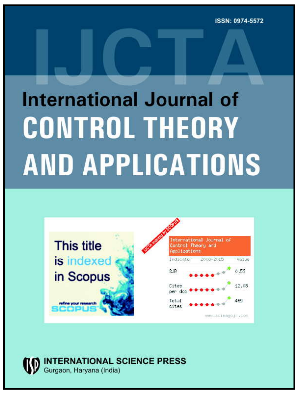


International Journal of Control Theory and Applications

ISSN : 0974-5572

Volume 10 • Number 19 • 2017

An Eye Tracking Study to Understand the Visual Perception Behavior while Source Code Comprehension

Chandrika K.R.[1] and Amudha J.[1]

[1] *Dept. of Computer Science & Engineering, Amrita School of Engineering, Bengaluru, Amrita Vishwa Vidyapeetham, Amrita University, India, Emails: kr_chandrika@blr.amrita.edu, j_amudha@blr.amrita.edu*

Abstract: An eye-tracking study to understand the visual attention of participants with varied learning experience in programming while they find bugs and comprehend source code is the focus of the paper. The participants were grouped as Novice, Learners and Experts based on their learning experience in programming. The statistical analysis carried on the eye gaze data indicate significant differences in visual attention measured by eye movement data between these groups. The visual attention maps indicates that experts scanpath were more focused towards the error region in the program segment whereas the novices and learners focused on the whole stimuli indicating search for errors. This also indicate that the experts perceive the programs better and hence able to fixate on the error regions. Statistical analysis also revealed that source code comprehension is influenced only by their learning experience and not by their age or gender of participants.

Keywords: Eye Tracking, Software Engineering, Source Code Comprehension, Debugging

1. INTRODUCTION

Visual Attention is a process by which some objects in a stimuli are selected to receive more processing than others. Eye movements, visual attention and memory allows to perceive a scene. Eye tracking technology determines the visual attention of a user on a visual scene. Tracking of eye movements provides a direct measure of where the attention is deployed [1]. Eye movements like fixations and saccades are two eye movements that determines the visual attention of a user. Fixation is where eye momentarily stops and saccades is a rapid eye movement from one fixation to another. Fixations helps to focus on an object and saccades to move the focus on to the object. The Eye tracking technology have been used in the area of physiology, usability testing, marketing, industrial design and human-computer interaction. Recently, researchers from the software engineering community started showing interest in eye tracking methodologies to study the reading strategies [2] and their effect on the program comprehension [4]. The task of comprehending source code improves the quality of the software under development and helps in reducing the bugs. Eye tracking technology can providesthe visual attention of the software developers that can be helpful in their daily tasks [3]. The availability and affordability of eye trackers can help software engineering researchers to perform different studies to understand the behavior of software developers.

In 1990, Crosby and Stelovsky reported the very first eye tracking study. They examined the source code reading and its effect on the program comprehension [4][7]. In 2006, Umano et al analyzed the individual performance of source code review using eye tracking [4]. They identified a scan pattern and observed that software reviewers who did not spend more time in scanning the source code incline to take more time in detecting bugs [5]. In 2006, Aschwanden and Crosby, investigated visual attention of expert and novice. Their study reveals that experts and novices have significant differences in visual attention on different parts of the program [6]. In 2012, Sharif and Maletic [7] replicated the study of Uwano with large number of participants and varied programming expertise. They investigated if the scan pattern holds in their own study and the impact of subject's experience on effectiveness and efficiency of the code reviewing process [4]. The results indicate significant correlation of scanning with defect detection time. In 2014, Turner et al, compared expert's and novice's scanpath to investigate the influence of programming languages C++ and Python on code comprehension The results indicates no significant difference in the behavior of experts and novices in terms of accuracy and time but a significant difference in the rate at which students looked at defected lines[8].

The focus of the study is to understand the visual attention of participants with varied learning experience in programming while they find bugs and comprehend source code. The study is a motivation from the research work conducted by Sharif et al. [7]. The research work reported was conducted on large number of participant with varied programming expertise. Our research work is also conducted on large participants with varied learning experience of programming. Our aim is to prove that source code comprehension is only influenced by the learning experience of the participants but not their age or gender. The task is to read and comprehend a source code with the goal of finding errors.

The paper is organized as follows: Section II provides the focus of study, Section III on materials and methods followed by results and analysis in Section IV and conclusion and future work in Section V.

2. FOCUS OF STUDY

The aim of the eye-tracking study is to analyze and interpret the eye gaze behavior of participants while performing code comprehension to detect the bugs. The study focuses on visual attention measured as eye movement data, of participants who are learning programming concepts. Based on their learning experience, the participants are grouped as Novice, Learners and Experts. The research questions addressed in this paper are:

RQ1: Is there a difference in Visual Attention of participants with varied learning experiences?

RQ2: Does Age and Gender influence the Visual Attention of participants?

The research question RQ1 is to study the visual attention, measured as eye movement data, of participants with varied learning experience in programming concepts and languages. The study focuses on understanding the visual attention of participants who are beginners with those learning and experts in the language. The Novice participants had few months of learning experience, Learners had an experience of six months to two years and Experts had an experience of more than two years. The second research question RQ2 is to confirm our study that Visual Attention of participants are only influenced by learning experience but not age and gender of the participants.

3. EXPERIMENT DESIGN

The experiment is to analyze the gaze behavior of individuals while comprehending and debugging programs written in C language. Eye movements were captured using Eye Tribe Eye Tracker which has a sampling frequency of 60Hz [9]. The experiment was set up and the eye gaze data was collected and analyzed using the open source OGAMA [Open Gaze And Mouse Analyzer][10]. The statistical analysis of the eye movement data were realized in R language. The source code was presented on a 24" screen with resolution 1366 x 768. The participants were seated about 60 to 65 cm from the stimuli. The participant's audio and video were recorded for estimating the defect detection time. A Times New Roman font of size 14 pt with text color black and background color white was used for presenting the source code. There is no time limit for performing the task.

3.1. Task

The task was to detect the bugs in the source codes written in C programming language and think aloud the line number of the same. Table 1 represents the stimulus description and the number lines of code. The first program had syntax errors only and the second program had syntax errors and logical errors.

Table 1
Stimulus Description

Sl No.	*Stimulus*	*Lines of Code*	*No of Errors*	*Error statements*
1	Swapping of two numbers	18	2	Input and output statements
2	Implementation of Stack data structure using functions	43	3	Function call, Decision making statement and Function return

3.2. Participants

The participants were volunteered and recruited from novice to advanced learners. The participants were students and faculty members in the Department of Computer Science and Engineering aged between 18 to 50 years. No compensation was provided for the participants. A total of 50 participants, participated in the eye tracking study. All the recruited participants were learners or experts in C programming language. Out of these 50 participants, 2 expert participants' trials were not included for analysis as their response was too quick in debugging task that the analysis software failed to estimate fixations over the stimuli. Another 6 participant's eye tracking data were discarded due to poor quality. This was due to the technical issues with the device and recording software. All the participants had normal vision or corrected vision. From the 42 participants selected for data analysis, 23 are females and 19 males. There were 4 experts, 28 learners and 10 novices. Fifteen participants were within 20 years of age, seventeen participants between 21years and 30 years of age and ten participants above 31 years of age.

3.3. Metrics and Hypothesis

The metrics used are Accuracy, Total Fixation Count (FC) (number of fixations), Total Fixation Duration(FD), Fixation Count in Area of Interest (AOI)(FCAOI) and Total Fixation Duration in AOI(FDAOI). AOI is the defect lines in the code. Accuracy is measured in percentage based on the number of defects correctly recognized by the participant. Visual Attention is measured by Total Fixation Count, Total Fixation Duration, Fixation Count in AOIs and Fixation Duration in AOIs. Figure 1 represents a visualization of the stimuli with various eye movement metrics. The diagram shows a scanpath(green colour) of a participant with various AOIs (blue box). The green dots represent the fixations, and the green line represents saccade. The number(red colour) inside the AOIs represent the Fixation Count in AOIs. The figure represents how a participant has scanned the code given for code comprehension. Fixations represents that the participant's focus on the lines of code and saccade represents the movement of eye from one line to another. The lines 6, 8, 10 and 11 are error lines and are the AOIs.

The individual hypotheses formulated from the research question discussed earlier are illustrated below:

$H1_0$: Bug Detection Accuracy is same for Novice, Learners and Experts.

$H2_0$: Visual Attention of Novice, Learners and Experts are same.

$H3_0$: Visual Attention in Men and Women are same.

$H4_0$: Visual Attention of participants in varied age groups are same.

The hypothesis $H1_0$ tests the accuracy of bug detection between various groups of learners. Hypothesis $H2_0$ tests the visual attention of participants in varied groups based on the eye tracking metrics. The hypothesis $H1_0$ and $H2_0$ were derived based on the research question RQ1. Visual Attention of men and women and participants in varied age groups were tested in hypothesis $H3_0$ and $H4_0$ and derived from research question RQ2.

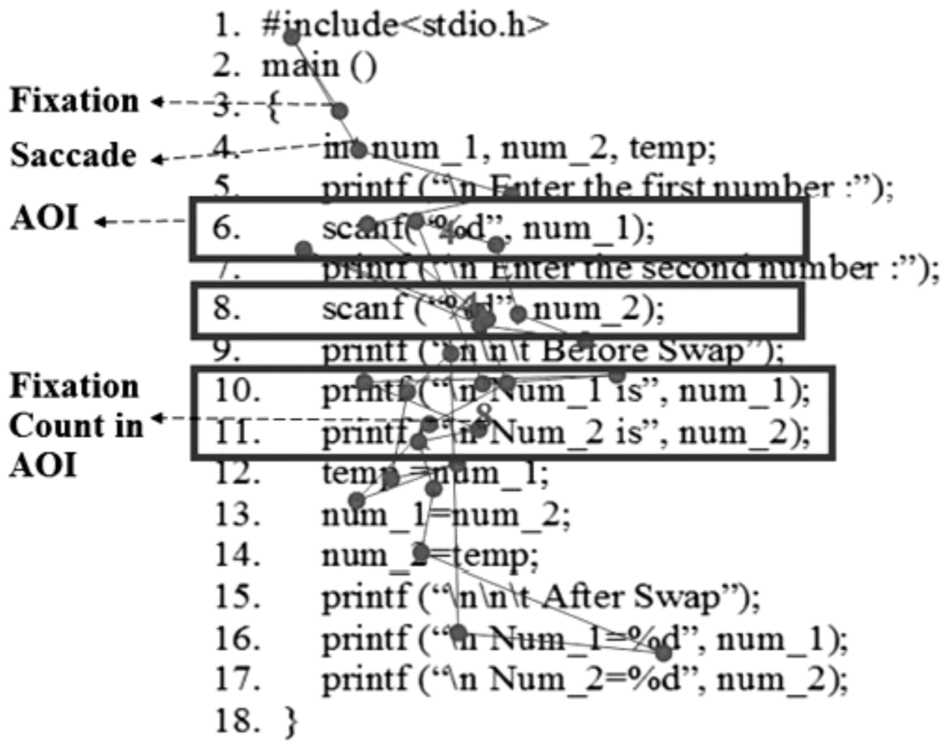


Figure 1: Visualization of the Stimulus with Eye movement metrics

4. EXPERIMENT RESULTS AND ANALYSIS

The eye movements of the participants were recorded and visualized using OGAMA. The groups where formed based on the learning experience in programming of the participants. The visualization images of eye tracking data collected from the eye tracker were individually inspected for the quality. The eye tracking data were then normalized for best results.

Figure 2shows the Attention map Visualization of Novices, Learners and Experts respectively. The bugs were induced in line numbers 15,17,19,24, 35 and 38. The attention map visualization of experts' shows that they focused more on the line numbers where errors were induced, whereas novices and learners scanned the complete code indicating more searching for detecting bugs. This also indicate more visual effort to perform the task. The novices and learners focused more on main function and the declaration statements and method calls rather than on the logical statements. Even though there was some focus on the buggy lines in the main function most of the learners failed to identify the same. The experts scanned the whole code and spend more time on the buggy lines than on other statements. The visualizations shows clear distinction between varied groups.

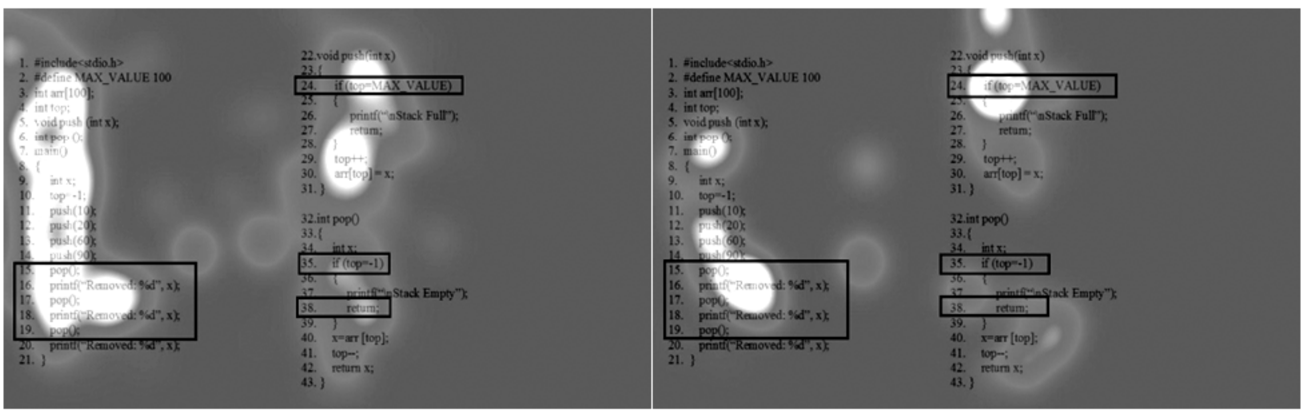
A statistical analysis based on Welch Two Sample t-test using R language was conducted to test the hypothesis. Table 2 represents the p value of t-test on bug detection accuracy to test hypothesis $H1_0$. The results shows that the bug detection in novices, learners and experts have significant differences. We infer that experts are able to find bugs than other groups and hence reject the hypothesis $H1_0$.

Table 2
t-test results of Bug Detection Accuracy of varied groups of participants

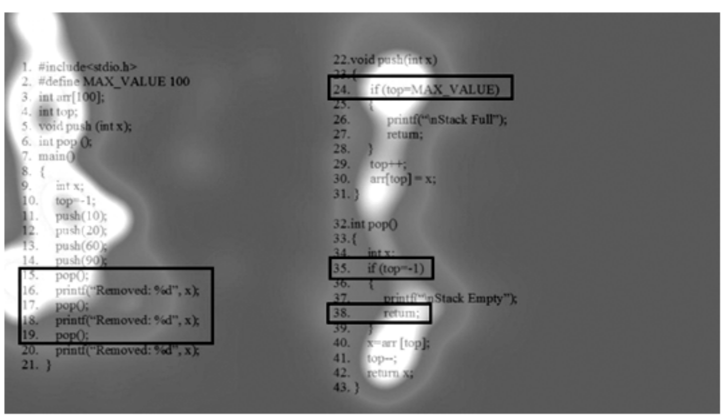
	Novice				*Learner*				*Expert*			
	FC	*FD*	*FCAOI*	*FDAOI*	*FC*	*FD*	*FCAOI*	*FDAOI*	*FC*	*FD*	*FCAOI*	*FDAOI*
Novice	-	-	-	-	0.4851	0.7608	0.6742	0.783	0.04304	0.009487	0.03355	0.8834
Learner	0.4851	0.7608	0.6742	0.783	-	-	-	-	0.0002	0.01085	0.002884	0.7143
Expert	0.04304	0.009487	0.03355	0.8834	0.0002	0.01085	0.002884	0.7143	-	-	-	-

Table 3
t-test results of Visual Attention between varied groups

Groups	*Novice*	*Learner*	*Expert*
Novice	-	4.888E-12	1.993E-04
Learner	4.888E-12	-	2.273E-05
Expert	1.993E-04	2.273E-05	-



(a) **(b)**



(c)

Figure 2: Attention map Visualization of (a) Novices (b) Expert (c) Learners

The p value of the t-test to test hypothesis $H2_0$, the differences in visual attention between various groups, is represented in Table 3. The results indicate significant differences in number of fixations [p-value=0.04304], fixation duration [p-value=0.009487] and fixation count in AOI [p-value=0.03355] between Novices and Experts.

We also found significant differences in number of fixations [p-value=0.0002], fixation duration [p-value=0.01085] and fixation count in AOI [p-value=0.002884] between Learners and Experts. There was no significant differences found between Novices and Learners. Collectively results shows that experts, learners and novices behave differently while performing source code comprehension. We infer that experts are able to fixate on the buggy lines whereas novices and learners are found less effective to fixate gaze on it. This indicate that experts perceive the programs better than other groups and so ale to fixate on errors. Thus we rejectthe hypothesis $H2_0$.

A gender based analysis on eye movement data was performed to understand the differences in visual attention of men and women while debugging the source code. Our results represented in Table 4 indicate no significant differences in fixation count, fixation duration, fixation count in AOI and fixation duration in AOI between men and women. We infer that source code comprehension is not influenced by gender of a participant and hence we reject the hypothesis $H3_0$.

A statistical test was also conducted to analyze the visual attention of participants in varied age groups namely within 20 years, between 20 to 30 years and above 30 years. The t-test results are presented in Table 5. The results indicate no significant differences in fixation count, fixation duration, fixation count in AOI and fixation duration in AOI of participants in varied age groups. We infer that the source code comprehension is not influenced by the age of participants and hence reject the hypothesis $H4_0$.

Table 4
t-test results based on Gender

	Women				
	Accuracy	*FC*	*FD*	*FCAOI*	*FDAOI*
Men	0.7702	0.1044	0.6247	0.8152	0.2491

Table 5
t-test results based on Age groups

	<=20years				*>20 and <=30 years*				*>30 years*			
	FC	*FD*	*FCAOI*	*FDAOI*	*FC*	*FD*	*FCAOI*	*FDAOI*	*FC*	*FD*	*FCAOI*	*FDAOI*
<=20years	-	-	-		0.88	0.69	0.46	0.39	0.03	0.30	0.68	0.65
>20 and <=30 years	0.88	0.69	0.46	0.39	-	-	-	-	0.0	0.1	0.3	0.3
>30 years	0.03	0.30	0.68	0.65	0.00	0.11	0.27	0.26	-	-	-	-

5. CONCLUSION

The research work aimed at investigating visual attention of participants with varied learning experience in programming. The participants were grouped as Novice, Learners and Experts based on their learning experience. The task was to comprehend and detect bugs in source code. A statistical analysis based on t-test revealed significant difference in bug detection accuracy and visual attention between the groups. A significant difference in total number of fixation (FC), total fixation duration (FD) and number of fixation in AOI (FCAOI) were found between Experts and Learners and Experts and Novices whereas no significant difference in the eye movement data between Novices and Learners were found. The visual attention maps indicates that experts scanpath were more focused towards the error region in the program segment whereas the novices and learners focused on the whole stimuli indicating search for errors. The results collectively conclude that the experts were able to find bugs and fixate on the buggy lines than the novices and learners. The participants were further grouped based on gender and age to analyze their influences on visual attention while source code comprehension and debugging.The results indicate no significant differences in eye movement data between men and women and people with varied age groups. Our experimental study conclude that experts visual perception behavior is different from novices and learners. Experts perceive programs better and hence fixate on the error lines. We also conclude that bug detection and source code comprehension are influenced only by the learning experience not by gender or age of participants. In future work, we extend the work for studying the eye patterns of software developers in debugging large complex programs containing nested loops, algorithms and complex mathematical functions.

REFERENCES

[1] Duc, Albert Hoang, Paul Bays, and Masud Husain. "Eye movements as a probe of attention."*Progress in Brain Research,*2008

[2] Sharafi, Zohreh, Zéphyrin Soh, and Yann-Gaël Guéhéneuc. "A systematic literature review on the usage of eye-tracking in software engineering." *Information and Software Technology,* 2015

[3] Sharif, Bonita, and Timothy Shaffer. "The Use of Eye Tracking in Software Development." *International Conference on Augmented Cognition.* Springer International Publishing, 2015.

[4] Crosby, Martha E., and Jan Stelovsky. "How do we read algorithms? A case study." *Computer* 23.1, 1990

[5] Uwano, Hidetake, et al. "Analyzing individual performance of source code review using reviewers' eye movement." *Proceedings of the 2006 symposium on Eye tracking research & applications*. ACM, 2006.

[6] Aschwanden, C., and Crosby, M. "Code scanning patterns in program comprehension" *In Proceedings of the 39th Hawaii International Conference on System Sciences (HICSS) ,Kauai, Hawaii*, 2006.

[7] Sharif, Bonita, Michael Falcone, and Jonathan I. Maletic. "An eye-tracking study on the role of scan time in finding source code defects." *Proceedings of the Symposium on Eye Tracking Research and Applications*. ACM, 2012.

[8] Turner, Rachel, et al. "An eye-tracking study assessing the comprehension of C++ and Python source code."*Proceedings of the Symposium on Eye Tracking Research and Applications*. ACM, 2014.

[9] Divya Venugopal, Amudha J, C Jyotsna, "Developing an application using Eye Tracker",*IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology*, 2016

[10] Voßkühler, A., Nordmeier, V., Kuchinke, L., & Jacobs, A.M. "OGAMA - OpenGazeAndMouseAnalyzer: Open source software designed to analyze eye and mouse movements in slideshow study designs." *Behavior Research Methods*, 2008