Effects of Display Blurring on the Behavior of Novices and Experts during Program Debugging

Roman Bednarik
Department of Computer Science
University of Joensuu
P.O. Box 111, FIN-80101, Joensuu, Finland
bednarik@cs.joensuu.fi

Markku Tukiainen
Department of Computer Science
University of Joensuu
P.O. Box 111, FIN-80101, Joensuu, Finland
mtuki@cs.joensuu.fi

ABSTRACT
The Restricted Focus Viewer (RFV) relates a small part of an otherwise blurred display to the focus of visual attention. A user controls which part of the screen is in focus by using a computer mouse. The RFV tool records these movements. Recently, some studies used the RFV to investigate the cognitive behavior of users and some others have even enhanced the tool for research of usability issues.

We report on an eye-tracking study where the effects of RFV's display blurring on the visual attention allocation of 18 novice and expert programmers were investigated. We replicated a previous RFV-based study and analyzed attention switching and fixation durations reported by an eye tracker. Our results indicate that the blurring interferes with the strategies possessed by experts and has an effect on fixation duration: however, we found that debugging performance was preserved. We discuss possible reasons and implications.

Author Keywords
Eye-movement tracking, visual attention, psychology of programming, experimental tools

ACM Classification Keywords
H5.1. Information interfaces and presentation (e.g., HCI): Evaluation/methodology

INTRODUCTION
In fields of HCI research such as usability, the psychology of programming, or diagram-understanding, it is important to investigate the limitations and impacts of tools used for collecting user behavior as indicators of the cognitive processes of users. We are highly interested in whether the actual use of tools does interfere with the (otherwise unaltered) behavior of participants in an experiment.

Computer programming and research into the related cognitive processes is typically a domain where researchers benefit from applying tools to investigate visual attention allocation. A great amount of research in the past has consistently confirmed the relations between shifts of visual attention focus and movements of the eyes; see [4] for a review.

Related Work
The Restricted Focus Viewer (RFV) [6, 2] is a visual attention tracking system which displays visual stimuli in a blurred form and allows only a small region of the screen to be seen in focus. To get a portion of the stimuli in focus, users have to move the computer mouse over the area that they want to come into focus ; the focus of visual attention is then thought to be linked to the position of the spot. The RFV tracks the movements of a computer mouse over the stimuli and records them together with the indices of areas of interest (AOI, e.g. the predefined areas of display stimuli) and timestamps.

In the context of Java program debugging, a modified version of RFV was employed in various studies [7, 8, 9]. In these studies, a Software Development Environment (SDE) was built on top of the RFV and used to track visual attention and to investigate the coordination of multiple representations of programmers debugging Java programs. Figure 1 shows a screenshot of the SDE when the restricted condition is on. The SDE's interface contains three AOIs: the code is on the left, the visualization on the top-right, and the output is on the bottom-right. The focused region, as set by user's mouse, is located over the code in left panel.

The RFV was also employed in the research of shifts of visual attention during integration of text and graphics [3]. Other researchers used the RFV idea for usability studies of hyperlinked documents [10].

In our previous report [1], we showed that RFV's blurring condition introduces interference to the cognitive strategies of programmers: representation (attention) switching was inhibited and the mean fixation duration decreased. An interesting and important question could be raised: "What is the effect of this intervention on different classes of experimental participants, when the experience is considered as a study criterion?" Our hypothesis is that with *greater experience the actual usage of a tool does not interfere with strategies possessed*.

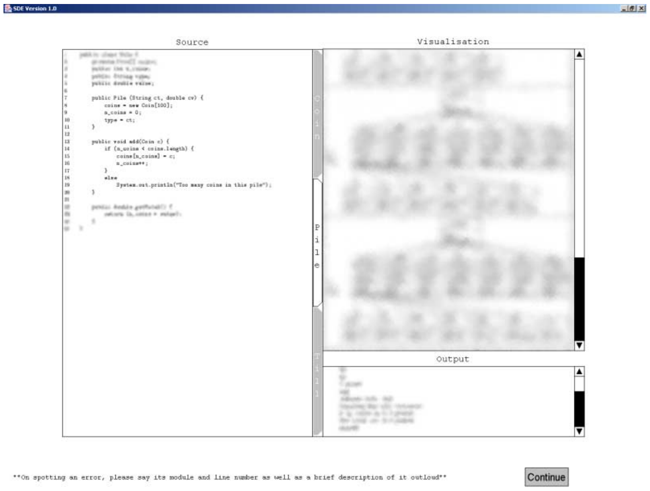

Figure 1. A screenshot of SDE used in experiment. Focused spot is located in the code on left.

METHOD

We analyzed and compared the locations of the focus of visual attention between two conditions and two levels of experience. The SDE interface, based on the RFV-tool, was used to present stimuli blurred (RFV-on) and to present stimuli unblurred (RFV-off). Two groups of programmers (a novice and an expert group) worked with the environment to debug three Java programs. The visual attention was recorded using an eye-tracker.

Design and Participants

A mixed one within-subject (*RFV restricting condition*), one between-subject (*a level of experience*) design was used with four dependent variables (*number of errors spotted*, *accumulated fixation time, mean fixation duration*, and *switching frequency as measured by eye tracker*). The accumulated fixation time is the total time a participant spent during a session fixating an AOI. For an AOI, all fixations were summed and the number was divided by the total fixation count throughout the experiment, giving the mean fixation duration. The switching frequency refers to the average number of switches per minute between each of the AOIs. Most of the results were analyzed by performing ANOVA and/or planned paired t-tests.

In the study a total of 18 participants were recruited from a population of students, researchers, and teachers from the authors' department. All subjects had normal or corrected-to-normal vision according, by their own report, and had never taken part in an eye-tracking experiment. The average age was 25.3 (SD=4.4) years. Three participants were females. The programming and Java experience varied from having just passed a Java course and having little experience to professionals working in programming-related careers. The less-experienced group consisted of 10 programmers, who had an average of 63 months of programming experience, 8.13 months of which were Java programming. No novice participant had ever worked as a professional programmer. The expert group was formed from the remaining 8 participants, whose programming experience was 96 months, whose Java experience was 16.25, and who all, except one, had professional experience with programming.

Procedure

Before the experiment, participants had to pass an automatic eye-tracking calibration procedure. After that, the participants read detailed instructions about the experiment and the environment used. Three programs were debugged. The first warm-up session was performed under the RFV restricted view condition (RFV-on) so that the participants could become familiar with controlling the focused spot and operating the debugging environment. Then, the two main debugging sessions were performed; one session was performed under the RFV-on condition, the other session was performed under the RFV-off condition where the whole display was presented in focus. The order of the programs and conditions was counterbalanced.

Each session had two phases. First, the specification of the program was displayed. It described the problem the program was supposed to solve and the approach to the solution. Two sample interactions were provided - the desired behavior and actual behavior of the program. Second, the participants were given ten minutes to debug the program and were instructed to find as many errors as possible and to report them aloud.

Materials and Setup

The target programs were identical to those used in [7]. The object of the warm-up program was to determine whether a point was inside a rectangle. The first program printed out the names of the children of a sample family and the second program counted the cash in a register till which gave subtotals for different denominations. In a previous study done by Romero et al. [7], two versions of the target programs and several visualizations were used. In our replication of the experiment, we used Romero's less sophisticated versions of the programs and graphical functional representations. The two main target programs were seeded with four errors each; the warm-up program contained two errors. The programs contained no syntactical error and participants were notified of this.

For eye-tracking, the remote Tobii ET-1750 (sampling at 30Hz) eye tracker was used. The eye tracking data were collected throughout the whole experiment; the RFV collected data in the RFV-on condition, but for the purposes of this study this data was not used. The AOIs were defined to correspond with the three main panels in the SDE window: the code, visualization, and output panel.

The Software Debugging Environment (SDE) used in the previous studies [7, 8, 9] was employed for the experiment as a source of stimuli. In these studies and in the present experiment, the program code, the visualization, and output were pre-computed and static.

RESULTS

Debugging performance

The debugging performance was measured by the number of errors spotted. Under the RFV-on condition, the less experienced group found 2.1 (SD=1.10) errors on average and the more experienced group spotted 3.125 (SD=0.84) errors on average: $t(7) = 2.53$, $p<.05$. Under the RFV-off condition, the less experienced group found 2.1 (SD=0.88) errors on average, and the more experienced group spotted 2.88 (SD=1.13) errors on average. The effect of the restricted view condition on debugging performance was not significant.

Gaze related behavior

Figure 2 presents the mean fixation durations for each of the three main areas of interest and the overall mean fixation duration. A two way ANOVA revealed an effect of RFV condition on mean fixation duration ($F(1,16) = 4.45$, $p<.051$) and no interaction between level of experience and RFV condition ($F(1,16) = 0.26$, *ns*). The planned paired t-tests revealed that, for experts, the overall mean fixation duration and the mean fixation durations over the code AOI significantly differed between RFV-on and RFV-off conditions ($t(7) = 2.80$, $t(7) = 2.66$, respectively, all $p<.05$). The overall mean fixation durations of the experts were 308.82 ms (SD=83.95) and 263.09 ms (SD= 70.60) under RFV-on and RFV-off, respectively. For the code panel, the mean fixation durations of the expert group were 312.44 ms (SD=85.69) and 268.23 ms (SD= 73.64) under RFV-of and RFV-off, respectively. Considering novices, there was no significant difference in fixation durations between RFV-on and RFV-off conditions according to pair-wise tests. However, the mean fixation duration between the areas was significantly different, $F(2,16) = 10.13$, $p<.005$).

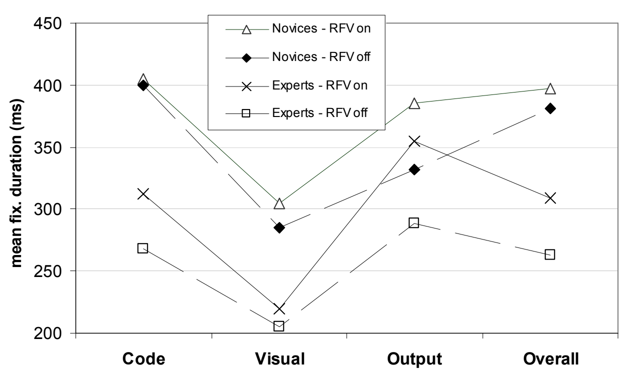

Figure 2. Mean fixation durations over the main panels of interface and overall mean fixation duration.

The distribution of relative accumulated fixation time over the areas of interest was not affected by the RFV condition for either of two experimental groups. Novice participants spent on average 82% of whole time fixating on the code panel, 14% over visualization, and 4% of total time over the output area of interest. For experts the relative accumulated fixation time followed distribution 87%, 10%, and 3%.

The dynamics of attention switching behavior was measured by the average number of switches per minute between any two of all areas of interest (Figure 3). The effect of RFV condition was significant, $F(1,16) = 7.82$, $p<.05$), and the interaction between the level of experience and RFV condition was significant at an alpha of 0.92, $F(1,16) = 3.59$, $p<.08$). We observed a decrease in the number of switches per minute, which was significant for experts ($t(7) = 2.53$, $p<.05$). More over, the average number of switches per minute of novices was significantly correlated under RFV-on and RFV-off ($r(10) = 0.642$, $p=.046$), while the same correlation for experts was low and not significant ($r(8) = 0.068$, $p=.873$).

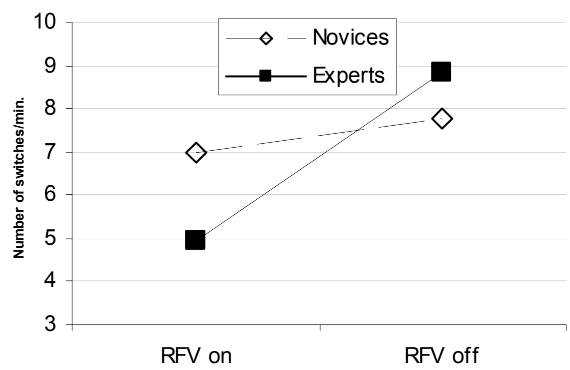

Figure 3. Number of switches per minute.

DISCUSSION AND CONLUSIONS

The purpose of this experiment was to investigate the effects of RFV's display blurring on the behavior of experimental participants. The RFV displays a focused spot within otherwise blurred stimuli images; the spot is controlled using a computer mouse. An eye tracker reports the coordinates of point of gaze which is thought to be connected to the focus of visual attention. We used an eye tracker to measure (1) the visual attention location of participants, (2) the accumulated fixation time over areas of interest, and (3) the fixation duration while debugging using the RFV based environment. We replicated one of the previous studies which used the RFV as a tool to measure visual attention switching.

The accumulated fixation time distributions of either novices or experts were not affected by the RFV's restricted view. This indicates that an RFV-based tool does not interfere with this measure; moreover, it means that participants spend the same amount of time fixating the areas of interest.

The mean fixation durations of all participants were increased under the RFV-on condition; for experts the effect led to a significant increase under the RFV-on condition. In eye movement based studies, the mean fixation duration is a measure of processing, which is related to the depth of required processing [5]. The RFV's blurred display caused our study's experts to process visual information longer that it was under the unrestricted view.

The RFV makes switching a manual task rather than a perceptual one. When a display is blurred the natural switching frequency decreases. In our study, the effect of the blurring was more significant for experts than for novices. As also seen from the correlations, novices' strategies seemed to be almost unaffected, while experts' behavior was different when the display was blurred.

These results indicate that experts are most probably processing much information through peripheral vision during debugging and the blurring is creating an obstacle causing the processing to take longer. This hypothesis is supported by a look into the video protocols. It is common that a participant places the focused point over the investigated piece of interface, while visually attending to some other, blurred part of the interface. The focused spot is therefore functioning as a kind of bookmark, but not as a single spot through which the information is exclusively extracted.

Despite the fact that RFV does not interfere with debugging performance, we conclude that it creates several effects on the behavior of experimental participants. The effects, in our study, were more serious for a group consisting of participants with higher experience levels. Since some researchers continue the idea of measuring visual attention allocation by display blurring, our results provide a warning: some conclusions based on a behavior measured using the RFV might be inaccurate.

ACKNOWLEDGMENTS

The work of first author was supported by a grant of Faculty of Science, University of Joensuu, Finland.

We would like to thank Pablo Romero for sharing the materials used in this study.

REFERENCES

1. Bednarik, R., Tukiainen, M. Visual attention tracking during program debugging. In *Proceedings of NordiCHI 2004, The Third Nordic Conference on Human-Computer Interaction*, October 23-27, 2004, Tampere, Finland, ACM Press, pp. 331-334.
2. Blackwell, A. F., Jansen, A.R., Marriott, K. Restricted Focus Viewer: A tool for tracking visual attention. In M. Anderson, P. Cheng & V. Haarslev (Eds.), *Theory and Applications of Diagrams*. Lecture Notes in Artificial Intelligence 1889, pp. 162-177, Springer Verlag, 2000.
3. Futrelle, R. P. & Rumshisky, A. Discourse Structure of Text-Graphics Documents. *1st International Symposium on Smart Graphics* Hawthorne, NY. ACM Press, 2001.
4. Godijn, R. & Theeuwes, J. The relationship between exogenous and endogenous saccades and attention. In Jukka Hyönä, Ralph Radach & Heiner Deubel (Eds). *The Mind's Eyes: Cognitive and Applied Aspects of Eye Movements*, pp. 3-26, 2003.
5. Goldberg, J. H. and Kotval, X. P. Eye Movement-Based Evaluation of the Computer Interface. In Kumar, S. K. (Eds.), *Advances in Occupational Ergonomics and Safety*, Amsterdam: IOS Press, pp. 529-532, 1998.
6. Jansen, A.R., Blackwell, A.F. and Marriott, K. A tool for tracking visual attention: The Restricted Focus Viewer. *Behavior Research Methods, Instruments, and Computers,* 35(1), 57-69, 2003.
7. Romero, P., Cox, R., du Boulay, B., Lutz, R. Visual attention and representation switching during Java program debugging: A study using the Restricted Focus Viewer. *Diagrammatic Representation and Inference: Second International Conference, Diagrams 2002* Callaway Gardens, USA. Lecture Notes in Artificial Intelligence, 2317, pp. 221-235, Springer Verlag, 2002.
8. Romero, P., Lutz, R., Cox, R., du Boulay, B. Co-ordination of multiple external representations during Java program debugging. *Empirical Studies of Programmers symposium of the IEEE Human Centric Computing Languages and Environments Symposia*, Arlington, VA, 2002.
9. Romero, P., du Boulay, B., Cox, R., Lutz, R. Java debugging strategies in multi-representational environments. *15th Annual Workshop of the Psychology of Programming Interest Group (PPIG'03)*, pp. 421-434, 2003.
10. Tarasewich, P., Fillion, S. Discount eye tracking: The Enhanced Restricted Focus Viewer. In *Proceedings of AMCIS*, August 2004, New York, pp. 1-9.