EyeExposé: Switching Applications with Your Eyes

Manu Kumar, Andreas Paepcke, Terry Winograd
Stanford University, HCI Group
353 Serra Mall, Stanford, CA 94305-9035
{sneaker, paepcke, winograd}@cs.stanford.edu

ABSTRACT

We present a technique for switching between active applications by using a combination of keyboard (or any other trigger) and eye gaze. In particular, our approach combines the use of a two-dimensional layout visualization for showing the user all open applications and the use of eye gaze tracking for selecting the desired window. Our studies show that this combination of gaze and the visual representation of active tasks allows users to switch between applications quickly and naturally. Users strongly preferred this technique of switching between applications compared to other alternatives.

Author Keywords

Task Switching, Application Switching, Eye Tracking, Gaze-enhanced User Interface Design.

ACM Classification Keywords

H5.2. User Interfaces: Windowing Systems.

INTRODUCTION

The keyboard and mouse have long been the dominant forms of input on computer systems. Eye gaze tracking as a form of input was primarily developed for disabled users who are unable to make normal use of a keyboard and pointing device. However, with the increasing accuracy and decreasing cost of eye gaze tracking systems [2, 6, 11, 17] it will soon be practical to use gaze as a form of input in addition to keyboard and mouse for able-bodied users – if the interaction can thereby by improved. The GUIDe (Gaze-enhanced User Interface Design) project [16] in the HCI Group at Stanford University explores how gaze information can be effectively used as an augmented input. In this paper we focus on using eye gaze for the purpose of application switching.

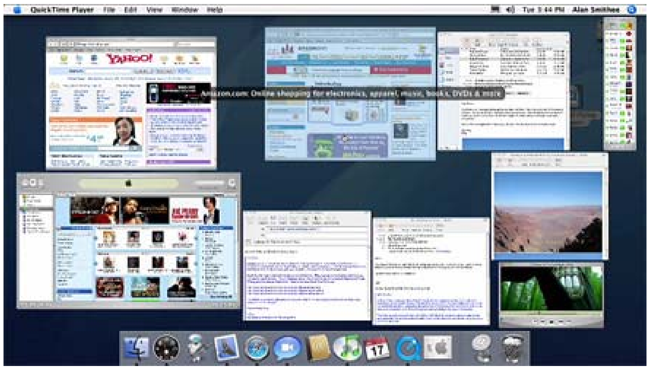

QuickTime Player File Edit View Window Help
Tue 3:44 PM Alan Smithee
Amazon.com: Online shopping for electronics, apparel, music, books, DVDs & more

Figure 1. Exposé view of open applications (image from http://www.apple.com/macosx/features/expose/).

Application switching is an integral part of our daily computing experience. Users are increasingly engaged in multiple tasks on their computers. This translates into a larger number of open windows on the desktop. On the average, users have 8 or more windows open 78.1% of the time [13]. While there has been extensive research in the area of window managers and task management [5, 9, 12, 19-21], there has been little innovation in commercially available desktop interfaces for switching between applications. Clicking on the iconic representation of the application in the taskbar/dock or using Alt-Tab/Cmd-Tab have been the de facto standard for application switching for several years. Probably the most notable advance has been the introduction of the Exposé [1] feature in Apple's Mac OS X operating system.

Exposé allows the user to press a key (default *F9*) on the keyboard to instantly see all open windows in a single view (Figure 1). The windows are tiled, scaled down and neatly arranged so that every open application is visible on the screen. To switch to an application the user moves the mouse over the application and then clicks to bring that application to the foreground. Every open application window is restored to its original size and the window clicked upon becomes the active window.

Windows Vista will also include new application switching features. The taskbar in Windows Vista displays live thumbnail views of open applications when the user hovers the mouse on the taskbar. Alt-Tab functionality has been updated with Windows Flip and Flip3D [4]. Flip allows users to view live thumbnails of the applications as they press Alt-Tab. Flip3D shows a stacked 3-D visualization of the applications with live previews and allows users to cycle through applications with the scroll wheel or the keyboard.

In this paper we introduce and evaluate a technique that uses eye gaze for the selection of the desired window in conjunction with Exposé-like visualization of the open application windows.

BACKGROUND AND RELATED WORK

Application switching has been necessary ever since computers were able to multi-task. In the days of command-line UNIX this was achieved with the commands *bg*, *fg* and *jobs*. With the advent and ubiquity of graphical interfaces and the desktop metaphor, application switching has become one of the most common operations performed on a computer.

The techniques for application switching can be categorized into three approaches: *Temporal*, *Spatial* and *Hybrid*. Temporal approaches sort windows based on their time of last access, and therefore the order in which the windows are shown to the user changes depending on which application was last used. Spatial approaches may use an initial ordering based on when the application was launched or where it is located on the screen. The relative order of applications in the application switching view does not change unless there is a change in the number of open applications or the spatial location of an application. Hybrid approaches use a combination of spatial and temporal characteristics of the open application windows.

Alt-Tab is a temporal approach. It organizes applications in the order in which they were last used. Users are able to cycle through the list of applications by sequentially stepping through the list until they arrive at the application they desire. Such techniques make best use of the user's temporal memory and make switching among a limited number of tasks very efficient.

The organization of window buttons on the Taskbar or in the dock follows the spatial approach. The user can access any open application directly by clicking on a button/iconic representation of the application. The location of the iconic representation of the application on the Taskbar is fixed and therefore this approach takes advantage of the user's spatial memory.

Exposé uses a spatial layout to arrange the open application windows in a visual representation. It also uses heuristics to keep the current application in the center of the visualization and to arrange windows based on their relative spatial position [15]. While the location of the windows in the Exposé view may change, it is relative to the spatial locations of the open applications.

Hybrid approaches, which use a temporal ordering but allow for random access (as opposed to the sequential access of Alt-Tab) are becoming more popular. The Windows XP PowerToy TaskSwitch [3] shows a thumbnail of the current application and allows users to either cycle through the open applications by repeatedly pressing Alt-Tab or to use the mouse to click on the icon for the desired application.

In EyeWindows [10], Fono and Vertegaal explore two window management techniques for *non-overlapping* windows which use the elastic windowing algorithm to spatially lay out application windows. The EyeWindows approach suffers from two major drawbacks. First, the technique is limited to use with non-overlapping windows (all techniques described previously allow overlapping windows). Secondly, switching between applications in EyeWindows requires windows to be zoomed in and out, which can be visually distracting for the user.

Several research systems [5, 9, 12, 19-21] have been proposed with novel window management and task switching techniques. Unfortunately, none of these approaches have seen widespread adoption in mainstream operating systems. Our gaze-based selection technique can complement the techniques in other research systems. For this paper we focus on task/window switching techniques in commercially available and commonly used operating systems.

DESIGN RATIONALE

The user's eyes provide an immense amount of information and in particular are the best source of information regarding the user's current attention and intention [22]. We hypothesized that it would be preferable to switch between applications simply by looking at the application the user wants to switch to – a concept similar to EyeWindows.

Exposé in Mac OS X provides a well established and highly usable technique for switching between applications. Unfortunately, we did not find research papers that provide a scientific evaluation of different application switching techniques (Alt-Tab/Cmd-Tab vs. Taskbar/Dock vs. Exposé vs. Flip/Flip3D). Anecdotal evidence, however, suggests that the Exposé approach is superior for random access to open applications, while the Alt-Tab/Flip approach is superior for access to the last used application.

To use Exposé, users press a hotkey (*F9*) and then use the mouse to point at and click on the desired application. Using this approach requires both the keyboard and the mouse, whereas with the Alt-Tab approach, the user can switch applications using only the keyboard. Exposé does allow users to activate application switching by moving the mouse to a designated hotspot (one corner of the screen) and then clicking on the desired application. This still requires users to move their hands from the keyboard to the pointing device.

In general, using eye gaze for pointing and selection tasks is not desirable since the accuracy of the trackers limits accurate pointing and selection to a visual angle of approximately 0.5-1^o (16-33 pixels on at 17", 1280x1024, 96 dpi screen at a 50cm viewing distance [7]). Though some eye-trackers claim a sub-1^o accuracy, in practice the

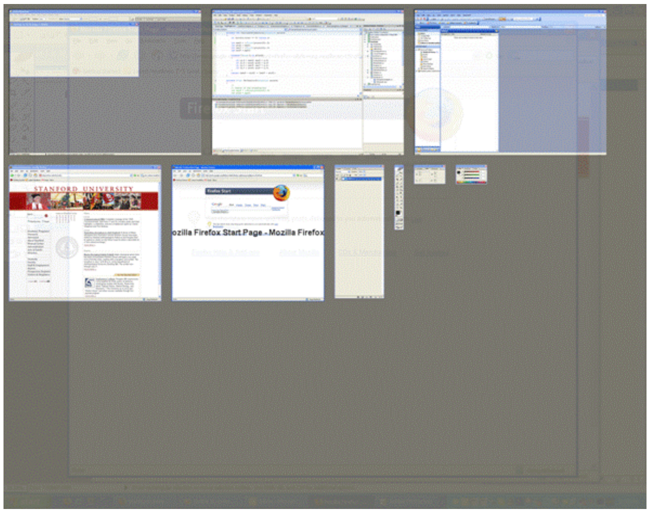


Figure 2. Pressing and holding the EyeExposé hotkey tiles all open applications on the screen. The user simply looks at the desired target application and releases the hotkey.

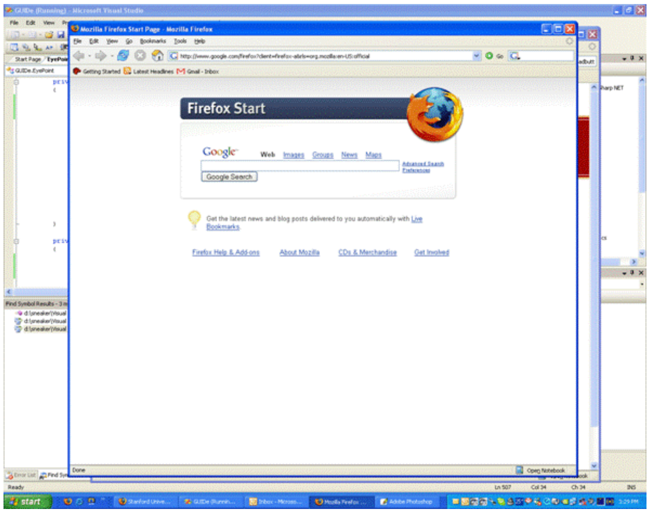


Figure 3. Releasing the hotkey restores the windows to their original size and brings the selected application to the foreground.

jitters in eye-movement or tracking cause the data to be noisy and increase the overall error [7, 14]. However, the size of the tiled windows in Exposé is usually large enough for eye-tracking accuracy to not be an issue.

EYEEXPOSÉ

Our system, EyeExposé, combines a full-screen two-dimensional thumbnail view of the open applications with gaze-based selection. EyeExposé has been implemented on Microsoft Windows using a Tobii 1750 eye gaze tracker for the gaze-based selection.

Figures 2 and 3 show how EyeExposé works. To switch to a different application, the user presses and holds down a hotkey. EyeExposé responds by showing a scaled down view of all the applications that are currently open on the desktop. The user simply looks at the desired target application and releases the hotkey.

The use of eye gaze instead of the mouse for pointing is a natural choice. Whether the user relies on eye gaze or the mouse, the visual search task to find the desired application in the tiled view is a required prerequisite step. By using eye gaze with an explicit action (the release of the hotkey) we can leverage the user's natural visual search to point to the desired selection. If we analyze the actions needed by the user to select a target window using the mouse, the total time would be:

$$T_{mouse} = t_{activation} + t_{visual\ search} + t_{acquire\ mouse} + t_{acquire\ cursor} + t_{move\ mouse} + t_{click\ mouse}$$

where $t_{activation}$ is the time for the user to press the hotkey or move the mouse to a corner of the screen to activate application switching; $t_{visual\ search}$ is the amount of time it takes the user to locate the target on the screen; $t_{acquire\ mouse}$ is the amount of time it takes the user to move the hands from the keyboard to the mouse; $t_{acquire\ cursor}$ is the amount of time to locate the cursor on the screen and $t_{move\ mouse}$ and $t_{click\ mouse}$ are the times to move and click the mouse button respectively.

We assume here that the visual search only needs to happen once since short term spatial memory enables the user to remember where the mouse needs to be moved. By contrast, the total time for selection using EyeExposé should be:

$$T_{eyeexposé} = t_{activation} + t_{visual\ search} + t_{release}$$

where $t_{release}$ is the time to release the hotkey. We expect $t_{release}$ to be considerably lower than ($t_{acquire\ mouse} + t_{acquire\ cursor} + t_{move\ mouse} + t_{click\ mouse}$). Gaze based application switching can therefore result in time savings by eliminating several of the cognitive and motor steps and replacing them with the single action of releasing the hotkey/trigger.

However, research has recognized that efficiency is not the only measure of the success of a particular interaction. The affect generated by that interaction and the subjective user experience is a key measure of the success and factor for adoption [18]. We hypothesized that users would like using EyeExposé since it provides a very simple and natural way of switching between applications. Therefore, we also chose to evaluate the user's subjective experience when using the gaze-based application switching.

EVALUATION

To evaluate EyeExposé, we conducted a user study with 20 subjects. Subjects were mostly graduate students and professionals and as such were experienced computer users who used various ways of switching applications (46% Alt-Tab, 38% Taskbar, 13% Exposé and 4% Virtual Desktop). Our subject pool had 13 males and 7 females with an average age of 28 years. 14 subjects did not require any vision correction. Four subjects wore contact lenses and 2 wore eyeglasses. None of the subjects were colorblind. Subjects had an average of 15 years of experience using the mouse.

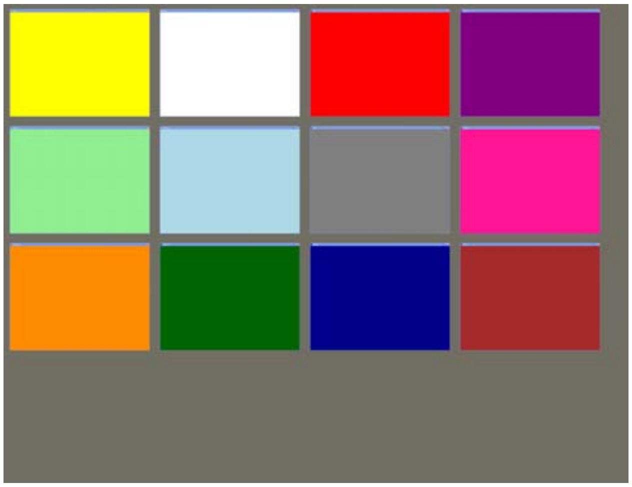
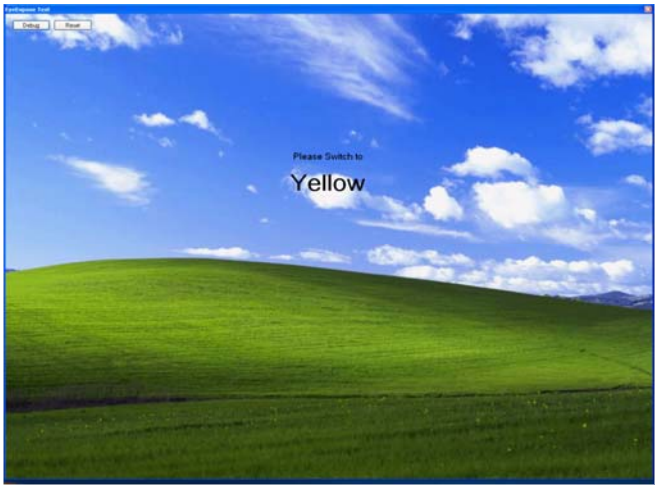


Figure 4. Exposé/EyeExposé view of 12 open windows, each window being a distinct color.



Please Switch to

Yellow

Figure 5. Instructions for which window to switch to next were shown on a second monitor.

Quantitative Evaluation

We tested speed and accuracy for 4 different application switching techniques - the Taskbar, Alt-Tab, an Exposé-clone with mouse based selection and EyeExposé in a 4x3 within-subjects experiment. For each application switching technique (Taskbar, Alt-Tab, Exposé and EyeExposé) we conducted trials with 4, 8 and 12 open windows to account for the number of windows being below, at and above average [13]. The order of the trials for each combination of technique and number of windows was varied to counterbalance and minimize learning effects.

Our original experiment design used real application windows such as Word, Excel and PowerPoint as the target windows. We believed that subjects would be easily able to recognize real application windows. However, our pilot studies revealed that subjects found it difficult to recognize applications in a testing environment. We therefore chose to use colored windows to reduce the cognitive load and the search time for subjects to identify the right target window.

Each window was a unique color and the name of the window matched the color of the window. Colors were carefully chosen to maximize recognition of the color by name (Figure 4). We verified that subjects were able to easily identify windows by the name of the color in our pilot studies. The window icon that appeared on the Taskbar and in the Alt-Tab view matched the color of the window. Maintaining the color consistency on window icons and names ensured that the Taskbar and Alt-Tab techniques also benefited from the use of colors. The final design used a unique icon and color for each window and was therefore biased in favor of the Taskbar and Alt-Tab since there were no applications with the same icon.

We used our implementation of an Exposé clone to perform the tests in a Windows environment and to instrument the code to capture timing data. Our implementation differs from the Mac OS X implementation in that we do not use a sophisticated layout algorithm. Our algorithm ordered windows heuristically based on the height, width or area of the window. We did not optimize window placement based on the spatial location of windows (not a variable in the study since all the applications were full-screen). The eye-based selection and mouse-based selection both used the same underlying code and layout algorithm and therefore the only difference in the setup was the selection technique used.

In the Exposé and EyeExposé conditions the placement of windows was randomized for half of the subjects, i.e. each time the user would activate the view the order of the windows would change. For the other half of the subjects the order of the windows would remain the same as in previous trials.

In the Taskbar condition, users had to click on the application button on the Taskbar and then click on a randomly placed "Next" button. This was done to force

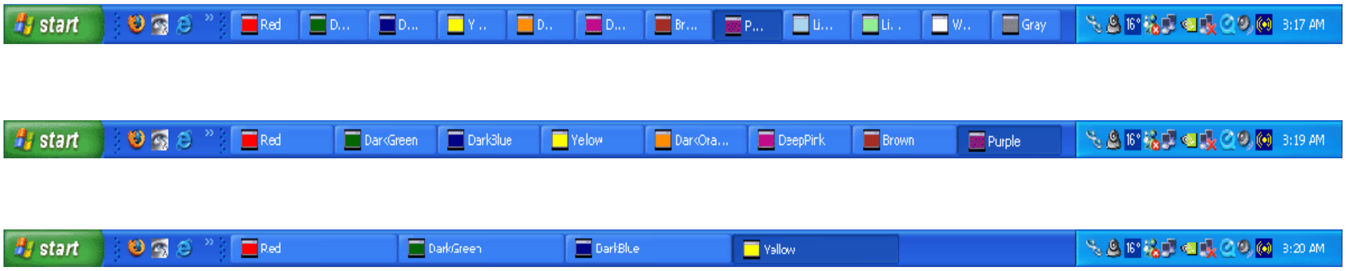


Figure 6. Taskbar in each of the 4, 18 and 12 window conditions.

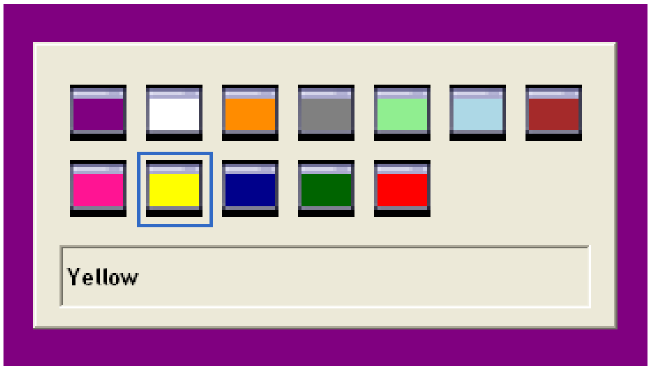


Figure 7. Alt-Tab view of 12 open application windows.

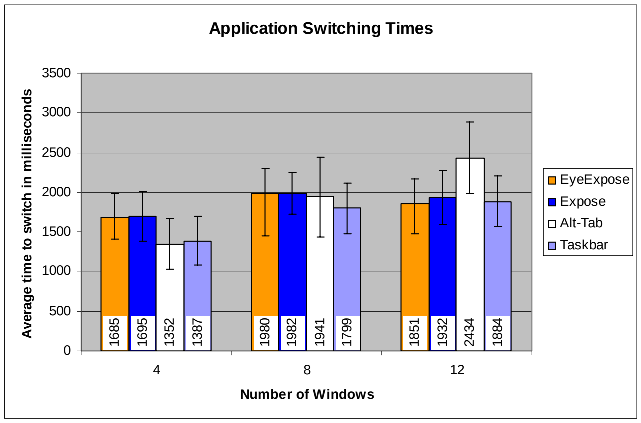


Figure 8. Quantitative evaluation results - performance.

users to move the mouse away from the Taskbar before the subsequent trial. For all other techniques users were prompted with the name of the next target window as soon as they completed the current trial. The number of windows on the Taskbar never exceeded a threshold that would cause it to add a second line with a scroll button (Figure 6).

The experiment used a Tobii 1750 (17" LCD) eye gaze tracker as the primary display. The screen resolution was set to 1280x1024 pixels. The test environment presented a window on a second monitor placed to the right of the primary screen, which displayed the instructions for the user (Figure 5).

We recorded the amount of time it took a user to select the target window, starting from the time the instruction appeared on the screen. If the user switched to an incorrect window, we recorded an error. In each of the 12 conditions (technique x number of windows), the user was asked to switch windows until they had completed 20 successful trials.

Qualitative Evaluation

We asked each of our subjects to complete a survey at the end of the study in which they ranked each of the four techniques on dimensions of speed, accuracy, ease of use, and user preference.

RESULTS

Figure 8 shows the performance results from the quantitative evaluation. A repeated measures ANOVA for number of windows and technique showed a significant effect for number of windows (F(2,38)=55.07, p < .01), for technique (F(1.9,36.9)=5.29, p < .01, Greenhouse-Geisser corrected) and interactions between number of windows and technique (F(6, 114) = 22.22, p < .01). Contrast analyses showed no significant difference between the Exposé and EyeExposé techniques. For the 4 window condition, as we expected, Alt-Tab was faster than Exposé and EyeExposé. For 8 windows, switching times for all four techniques were about the same, with the Taskbar showing a slight (but significant) advantage over Exposé. For the 12 window condition, EyeExposé had the lowest switching time (significant compared to Alt-Tab only).

Error! Reference source not found. shows the error rates from the study. It should be noted that the maximum error was less than 5%, or 1 error in over 20 trials. Several subjects performed the trials with no errors at all. The error rate distributions were highly non-normal. We therefore performed a Friedman's (non-parametric) ANOVA to compare participants' error rates. Results are shown in **Error! Reference source not found.**. The first row shows the ANOVA results. The second row shows the result of Bonferroni corrected pair-wise comparisons between the conditions. Only the listed condition pairs exhibited significant differences in error rates.

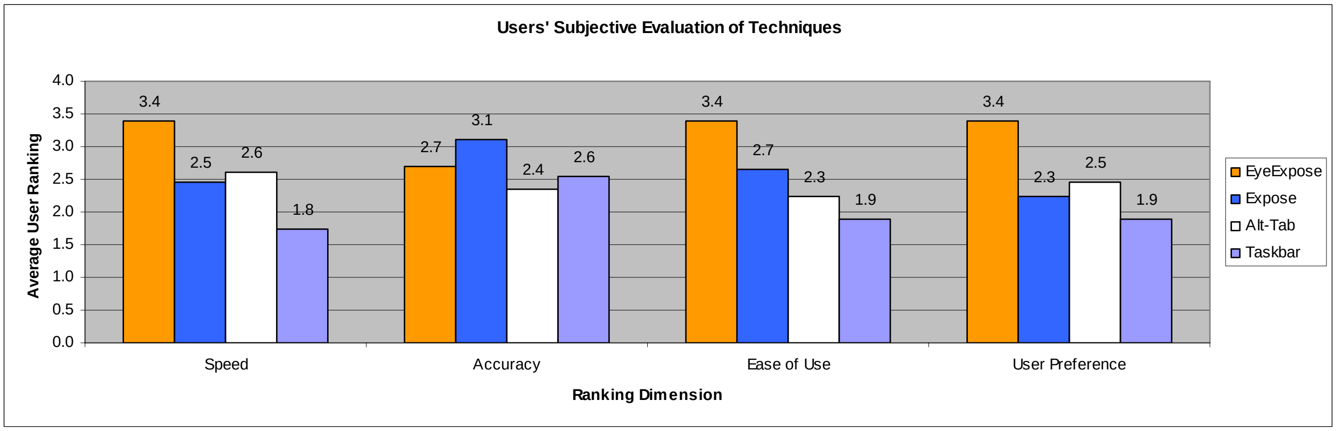


Figure 9. Qualitative evaluation results - survey ranking data.

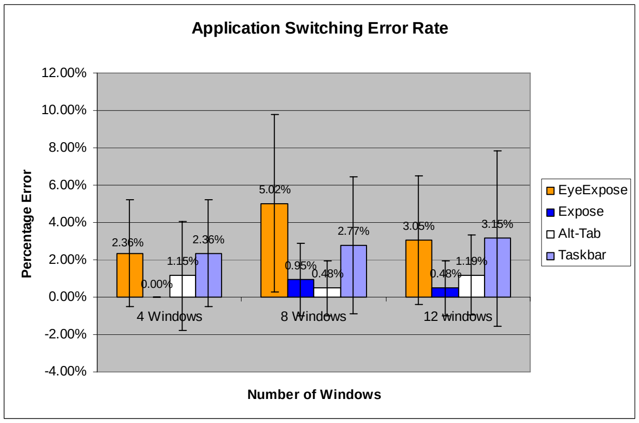


Figure 10. Quantitative study results - error rate.

As expected, the error rates for Exposé were the smallest since it provides large well recognizable targets, which are clicked on with a mouse.

Figure 9 shows a summary of the results from the qualitative evaluation where subjects ranked the four techniques for speed, accuracy, ease of use and user preference. EyeExposé was the subjects' choice for speed, ease of use, and the technique they would prefer to use most if they had all four approaches available. Again as expected, Exposé was the subjects' choice for accuracy.

DISCUSSION

The qualitative results from our user survey show that users have a strong preference for EyeExposé for switching between applications. Subjects felt that the approach was "natural," "faster" and "less annoying" when compared to other approaches.

Performance Results

We expected the performance of EyeExposé to show a clear advantage over using the mouse in the Exposé condition. However, quantitative results did not show a strong advantage in the time to switch and EyeExposé had a higher or comparable error rate than most other techniques.

We suspect that there are two reasons for this. First, the experimental design was such that users could keep one hand on the keyboard and the other hand on the mouse in the Exposé condition. Therefore, the cost of acquiring the mouse was zero. Furthermore, since the users already knew the location of the cursor on the screen from previous trials, the cost to acquire the cursor was also negligible. In real-world usage users may undertake other actions and may not remember the location of the cursor. The time to acquire the cursor in these cases would not be negligible.

One of the interesting observations during the study was that users could successfully parallelize some of the tasks required for pointing with the mouse. We noticed that users moved the mouse concurrently with visual search on the screen. This may be a result of the years of practice users have had with using the mouse as their primary pointing device. Therefore, the theoretical model for the time to switch we proposed earlier, which assumes a sequential ordering of the tasks is flawed due to the concurrent nature of some of the intermediate steps.

	Number of Windows		
	4	8	12
Friedman ANOVA	$X^2(3)=13.1$; $p<.01$	$X^2(3)=18.0$; $p<.01$	$X^2(3)=11.7$; $p<.01$
Significant Differences (pairs)	EyeExposé Exposé Exposé Taskbar	EyeExposé Alt-Tab EyeExposé Exposé	EyeExposé Exposé Exposé Taskbar

Table 1. Result of Friedman's ANOVA on errors.

Card et al. [8] measured the device switching time from the keyboard to the pointing device to be around 360ms. In real world use, users will incur this additional cost of acquiring the pointing device when using the Exposé technique. EyeExposé would then have a clear advantage over mouse-based selection.

Our implementation of the Exposé and EyeExposé technique took longer to show the visualization than the Alt-Tab condition or the Taskbar (always visible) due to the sluggishness of painting the screen in Windows. In the ideal scenario, the application switching technique would be integrated into the operating system and be optimized for drawing performance.

Accuracy Results

The error results exhibit high variance because most subjects were able to complete the task in a given condition with zero errors. The low number of errors suggests that the performance and the user preference may dominate as factors in the decision choice for which technique users choose to use.

A closer analysis of the errors in task switching suggests that Alt-Tab is prone to errors where the user overshoots or undershoots the target window. Only 4 subjects (20%) used the *Shift* key in order to cycle backwards when using Alt-Tab.

For the Taskbar errors usually stemmed from clicking on a neighboring window button or missing the Taskbar. This was especially true in the case of the 12 window condition where the size of the target decreased. This reaffirms the advantage of an Exposé like approach which provides large targets by using the transitional whole screen view as opposed to a permanently visible dedicated region of the screen. It should be noted that the number of windows was always low enough to show all the windows on the Taskbar without having to click the scroll button on the Taskbar.

Most errors in the Exposé and EyeExposé conditions occurred due to subjects picking the incorrect color (the brown color was initially confused with red by some users). The error counts therefore include both human error and error inherent in the switching technique. The Alt-Tab and Taskbar did not have as many errors since the name of the window (color) is readily visible in those techniques. In the Alt-Tab condition subjects would often notice that they had picked the incorrect target before releasing the Alt key and would therefore be able to correct the error immediately. Correcting errors in all other techniques requires subjects to repeat the trial. In the case of EyeExposé timing was another issue. We observed that users looked away at the side monitor in anticipation of instructions for the next target before they released the trigger key.

CONCLUSION

We found that using a combination of keyboard or other trigger to activate the Exposé-like visualization of open applications and then using eye gaze for selection was an effective technique for switching between applications quickly and naturally.

While the current cost of eye gaze tracking systems limits the widespread use of this approach, it is extremely likely that in the near future the availability of low-cost eye-tracking devices will increase and will be available for use on most desktops and laptop devices. Our studies showed that users strongly preferred EyeExposé as the application switching technique of choice on such devices.

ACKNOWLEDGMENTS

The authors would like to acknowledge Stanford Media X and the Stanford School of Engineering matching funds grant for providing the funding for the GUIDe (Gaze-enhanced User Interface Design) project.

REFERENCES

1. *Apple Mac OSX Aqua User interface feature: Exposé*, 2006. http://www.apple.com/macosx/features/expose/
2. *IPRIZE: a $1,000,000 Grand Challenge designed to spark advances in eye-tracking technology through competition*, 2006. http://hcvl.hci.iastate.edu/IPRIZE/
3. *Microsoft PowerToys for Windows XP: Alt-Tab Replacement - TaskSwitch*. http://www.microsoft.com/windowsxp/downloads/powertoys/xppowertoys.mspx
4. *Windows Vista; The Features*, 2006. http://www.microsoft.com/windowsvista/features/default.mspx
5. Agarawala, A. and R. Balakrishnan. Keepin' It Real: Pushing the Desktop Metaphor with Physics, Piles and the Pen. In Proceedings of *CHI*. Montréal, Québec, Canada: ACM Press. pp. 1283-92, 2006.
6. Amir, A., L. Zimet, A. Sangiovanni-Vincentelli, and S. Kao. An Embedded System for an Eye-Detection Sensor. *Computer Vision and Image Understanding, CVIU Special Issue on Eye Detection and Tracking* 98(1). pp. 104-23, 2005.
7. Ashmore, M., A. T. Duchowski, and G. Shoemaker. Efficient Eye Pointing with a FishEye Lens. In Proceedings of *Graphics Interface*. pp. 203-10, 2005.
8. Card, S. K., W. K. English, and B. J. Burr. Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys, for text selection on a CRT. *Ergonomics* 21(8). pp. 601-13, 1978.
9. Dragunov, A. N., T. G. Dietterich, K. Johnsrude, M. McLaughlin, L. Li, and J. L. Herlocker. TaskTracer: a desktop environment to support multi-tasking knowledge workers. In Proceedings of *IUI*. San Diego, California, USA: ACM Press. pp. 75-82, 2005.
10. Fono, D. and R. Vertegaal. EyeWindows: Evaluation of Eye-Controlled Zooming Windows for Focus Selection. In Proceedings of *CHI*. Portland, Oregon, USA: ACM Press. pp. 151-60, 2005.
11. Hansen, D. W., D. MacKay, and J. P. Hansen. Eye Tracking off the Shelf. In Proceedings of *ETRA: Eye Tracking Research & Applications Symposium*. San Antonio, Texas, USA: ACM Press. pp. 58, 2004.
12. Henderson, D. A., Jr. and S. K. Card. Rooms: The Use of Multiple Virtual Workspaces to Reduce Space Contention in a Window-Based Graphical User Interface. *ACM Transactions on Graphics* 5(3). pp. 211-43, 1986.
13. Hutchings, D. R., G. Smith, B. Meyers, M. Czerwinski, and G. Robertson. Display Space Usage and Window Management Operation Comparisons between Single Monitor and Multiple Monitor Users. In Proceedings of *AVI*. Gallipoli (LE), Italy: ACM Press. pp. 32-39, 2004.
14. Jacob, R. J. K. and K. S. Karn, Eye Tracking in Human-Computer Interaction and Usability Research: Ready to Deliver the Promises, in *The Mind's eye: Cognitive and Applied Aspects of Eye Movement Research*, J. Hyona, R. Radach, and H. Deubel, Editors. Elsevier Science: Amsterdam. pp. 573-605, 2003.
15. Jobs, S. P. and D. J. Lindsay. Computer interface having a single window mode of operation. USA Patent 2005, Apple Computer, Inc.
16. Kumar, M., *GUIDe: Gaze-enhanced User Interface Design*, 2006. Stanford. http://hci.stanford.edu/research/GUIDe
17. Kumar, M., Reducing the Cost of Eye Tracking Systems. 2006, Stanford University: Stanford.
18. Norman, D. A., *Emotional Design: Why we love (or hate) everyday things*. New York: Basic Books. 256 pp. 2004.
19. Robertson, G., *et al.* Scalable Fabric: Flexible Task Management. In Proceedings of *AVI*. Gallipoli (LE), Italy: ACM Press. pp. 85-89, 2004.
20. Robertson, G. G., *et al.* The Task Gallery: a 3D window manager. In Proceedings of *CHI*. The Hague, Amsterdam: ACM Press. pp. 494-501, 2000.
21. Smith, G., *et al.* Groupbar: The Taskbar Evolved. In Proceedings of *OZCHI*: ACM Press. pp. 1-10, 2003.
22. Zhai, S. What's in the Eyes for Attentive Input, *Communications of the ACM*, vol. 46(3): pp. 34-39, March, 2003.