User and Task Analysis of Multi-Level 3D File Browser

Işıl Doğa Yakut, Cansın Yıldız
Department of Computer Engineering
Bilkent University
06800 Ankara, Turkey
{yakut, cansin}@cs.bilkent.edu.tr

***Abstract*— 3rd dimension enables tremendous improvements to data visualization and user interfaces. Therefore, it would be most unreasonable to stick with 2D designs. In this manner, we propose a 3D file browser which will immensely improve ease of hierarchical file operations.**

***Index Terms*— user interface, 3d ui, file browser.**

I. Introduction

With the growing amount of 3D environments in everyday usage of computers, the traditional user interfaces also need to evolve to representations that can be perceived better with enabling 3rd dimension. There are already successful examples of widely used 3D user interfaces. Some examples are Compiz Cylinder and Apple's Coverflow (see **Fig. 1**). In this project, we were inspired from the way how these applications represent data, and provide easy navigation using 3rd dimension. Hence, we address the question of how to enhance a fundamental part of every operating system; file browser.

Fig. 1. Some widely used 3D user interface samples: Compiz Cylinder and Apple's Coverflow.

This document is a final overview on the proposed multi-level 3D file browser system. Second section describes the user profile in detail. In third section, the overall usage of proposed file browser system is detailed with a task analysis. The report is concluded with a discussion on implementation of this system.

II. User Analysis

Five different aspects will be considered to analyze users; usage objectives, user roles, user characteristics, usage environment, user interface guidelines.

File browser is one of the main applications in everyday computer usage. Thus, there are no characteristics that will distinguish the target users, such as gender, age, etc. Also the level of expertise of potential users will vary greatly. Taking this into account, the aim of design is simplicity and ease of use. For example, duplicate buttons will be avoided therefore there will be only one clear way to perform a task. Similarly, the traditional context menu (right-click menu) design will be abandoned for a less cluttered design of floating icons around the selected item.

Although we proposed enhancements on existing file browsing experience, we will take advantage of user habits by sticking to visual characteristics of a Mac file browser (see **Fig. 2**). Such a browser uses a left-to-right hierarchical relation between folders, and it lists folder's contents as vertical lists of files.

A 2D pointing device is sufficient to interact with this design. Therefore, we have a lot of options as an input device, such as an ordinary mouse, a touch screen, and

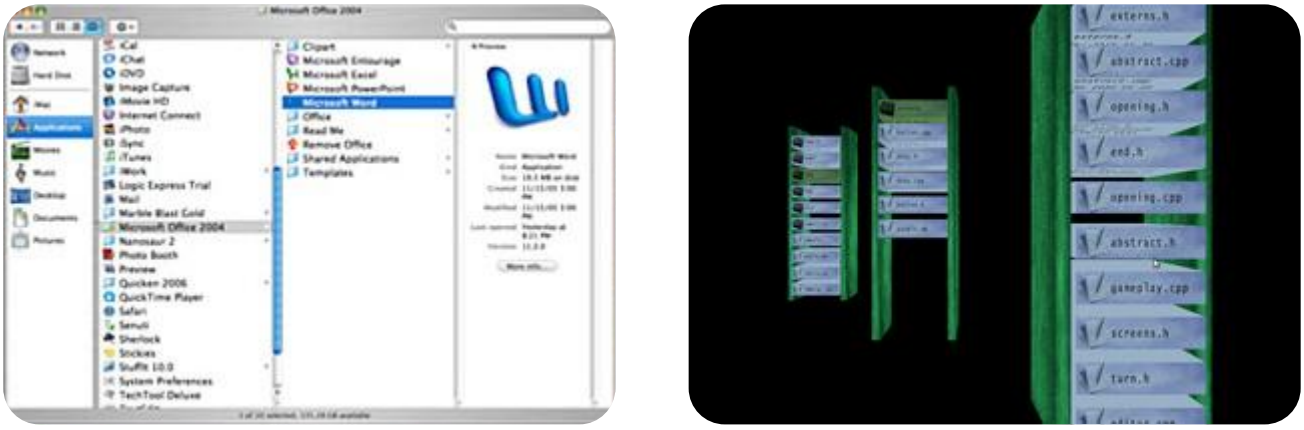

Fig. 2. Mac File Browser vs Our 3D Browser.
Note that both design uses a left-to-right hierarchical relation,
and they both list folder's contents as vertical lists of files.

even finger tracking. Even though the input is 2D, the graphics will be better represented with a 3D display. We chose to implement our file browser using a **multi-touch** display. Since this display lacks the required 120hz for shutter glasses, and we cannot lay a lenticular sheet on the device; our only option for the 3D visualization was to use **anaglyph**.

III. Features

We have five groups of tasks, namely; hierarchical view, hierarchical navigation, properties/thumbnails visualization, scrolling, and miscellaneous features.

A. Hierarchical View

Since our main concern is about making hierarchical operations simpler, we have an addition to existing windows design. We are always showing the parent folders to reveal their content as well. Of course, by the help of depth property of 3rd dimension, the ancestry relation of those folders is visualized (see **Fig. 3**).

B. Navigation

Our system's navigation task is more informative than usual. As user navigates to subfolders of focus folder, the hierarchical representation is preserved by the sliding motion of all content windows. Similarly, if user wants to navigate to (grand) parent folder of focus folder, one left touch gesture will assign it as the new focus folder. Also, it will be possible to directly navigate to a child of focus window, by performing a right gesture (see **Fig. 3**).

C. Properties/Thumbnail Visualization

It is possible to view additional information about or view a thumbnail of the file selected. This especially becomes handy since the user does not have to open the file to view its content. To give this additional information we take advantage of the occlusion (see **Fig. 4**). When the user clicks on an icon, the icon flips to

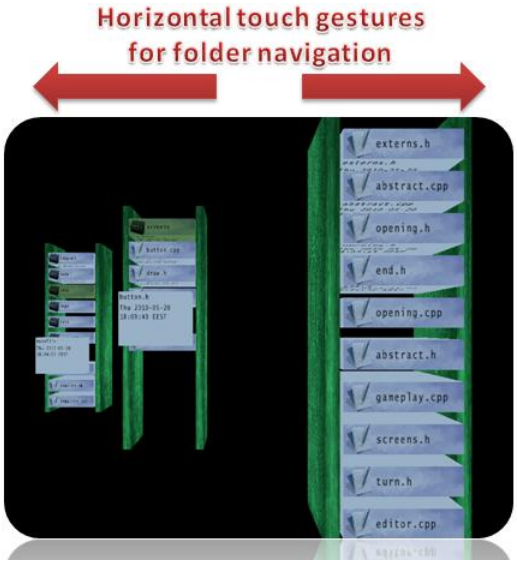

Fig. 3. Horizontal touch gestures for folder navigation.

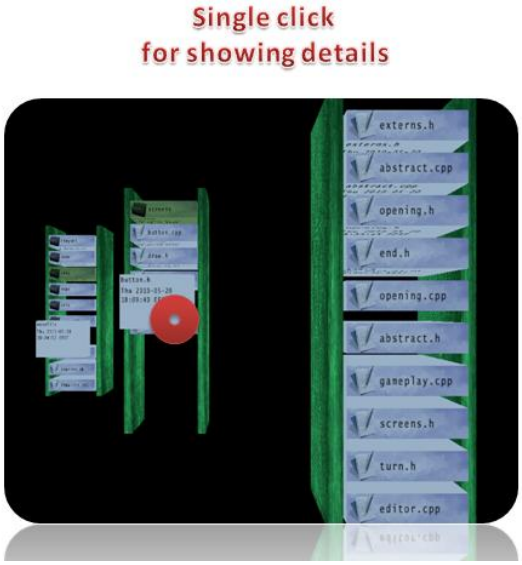

Fig. 4. Single click for showing details.
Notice how occlusion is used to hide properties when file is closed.

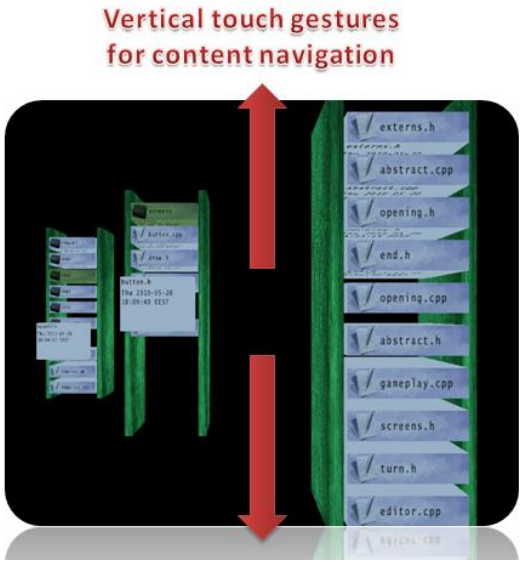

Fig. 5. Vertical touch gestures for content navigation.

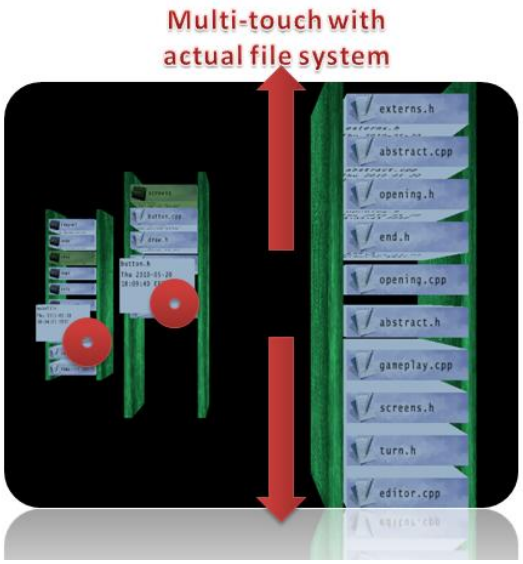

Fig. 6. Multi-touch with actual file system.

reveal it previously hidden faces to the user where the additional information can be presented at. Clicking it will close the thumbnail view and return the icon to its original state.

D. Scrolling

Scrolling through the contents of a folder is essential for many browsing applications. In our implementation we provided a gesture for this task. Instead of having an actual slide bar, users are able to scroll the current folders content by vertical touch gestures, similar to iPhone scroll (see **Fig. 5**). We avoided direct manipulation when scrolling since it would create a contradicting behavior with the hierarchical navigation gestures.

E. Multi-Touch

Multi-touch input devices are more common than ever. The attention that iPad device got indicates that multi-touch interfaces are here to stay. In addition to developing a system with multi-touch experience, it can be extremely useful when browsing through different levels of hierarchy. For example, selection of multiple files in different levels of hierarchy enables an enhanced user experience providing flexibility to the user (see **Fig. 6**).

F. Actual File System

Although the product of this project can be considered as a prototype application to test it's robustness the actual file system is used for file browsing. This proved us that the "3D file browser" is capable of handling many number of levels of hierarchy. For example a folder hierarchy of level 7, is easily loaded and presented to user. Also the users showed that using the actual file system increases the learning of the system.

IV. Comparison With Similar Systems

A. Tactile 3D

Fig. 7. Tactile 3D: A 3D file browser.

Tactile 3d is a file browser which takes advantage of the spatial positioning of the object, Depending on the humans ability to recall the position of an object. Simple file operations are provided with the tactile 3d system.

With tactile 3d system the user view a single folder only different from our system. Also, since it uses spatial position to characterize object, scrolling is not enabled which is a disadvantage when viewing a large folder with many entries.

Also, navigation is extremely hard because the system imitates the outer space navigation (see **Fig. 7**).

B. 3DOSX

Fig. 8. 3DOSX: Yet another 3D file browser.

3DOSX is a file system browser that uses three dimensions to view directory hierarchies. It supports a range of file system actions, such as copying, labeling, creating new folders, and ejecting disks. The developers claim that it is a quick and clean way to interact with hierarchies, but as can be seen in **Fig. 8** the environment get cluttered after a several level of hierarchies.

V. CONCLUSION

We have discussed our user and task analysis on the new introduced system in detail. We implemented a functional application that is actually able to visualize file system. The interface is implemented using OpenGL in GNU/Linux environment.

REFERENCES

[1] G. G. Robertson, J. D. Mackinlay, and S. K. Card, "Cone trees: animated 3d visualizations of hierarchical information," in CHI '91: Proceedings of the SIGCHI conference on Human factors in computing systems. New York, NY, USA: ACM, 1991, pp. 189-194. [Online]: http://dx.doi.org/10.1145/108844.108883

[2] Chin, R. 2002. Three-dimensional file system browser. Crossroads 9, 1 (Sep. 2002), 16-18. DOI= http://doi.acm.org/10.1145/571758.571765

[3] Andrews, K. 1999. Visualising cyberspace: information visualisation in the Harmony Internet browser. In Readings in information Visualization: Using Vision To Think, S. K. Card, J. D. Mackinlay, and B. Shneiderman, Eds. Morgan Kaufmann Publishers, San Francisco, CA, 493-502.