Do Computer Programmers With Dyslexia See Things Differently? A Computational Eye Tracking Study

Ian McChesney
School of Computing and Mathematics
Ulster University
Newtownabbey, County Antrim, BT37 0QB
Northern Ireland
+44 28 9036 6129
ir.mcchesney@ulster.ac.uk

Raymond Bond
School of Computing and Mathematics
Ulster University
Newtownabbey, County Antrim, BT37 0QB
Northern Ireland
+44 28 9036 6129
rb.bond@ulster.ac.uk

ABSTRACT
In this extended abstract, we describe the early stages of a study using computational eye gaze analysis to investigate the gaze behaviour of computer programmers with dyslexia.

CCS Concepts
• Software and its engineering~Maintaining software • Social and professional topics~Computing education programs

Keywords
Eye Tracking; Gaze Analysis; Program Comprehension; Dyslexia

1. INTRODUCTION
Dyslexia is defined as "a specific learning difficulty which affects the ability to recognize words fluently and/or accurately; causes problems with spelling, auditory short-term memory, phonic skills, multi-tasking, remembering instructions, and organizational skills" [4]. Approximately 10% of the population live with dyslexia and individuals with dyslexia experience the condition in different ways, and there is much debate surrounding its identification and support [1].

Computer programming is primarily a text-based activity, and as such it may present additional challenges to the dyslexic programmer over and above the typical cognitive challenges of software development. The challenges faced by programmers with dyslexia have been described both in relation to learning to program [8],[9] and working in industry [3],[7]. However, there is limited empirical work on the program comprehension strategies of programmers with dyslexia. Consequently there are many research questions to be addressed in this area; for example, how do models of reading such as the Dual Route Model [6] apply when reading program code? How does the visual aspect of program code (indentation, camel case, and IDE features) assist programmers with dyslexia? Furthermore, can cognitive assistants or better artificial intelligent agents be developed to assist neurodiverse programmers?

The exploratory study described here, currently in its early stages, is using eye tracking technology to gather data on the gaze behaviour of programmers with dyslexia.

2. STUDY ORGANIZATION
Subjects recruited from our undergraduate computing courses were presented with three programs based on the protocol used in the EMIP 2014 workshop [2] (referred to here as the Cake, CalcAvg and PrintRow programs). In this phase of the study we have 7 computer programmers with dyslexia and 5 in the control group. Recording sessions used the Tobii x60 eye tracking device and related Tobii Studio software (v2.3). For each program, subjects were presented with an instruction screen, the code screen and an evaluation screen, and were asked to verbalize their reading and understanding of the code (by thinking-aloud). After interpreting each program, subjects were also asked to rate how confident they were in their understanding of the program (1-low, 10-high). Subjects completed a short questionnaire which collected their personal characteristics such as their programming experience and preferred programming languages. The questionnaire also asked the subject if they had dyslexia and, if so, to self-rate this as mild, moderate or severe. The study was approved by our Faculty Ethics Filter Committee.

3. QUALITATIVE OBSERVATIONS
Each recording session was reviewed by the authors and using specified criteria the subject's understanding of each program was graded on a scale of 1-10 (1-low, high-10). Using the EMIP coding structure as a vocabulary toolbox, each program session was characterized in terms of the general pattern and sequence of gaze exhibited. While recognizing the subjective nature of this assessment, initial analysis using the coding structure did not show any distinct differences between the two groups.

4. QUANTITATIVE ANALYSIS
Using the Tobii Studio (v2.3) software, a range of visualizations and metrics have been explored.

4.1 Heat Maps
In Figure 1 (Cake program), we can see that the dyslexia group gaze (count) shows a distinct high volume of fixations on the modulus operator on line 5, with a focus just below the line of code. For the control group, gaze is focused on lines 4 and 5, namely the For loop header and following If statement condition respectively, with a focus again just below the lines of code.

For the CalcAvg program, the control group gaze is in the middle of the method, concentrated on line 7. For the dyslexic group, this is more vertically diffuse across lines 3-11.

For the PrintRow program, the dyslexia group and the control group both exhibit gaze which is concentrated on the outer and inner For loops headers. In the control group, gaze is concentrated directly on the variables `row` (in the outer For header) and `col` (in the inner For header). Interestingly, in the dyslexia group, this gaze is more diffuse across these two lines, and is also concentrated "between" the lines (between line 3-4 and lines 4-5).

Computing heat maps within the first 5 seconds of viewing can depict a programmer's automatic intuition and Gestalt perception in understanding a program. For the Cake and CalcAvg programs, a qualitative pattern is that the dyslexia group tend to limit their scan to the initial lines in the programs (LinearHorizontal), whereas for the control group gaze is more vertically dispersed

(LinearVertical). This is not the case however for the PrintRow program where the dyslexia group gaze is, albeit briefly, drawn to the method call in the public static void main method at the bottom of the screen –this is not the case for the control group.

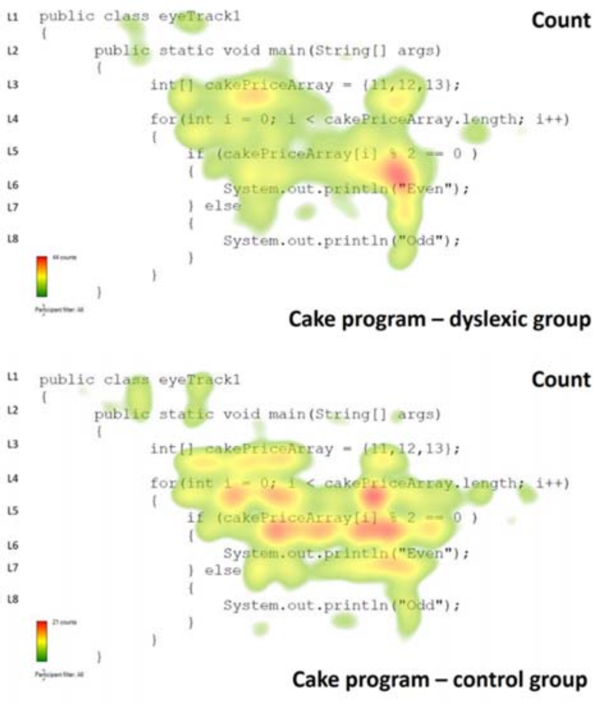

Figure 1. Heatmaps (count) for Cake program

4.2 Metrics

Considering Total Fixation Duration (TFD) and Fixation Frequency (FF) metrics at the program level, there appears to be no significant difference between the dyslexia and control groups for any of the three programs (Table 1, TFD, two-sample t-test).

Table 1. Total Fixation Duration, two-sample t-test (0.05 confidence level)

		Dyslexia group	Control group
Cake	$\bar{x}$ (σ^2)	53.02s (305.52s)	46.69s (309.50s)
	P	0.2756	
CalcAvg	$\bar{x}$ (σ^2)	77.60s (2175.76s)	60.69s (767.99s)
	P	0.2254	
PrintRow	$\bar{x}$ (σ^2)	88.01s (1942.97s)	81.74s (1777.81s)
	P	0.4049	

Each line of the program (ignoring braces) was defined as an area of interest (AOI). For these, metrics First Fixation Duration (FFD) and TFD were evaluated. Significant differences in gaze duration were revealed in only a small number of AOI instances as shown in Table 2. Considering FFD, in each significant case, the dyslexia group spent less time on the AOI than the control group. For TFD, the only significant difference was in the Cake program, line 6, where the dyslexia TFD duration was greater than the control group.

Each program interaction consisted of reading an instruction screen followed by reading the program code screen. There was no emerging pattern between the two groups regarding overall reading time during the experiment.

Table 2. AOIs with distinct gaze patterns [(P)rogram, (L)ine]

	AOI	Dyslexia group ($\bar{x}$)	Control group ($\bar{x}$)	p	Line description
FFD	P1, L4	0.2086s	0.3000s	0.02	`For loop iterating over array`
	P2, L10	0.2300s	0.3700s	0.06	`system.out.println ("Average")`
	P3, L2	0.1414s	0.2060s	0.02	`public static void printMethod`
TFD	P1, L6	5.1029s	2.0560s	0.08	`system.out.println ("even")`

Some correlations (using Pearson's correlation coefficient) between experiment variables have been investigated. Line length (number of characters) is used as a proxy for complexity (other measures could be used, e.g. Halstead's complexity metrics). For the Cake program, both groups displayed similar behaviour, namely the longer the line length the greater TFD (dyslexia group, r=0.62, control group, r=0.71). For the CalcAvg program, this pattern held, albeit weakly, for the dyslexia group (r=0.20), whereas for the control group it was the opposite case (a weak negative correlation, r= - 0.01), i.e. for the control group, the longer the line length, the lower the TFD. For PrintRow, both groups exhibited a positive correlation on these variables, but weaker for the control group (r=0.15) compared with the dyslexia group (r=0.42).

On completion of the experiment, the authors assessed each subject's understanding of the code on a scale 1-10 (1-unable to explain program, 10-full explanation of program given). Correlation of this score to the TFD showed no overall trend. However, limiting cases to those who did not demonstrate a good understanding of the programs (scoring 6 or less), there appears to be a slight negative correlation with FF (r= - 0.27), i.e. the lower the understanding, the higher the FF values at the program level.

5. INITIAL OBSERVATIONS

There are many limitations in the study at this stage, most obviously the number of subjects. However, it does show that there are many interesting avenues to explore in comparing the gaze of programmers with and without dyslexia. Arguably, most notable of all at this stage, is that there appears to be no striking differences in how these two categories of programmer read code?

6. NEXT STEPS

Further eye tracking sessions have been scheduled to increase the number of subjects. It is hoped a broader data set, and drawing upon related research in other fields (e.g. [5]) will help identify fruitful research questions for further investigation. Does reading program code employ the same visual and cognitive models as reading text and how is this exhibited by a programmer with dyslexia? Do the visual, orthographic and phonetic differences in program code compared with prose make it easier for a programmer with dyslexia to comprehend a program? Do programmers with dyslexia see things differently?

7. REFERENCES

[1] Armstrong, D. and Squires, G., 2014. *Key Perspectives on Dyslexia: An essential text for educators*. Routledge.

[2] Busjahn, T., Schulte, C., Tamm, S. and Bednarik, R., 2015. Eye movements in programming education II: *Analyzing the novice's gaze. Technical Report TR-B-15-01*, Freie Universität Berlin, Department of Mathematics and Computer Science, Berlin, Germany.

[3] Coppin, P., 2008. Developing drawing and visual thinking strategies to enhance computer programming for people with dyslexia. In *Visual Languages and Human-Centric Computing, 2008. VL/HCC 2008. IEEE Symposium on* (pp. 266-267). IEEE.

[4] "dyslexia." In *A Dictionary of Education*, edited by Wallace, Susan. : Oxford University Press, 2015.

[5] Kim, S., Lombardino, L.J., Cowles, W. and Altmann, L.J., 2014. Investigating graph comprehension in students with dyslexia: An eye tracking study. *Research in developmental disabilities*, 35(7), pp.1609-1622.

[6] Law, C. and Cupples, L., 2017. Thinking outside the boxes: Using current reading models to assess and treat developmental surface dyslexia. *Neuropsychological rehabilitation*, 27(2), pp.149-195.

[7] Morris, M.R., Begel, A. and Wiedermann, B., 2015. Understanding the challenges faced by neurodiverse software engineering employees: Towards a more inclusive and productive technical workforce. In *Proc. 17th Intl ACM SIGACCESS Conf.on Computers & Accessibility (pp. 173-184). ACM.*

[8] Powell, N., Moore, D., Gray, J., Finlay, J. and Reaney, J., 2004. Dyslexia and learning computer programming. *Innovation in Teaching and Learning in Information and Computer Sciences*, 3(2).

[9] Stienen-Durand, S. and George, J., 2014. Supporting dyslexia in the programming classroom. *Procedia Computer Science*, 27, pp.419-430.