on mobile terminals (see *e.g.* [2]).

Another fact is that actual 3D object manipulation has not been done on hand-held terminals at all. One may ask if it is even a reasonable task to try out. Most probably complex 3D object and scene manipulations require larger displays, either physical or virtual (such as head-worn displays). Not to mention the requirements for the rendering hardware. These issues are considered in [19].

One more aspect is purely aesthetic. One may have flat 2D graphics that are operated on a 2D input device; however, the objects that are manipulated may have 3D look in order to appear more pleasant. To achieve cosmetic 3D, one can use several depth cues. However, *e.g.* Mullet and Sano [14] state that the 3D should not be used only for decoration.

In conclusion, one may state that previous research does not provide strong argumentation for the benefits of using more than two dimensions in a graphical user interface as far as task efficiency is concerned. However, we were interested in exploring the spatial design space for small screens in more detail, particularly to gather information regarding the challenges and promises that spatial cues may imply. We chose document management as a concrete application domain for the exploration.

3. OUR APPROACH

We began to design a document managing system called Space Manager for a mobile terminal that combines some of the benefits of the previous studies. It uses tree view, which is a familiar way for most of the users for presenting the file structure. However, instead of the normal 2D tree visualization, we decided to add also the depth dimension in order to reap the benefits of spatial memory.

Bringing depth cues into small screens requires very careful crafting. In order to find the optimum techniques and enhancements, one has to create both conceptual still images and animated sequences. Our study is no exception: we created dozens of still images and a number of different animated versions of the concept until we were satisfied with the result and were able to proceed. In this section, we will briefly describe the design goals, constraints, and the application itself, followed by the design principles and why they were adopted.

The purpose of this method is to allow the user of a mobile terminal to get a spatial overview of the contents of his/her mobile terminal file structure. This means that instead of a plain list that contains all the files and folders on the terminal, the structure is presented in a more informative way in three dimensions. This implies that each folder receives an additional attribute – its location in space. Hence, a folder has a structural location in the folder hierarchy, and it also has a spatial location that defines its location on the screen (of course, higher-level folders still precede those that are deeper in the hierarchy).

The design is mostly constrained by hardware factors. Our target platform is Nokia 7650, an imaging mobile phone equipped with a built-in digital camera (VGA resolution) and 176×208 pixel display capable of using 4096 colors. The input device is a four-way joystick with a selection button in the middle (Figure 1). The two soft keys can be used for additional operations; however, we avoided using these buttons as much as possible in order to keep the concept more general. Further, the available space for the content is 176×144 pixels, which sets another remarkable design constraint. Making this concept work on such a small display ensures that it can be scaled up.

The design goals may be stated as follows:

- G1. Present visually the contents of the file structure stored in a terminal, using depth cues where appropriate in order to produce a 3D-like view.
 - G2. Aim at displaying as much of the structure as possible at a time, i.e. avoid panning.
 - G3. Retain the clarity and readability of the fonts.
- Apply “in-place activation” where possible.



Figure 1. The Nokia 7650. The keypad can be slid out from behind the display, revealing the camera lens on the back

In brief, we ended up into a spatial file manager, called SpaceManager (Figure 2). The design is based on two levels: There is the *navigation level*, which shows the folder structure only. The folders are displayed so that the root folder resides on the bottom edge of the screen, and the subfolders are displayed above it. The user can navigate in this structure by using the four joystick directions. When a folder is opened, the user enters the

folder level, where he/she gets access to the contents of the folder and tools that are relevant for that folder.

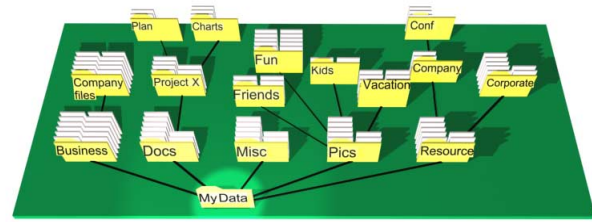


Figure 2. The Space Manager concept

When entering the folder level, the folder is opened and zoomed in on the screen. The folder opens like a box with the cover open and facing the user (Figure 3). The transition from the navigation level to the folder level is smooth and animated, indicating that the folder is opened in-place with the viewpoint brought closer to the folder. At this point, the user has four basic options to make. They can click the up button to start browsing the individual files in the folder (i.e., to go inside the folder); or click down to return to the previous state (i.e., to close the folder and zoom out to the navigation level). Left button allows browsing the contents of the folder in regular 2D filelist mode, and right button activates content-specific folder tools.

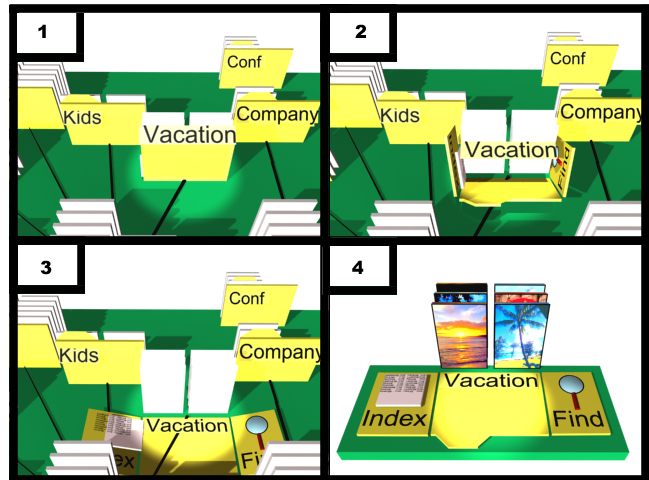


Figure 3. The animation sequence when a folder is opened

Once the user is browsing the files, s/he can use the four directional buttons to reach any file rapidly (Figure 4). The files are flipped as the user is browsing, just like browsing unsorted paper photos that are stored in a shoebox. Selecting a file opens it. This causes the view to zoom into the file (i.e., the file is opened in-place instead of explicitly launching another application). Once the user chooses to exit/close the file, the view is returned back to the state it was in before opening the file. When browsing the files in a file list mode, the Exit button (left softkey in the terminal) is used to return to the previous level.

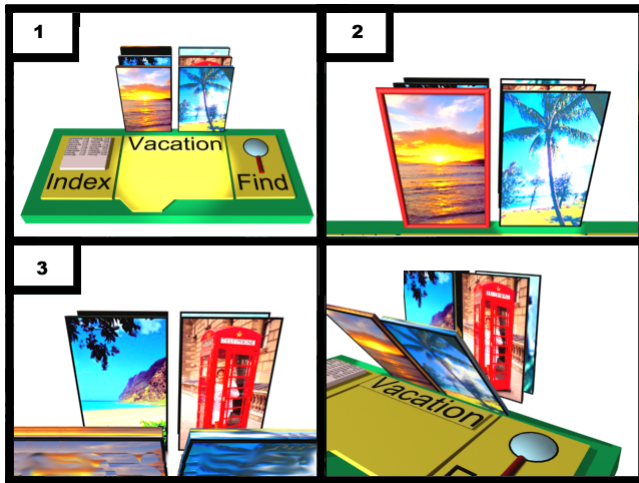


Figure 4. Browsing the files. 1) The folder open. 2) The user has clicked “up” and starts browsing files. 3) Files are flipped when browsing (with “up” and “down”)

3.1 The Flashlight Metaphor and Navigation Model

One of the crucial aspects was the choice of a selection method. There are two basic choices available: either the user is controlling a pointer (e.g., a cursor) that moves around the structure, or the user is manipulating the structure itself. In the latter case, a static location on the screen resembles the selection area: the part of the structure that is currently in that area can be selected.

We chose the former approach and ended up to a flashlight metaphor: the user is holding a flashlight in his/her hand, and can point the light on any neighboring folder on the surface. The highlight can be moved to any of these folders by using the four directional buttons. By clicking the up button, the highlight is moved to the closest folder on the next level in the hierarchy. The user can choose either to move deeper into the structure by further clicking the up-arrow, or alternatively move left or right in order to move between the folders on the same level. In case all the folders do not fit on the screen at once, the view pans as the user is about to move out of the screen. By clicking the down button, the user can return to the previous level, and finally reach the root level by clicking the down button again. Any folder can be opened by clicking the *selection* button.

The working principle of the flashlight is shown in Figure 5, where the user is moving from *MyData* to *Misc* and then to *Pics*. Note the slight view transition during the movement in order to keep the target object visible.

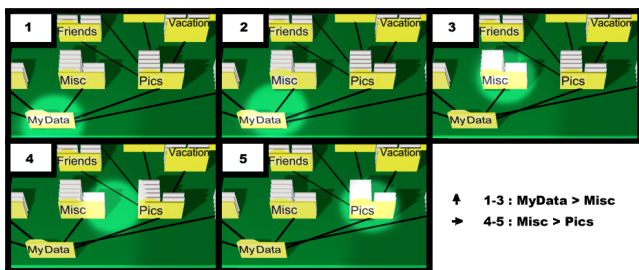


Figure 5. The flashlight metaphor

There are obvious reasons for choosing a brightness-based metaphor. The most important of these, of course, is the ability to conserve screen real estate: no additional pixels are needed for the selection indicator, but only the pixel brightness values are modified. Further, flashlight is a common artifact in everyday life, making it easy to comprehend its behavior. Adding a “light source” is also one means of increasing the feel of additional dimensions on the screen.

Our navigation model implies an important difference compared to any other hierarchical navigation method that is based on discrete input. In our model, the navigation is not restricted by the folder structure, but the navigation occurs on spatial level. In other words, the user is able to move to any spatially neighboring folder even if it does not belong to the same hierarchy with the current folder.

While navigating, the viewpoint is fixed, *i.e.* the user is not able to move freely in 3D space. The only viewpoint changes take place at predetermined occasions, such as when opening a folder. The choice to restrict the viewpoint was deliberate, otherwise we would have needed quite more complex interaction devices or techniques.

3.2 On the graphical model

Creating a 3D graphical model that looks pleasant to a user is not trivial, especially so on a small screen. There are many factors, which have to be taken into consideration when creating a 3D environment, and using badly (or not using at all!) one of the factors will ruin the whole graphical model. People will recognise and explore 3D environments only if they look realistic enough.

In our model the first objective was to find and define the proper camera angle. Many experiments were made in the beginning. At first, the angle was as low as 23°. With this viewpoint, the folders in the forefront were clear and easily readable, but the subsequent folders were more messy and created undesired visual noise on the whole scene. We tried to solve this by adding fog to the background in order to lessen the clutter (Figure 6). The resulting scene appeared visually better, but deliberately decreasing the clarity of a part of the screen is not a preferred method when the aim is to have more information on the screen.

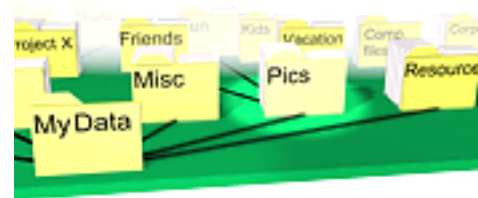


Figure 6. An experimental “fog-model” version of the Space Manager

In order to increase the visual appearance further, we continued experimenting with the camera angle. We lifted the camera up to 60°. This resulted in a clearly improved scene. Since the objects are viewed from higher, the screen space taken by an object is less than in the previous model. Consequently, the objects do not overlap as often as before, resulting in a more clear overall view. Likewise, locating the flashlight became easier due to the decreased overlapping. The folder labels were forced to tilt straight against the camera to avoid font distortion (Figure 7). If the labels were not folded, the text would not be readable anymore, especially when the camera angle is as high as ours.

The use of light and shadow is important; these factors will create a realistic impression of the 3D world. They provide a final touch to the depth appearance of the view. Without light and shadow the impression is more or less “flat”. The light source should be placed beside either side of the viewing point and vertically either up or down of the viewing point, otherwise the shadow is minimal and the feeling of the depth will decrease. In our model, target light comes from the left-down direction, which creates quite a natural feeling. Changing the light direction a lot will cause inconvenient results: long shadows or light saturated details in a model. Also, the light hotspot and falloff radius have to be relative to the object size.



Figure 7. The font used in the folder labels is tilted straight against the camera

The colors we used are yellow for the folders and green for the ground. The colors could be something else, of course, but these colours are already familiar to the user, yellow folders from the PC-interface and green as grass color in a ground. The background color is blended from black to white downwards; this makes the whole model look like it is floating in space.

A small screen with small 3D elements is a challenge, even more so when there is also text involved. The majority of the text on the screen should always be readable. When texts are in the background in a 3D graphical model but not well readable, fonts could still give valuable hints to the user. A user could guess the meaning if he/she is already familiar with the content. A text in a very small and not readable size should not be used at all; it will give no information but increases visual noise. The folder and file shapes in our 3D graphical model are generally known from traditional desktop interfaces, which help the user to recognise them straight away.

The focal length in our 3D graphical model is equivalent to SLR cameras 35 mm. This means it will emphasize the forefront a little. Increasing the focal length will noticeably change the view: the folders seem to be more compressed to each other (see Figure 8). When decreasing the value, the forefront will get more space, and aberration in objects' shape will increase.

The viewing angle in our model, 60 degrees, is quite high. The reason for this is to maximize the amount of visible folders. Folders are still well recognizable even from this view of angle. In general, the forefront folders should not overlap, while some background folders may overlap a little. Text is shown always straight to the viewer, distorting text in a small size will not work.

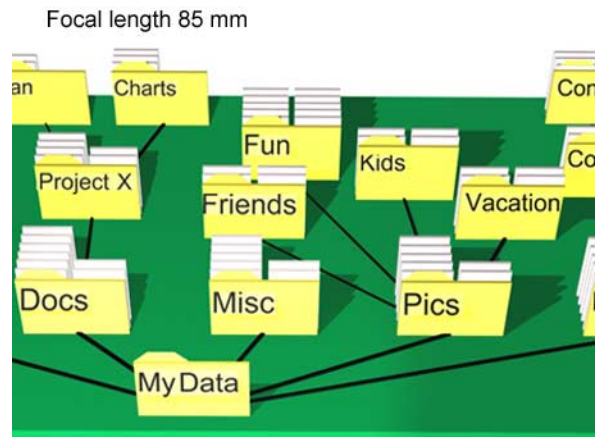
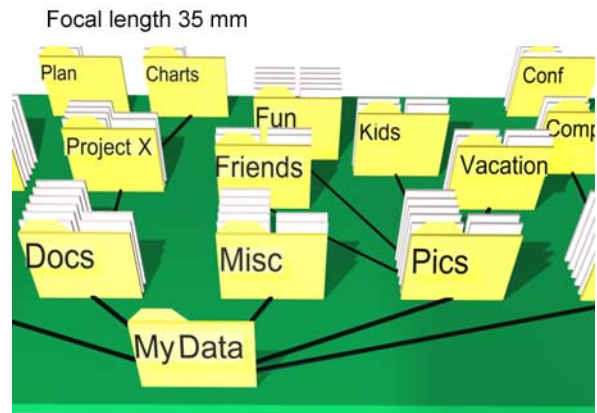
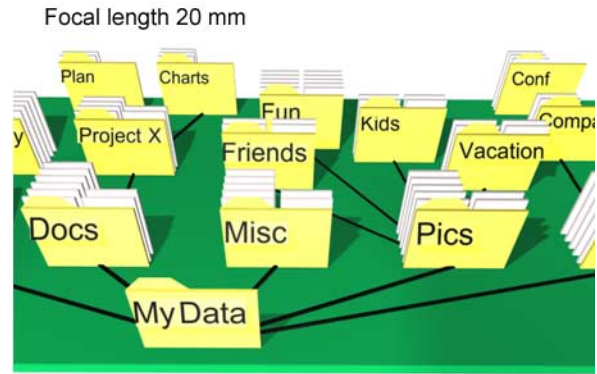


Figure 8. Three shots with different focal lengths but an equal camera angle. Notice the changes in e.g. the edge of the Docs folder

An overview of the graphical model is shown in Figure 9. The top-left icon denotes the camera, the boxed cross in the middle is the omni light, and the arrow close to the plane is the target direct light.

4. EVALUATION

In order to gain initial response from the users, we arranged an evaluation. The goals of the evaluation were to find subjective

opinions on using a document manager like ours, and to find any major usability problems.

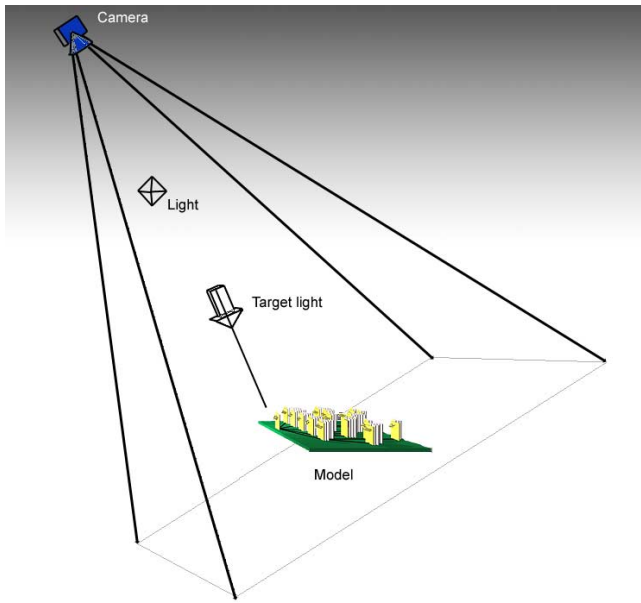


Figure 9. An overview of the graphical model

4.1 Participants

There were 10 subjects with various backgrounds, including engineers, students, and a sociologist. The ages varied between 20 and 38 years. None of the participants had seen Space Manager prior to the experiment.

4.2 Apparatus

The evaluation was performed on a Windows 2000 workstation. We replaced the standard PC keyboard with a numeric keypad with all but directional and selection keys removed. We implemented an interactive 7650 simulator in Macromedia Director v8.5. The images required in the simulation were constructed with 3ds Max v4.26 by Discreet, and retouched in Adobe's Photoshop 7.0. The file structure used in the evaluation is as in Figure 2.

4.3 Procedure

A test session began with an introduction. The subject was explained the idea of Space Manager and the way the simulator is controlled. The test procedure was also explained.

The introduction was followed by 10 test tasks. The tasks were as follows (comments for the tasks are in italics):

1. Enumerate orally the immediate subfolders of "MyData" (*How does the user move? First up and then left/right or directly left/right?*)
2. Move back to "MyData" (*Does the user do this with a single click or by using the same route?*)
3. Move to "Pics/Vacation" and open a file called "NewYork.jpg" (*Moving, opening a folder, browsing files*)
4. Go back to "MyData" (*Moving*)
5. Enumerate orally the subfolders of "Resource" (*This task prepares for the task #8*)

6. Move to "Docs/Project X" (*Does the user move sideways or via the root folder?*)
7. Move to "Pics/Vacation" (*This task prepares for the task #8*)
8. Move to "Resource/Comp" (*Does the user move sideways or via the root folder?*)
9. Go back to "MyData"
10. Enumerate orally the subfolders of "Business"

Finally, the emulator was closed and the user was asked to draw the file structure as s/he remembered it by hand on a sheet of paper. The user was not told about this portion of the evaluation until the interactive tasks were finished. This task verified whether the user had implicitly learned the folder structure.

The tasks were performed one at a time. Once a task was finished, the user clicked a button, after which the next task was shown on the screen. The user was asked to think aloud while navigating.

After the drawing sessions, there was a semi-formal interview. As a whole, the test took around 30 minutes. The results were collected by observing the usage of the actual test tasks, recording the results of the interviews, and visually analyzing the drawings.

5. RESULTS

All test sections provided promising results. The usage sessions revealed that the users adopted the navigation rapidly. They also commented that the ability to move sideways greatly facilitates navigation, especially in cases where the folders did not belong to the same tree branch.

One aspect concerning navigation was the root folder. Even though with current model it is easy to return to the root, many users would have wanted a 'home' key that would return the focus immediately to the root folder.

There were evident problems with the navigation when it was not certain which folder is to receive the focus next (e.g., when clicking up and there were two folders above the current folder, it was not always obvious which of these folders is to receive the focus when moving up). One user commented: "*Sometimes it is very annoying, because you don't know beforehand where you will end up.*" Further, the users commented that when moving sideways, the next folder should always be visible.

The users appreciated the spotlight metaphor, especially when combined with folder opening. However, an experienced user should be allowed to turn the animation off in order to speed up navigation. Concerning opened folders, the users commented that the file browsing (Figure 4) works fine for images, but raised a concern whether it would work equally well with other file types.

The users learned to locate the folder locations easily. Both the actual test tasks and the drawing section confirmed this observation. This was true even though the structure was not familiar to any user. This was due to the fact that the spatial layout made the folder locations easier to remember, and it was possible to see more folders on the screen than with a regular treemap. The problem concerning the layout was that it was not always clear which folders belong to the same hierarchy level; the users commented that different levels should be indicated more clearly.

The combined results of the drawing section are shown in Figure 10, where the folders that the users remembered (i.e., both their name and the location) area drawn. The numbers in each folder indicate the number of users who remembered that folder. The

success of the actual test tasks would have indicated a higher number of remembered folders in this section. However, probably the users concentrated on the navigation and visual spatial layout, not the names of the folders, in the test tasks.

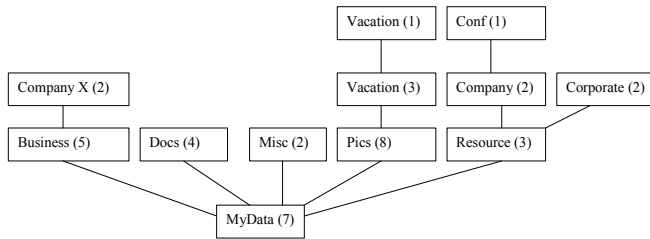


Figure 10. The results of the drawing section

On using 3D in general, some users commented that it facilitates navigation, while some thought that it does not make any difference, or it looks a bit strange. However, even these users commented that they would be ready to use this kind of visualization on their phone if it were available.

The overall appearance was considered appealing and clear.

6. DISCUSSION AND FUTURE WORK

Clearly, the problem that our current navigation model implies deals with predictability. When the user click the up button, it is not explicit that which folder receives the focus on the next level. Therefore, the model could be improved by adding visual cues showing the destination folder for each direction. Further, the algorithm could be improved by preferring the routes the user has taken before.

In addition to the PC simulator, we have also implemented a limited version of SpaceManager for the Symbian platform, using a proprietary Nokia 3D engine. This version remains to be finalized and thoroughly evaluated. Preliminary evaluations revealed that the navigation is very intuitive with the five-way joystick found on e.g. Nokia 7560, while displaying fonts in a readable fashion seems to be challenging.

Our results show that it is feasible, yet not trivial, to develop spatial user interfaces even for small screen devices. Therefore, more research on the topic is needed, including additional design alternatives and more thorough, comparative usability evaluations.

7. CONCLUSIONS

We have explored the spatial design space for small screen devices by developing a document manager application. The manager relies on the strong human spatial memory, and is based on a tilted plane that displays the document structure of the terminal. Any folder on the plane can be selected by operating a four-way input device; once the folder is activated, its contents are opened on the screen. We have also implemented a PC simulator and conducted qualitative usability studies. The results showed that the use of Space Manager can be learned rapidly, and that the spatial structure indeed makes it easier to perceive the contents of the phone. Spatial design for small screens deserves additional studies.

REFERENCES

[1] Bederson, B.B., and Hollan, J.D. Pad++: A Zooming Graphical Interface for Exploring Alternate Interface Physics. *Proceedings of the ACM UIST'94 Symposium on*

User Interface Software and Technology, 1994, ACM Press, 17 - 26.

- [2] Bergman, E. *Information Appliances and Beyond - Interaction Design for Consumer Products*. Morgan Kaufmann, 2000.
- [3] Carswell, C., Frankenberger, S., and Bernhard, D. Graphing in depth perspectives on the use of three-dimensional graphs to represent lower-dimensional data. *Behavior and Information Technology*, 10, 6, 1991, 459-474.
- [4] Cockburn, A., and McKenzie, B. 3D or not 3D? Evaluating the Effect of the Third Dimension in a Document Management System. *Proceedings of CHI 2001*, ACM Press, 434-441.
- [5] Cockburn, A., and McKenzie, B. Evaluating the Effectiveness of Spatial Memory in 2D and 3D Physical and Virtual Environments. *Proceedings of CHI 2002*, ACM Press, 203-210.
- [6] Combs, T.T.A., and Bederson, B.B. Does zooming improve image browsing? *Proceedings of the fourth ACM conference on Digital libraries 1999*, ACM Press, 130-137.
- [7] Durand, D.G., and Kahn P. MAPA: A System for Inducing and Visualizing Hierarchy in Websites. *Proceedings of Hypertext 1998*, ACM Press, 66-78.
- [8] Fitzmaurice, G.W., Zhai, S., and Chignell, M.H. Virtual reality for palmtop computers. *ACM Transactions on Information Systems*, 11, 3, 1993, 197-218.
- [9] Gutwin, C., and Fedak, C. Interacting with Big Interfaces on Small Screens: a Comparison of Fisheye, Zoom, and Panning Techniques. *Proceedings of the 2004 conference on Graphics interface*, ACM Press, 145-152.
- [10] Hornbæk, K., and Frøkjær, E. Reading of electronic documents: the usability of linear, fisheye, and overview+detail interfaces. *Proceedings of CHI 2001*, ACM Press, 293-300.
- [11] Johnson, B., and Shneiderman, B. Tree-maps: A space filling approach to the visualization of hierarchical information structures. *Proceedings of the IEEE Visualization '91*, IEEE 1991, 284-291.
- [12] Kamba, T., Elson, S.A., Harpold, T., Stamper, T., and Sukaviriya, P. Using Small Screen Space More Efficiently. *Proceedings of CHI 1996*, ACM Press, 383 - 390.
- [13] Lamping, J., and Rao, R. Laying out and visualizing large trees using hyperbolic space. In *Proceedings of the ACM UIST'94*, 1994, ACM Press, 13-14.
- [14] Mullet, K., and Sano, D. *Designing Visual Interfaces. Communication Oriented Techniques*. SunSoft Press/Prentice Hall, 1995.
- [15] Noik, E. A Space of Presentation Emphasis Techniques for Visualizing Graphs. *Proceedings of Graphics Interface 1994*, 225 - 233.
- [16] Rieck, A. Aspects of user interfaces in mobile environments. *Workshop on Information Visualization and Mobile Computing (IMC) 1996*, 26-27 Feb, Rostock, Germany.
- [17] Robertson, G., Mackinlay, J., and Card, S. Cone Trees: Animated 3D Visualizations of Hierarchical Information. *Proceedings of CHI 1991*, ACM Press, 189-194.

- [18] Robertson, G., Czerwinski, M., Larson, K., Robbins, D., Thiel, D., and van Dantzich, M. Data Mountain: Using spatial memory for document management. *Proceedings of UIST'98*, ACM Press, 153–162.
- [19] Roimela, K. *3D Graphics in Mobile Terminals*. M.Sc. Thesis, Tampere University of Technology, Finland, 2001.
- [20] Sarkar, M., Snibbe, S.S., Tversky, O.J., and Reiss, S.P. Stretching the Rubber Sheet: A Metaphor for Viewing Large Layouts on Small Screens. *Proceedings of UIST 1993*, ACM Press, 81-91.
- [21] SGI. File System navigator. Online document, available at http://www.sgi.com/fun/freeware/3d_navigator.html. Last checked Jul 29, 2004.
- [22] Ware, C., and Franck, G. Evaluating stereo and motion cues for visualizing information nets in three dimensions. *ACM Transaction on Graphics*, 15, 2, 1996, 121-139.