Empirical Software Engineering (2019) 24:1109–1154
https://doi.org/10.1007/s10664-018-9649-y

Eye tracking analysis of computer program comprehension in programmers with dyslexia

Ian McChesney[1] · Raymond Bond[1]

Published online: 10 September 2018

Abstract

This paper investigates the impact of dyslexia on the reading and comprehension of computer program code. Drawing upon work from the fields of program comprehension, eye tracking, dyslexia, models of reading and dyslexia gaze behaviour, a set of hypotheses is developed with which to investigate potential differences in the gaze behaviour of programmers with dyslexia compared to typical programmers. The hypotheses posit that, in general terms, programmers with dyslexia will show gaze behaviour of longer duration and a greater number of fixations on program features than typical programmers. An experiment is described in which 28 programmers (14 with dyslexia, 14 without dyslexia) were asked to read and explain three simple computer programs. Eye tracking technology is used to capture the gaze behaviour of the programmers. Data analysis suggests that the code reading behaviour of programmers with dyslexia is not what would be expected based on the dyslexia literature relating to natural text. In conjunction with further exploratory analysis, observations are made in relation to spatial differences in how programmers with dyslexia read and scan code. The results show that the gaze behaviour of programmers with dyslexia requires further study to understand effects such as code layout, identifier naming and line length. A possible impact on dyslexia gaze behaviour is from the visual crowding of features in program code which might cause certain program features to receive less attention during a program comprehension task.

Keywords Computer programming · Program comprehension · Eye tracking · Dyslexia

Communicated by: Emerson Murphy-Hill

✉ Ian McChesney
ir.mcchesney@ulster.ac.uk

Raymond Bond
rb.bond@ulster.ac.uk

[1] School of Computing, Ulster University, Newtownabbey, UK

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" (OUP 2015). Approximately 10% of people live with dyslexia (Sexton et al. 2012). Individuals with dyslexia experience the condition in different ways and there is much debate surrounding its identification and support (Armstrong and Squires 2014). Computer programming is primarily a text-based activity and as such, it may present additional challenges to the programmer with dyslexia over and above the normal cognitive challenges of software development. The impact of dyslexia on programming tasks, either learning to program or professional programming practice has been investigated directly and indirectly by a number of researchers. Powell et al. (2004) consider its impact on programming in terms of both its negative aspects (such as poor handwriting, spelling and short term memory) which can lead to reading deficiencies, and its positive manifestations (such as strong visualization, spatial awareness, and creativity) which characterize positive alternative learning styles. Powel et al. propose a mapping between these characteristics and stages in the program development process, suggesting that for tasks such as problem definition and system design, traits such as visualization and creativity bring benefits, whereas, for tasks related to coding and testing, traits such as poor spelling and short term memory are disadvantageous. Their mapping is supported by qualitative and anecdotal evidence from conversations with programmers with dyslexia. The link between the strong visual-spatial processing of a programmer with dyslexia and their ability to effectively problem solve in a programming context is also noted by Coppin (2008), who extends this observation to suggest how a workspace can be designed to capitalize on these traits (Coppin and Hockema 2009).

In a wider context, there is a long established research interest in the link between computer programming and personality. This has ranged from its relevance to the individual programming task (Bishop-Clark 1995), through to its impact on pair programming (Salleh et al. 2014) and into the wider sphere of team-based software engineering (Cruz et al. 2015). A recent line of enquiry has been in relation to learning disabilities, across the spectrum, and their impact on the individual's approach to computer programming. Morris et al. (2015) present results from a survey of professional programmers who have a range of conditions such as autism spectrum disorder, attention deficit hyperactivity disorder and dyslexia. Results from interviews with 10 neurodiverse technology workers and from a survey of a further 59 neurodiverse technologists are presented. The work reported refers to challenges they face during software development, such as rigid interpretation of rules, difficulty committing to certain types of tasks perceived as mundane or expression of, at times, inappropriate emotions. Though the number of programmers with dyslexia in the survey was small (16 identifying with dyslexia or other learning difficulties, other than Asperger Syndrome, Attention Deficit Disorder or Attention Deficit Hyperactivity Disorder), it represents a significant empirical attempt at identifying how neurodiverse programmers approach programming in ways which are different from the neurotypical programmer. For example, when asked to self-rate their skill at certain programming tasks, neurodiverse programmers' self-rated skill was significantly higher in tasks such as detecting patterns in code and adopting good programming style, whereas they were self-rated as less skilled in, for example, reviewing other's code and writing test cases. If it is possible to identify ways in which programmers with dyslexia engage with programming which are not typical, then the workplace in general, and software engineering tools in particular, can be adapted to support these ways of working.

This paper contends there is a need for empirical work in understanding how programmers with dyslexia actually develop, test and comprehend program code. The primary research question here is, when reading program code for the purpose of comprehension, do the eye movements of programmers with dyslexia differ from those of programmers with typical reading profiles? In pursuing this question, other subsidiary questions become apparent which cannot be answered directly from the study described here but are noted as areas for further investigation. For example, do models of reading such as the Dual Route Model (Coltheart et al. 1993) apply when reading program code? How does the visual aspect of program code (indentation, camel case and code editor features) assist programmers with dyslexia? Do orthographic and phonological deficiencies, as exhibited by readers with dyslexia when reading prose, persist as deficiencies when reading program code? If so, are such deficiencies amplified or attenuated by the external representation of the program and/or the mental models at work in program comprehension? To seek to answer the primary question, this exploratory study uses eye tracking technology to gather data on the gaze behaviour of programmers with dyslexia during code reading and program comprehension tasks.

The paper is organized as follows. In Section 2 related work from a number of areas is drawn together to help formulate the hypotheses for the study. This work is reviewed in relation to the role of eye tracking in program comprehension studies, reading models and eye movements, and eye movement studies of readers with dyslexia. Informed by this work, the study design is presented in Section 3, including the hypotheses which have been formulated to guide the enquiry. This is followed in Section 4 by a detailed presentation of the results arising from the experiment eye gaze data. The discussion in Section 5 explores possible interpretations of the results in relation to the code reading behaviour of programmers with dyslexia for the three programs in the study. Section 6 identifies threats to the validity of the study after which overall conclusions and areas for further investigation are presented in Section 7.

2 Related Work

2.1 Program Comprehension Models

Program comprehension is an established area of research within the discipline of computer science (Brooks 1978; Shneiderman and Mayer 1979; Shaft and Vessey 1995). Its study seeks to explicate the factors at work when a programmer reads program source code to understand its overall purpose and to identify the particular syntactic and semantic components from which the program is constructed. Program comprehension is a function of properties of the programmer, such as their cognitive processes and programming language experience, and properties of the program artifact, such as code layout, identifier naming style or the code editor in use. Various models of program comprehension have been developed to reflect the range of cognitive strategies adopted by programmers. For example, bottom-up models propose that programmers seek to understand individual statements and program features and then assimilate these into higher level semantic blocks of code (Shneiderman and Mayer 1979; Pennington 1987). Top-down models propose that an initial view of the program's purpose is formed, for example by using recognizable constructs in the code and then reading individual statements to support, reject or refine this initial view (Brooks 1983). In practice, an integrated approach may be used, with programmers switching between levels of abstraction as they move towards an understanding of a program's purpose (Von Mayrhauser et al. 1997). Maalej et al. (2014) found that in real-world settings, professional programmers adopt sophisticated program comprehension strategies which involve not only bottom-up and top-down strategies but also viewing a program's behaviour from the user's perspective thereby constructing a mental model of the program by visualizing its input and output.

Schulte et al. (2010) suggest that the range of program comprehension models which have been proposed have a number of elements in common. These are (i) the external representation of the program. This is typically the program source code but can also include representations such as class diagrams and dynamic code inspectors; (ii) an assimilation process by which a programmer views the external representation and assembles the building blocks for (iii) an internal, cognitive representation of the program, complemented by existing mental models and cognitive structures which are part of the programmer's experience and problem solving capacity (Fig. 1). With reference to this simple framework, the focus of this study is the assimilation process of the programmer with dyslexia as she reads the program artifact and seeks to build an understanding, using her cognitive model, of the purpose of the program. Specifically, when reading program code for the purpose of comprehension, do the eye movements of programmers with dyslexia differ from those of programmers with typical reading profiles?

2.2 Program Comprehension and Eye Tracking

In recent years eye tracking has been used as a mechanism for direct measurement of the reading processes of programmers and, from the data generated, for inferring strategies of program comprehension. It is accepted that eye gaze is a strong indicator of attention (Rayner 2009; Reichle and Sheridan 2015). As such, when used to study the reading of program code, eye movement gives an insight into the reading behaviour of the programmer and the mental model she is constructing. Bednarik and Tukiainen (2006) used eye tracking to identify differences in program comprehension strategies between expert and novice programmers when reading a program in conjunction with an execution visualization tool. They found that

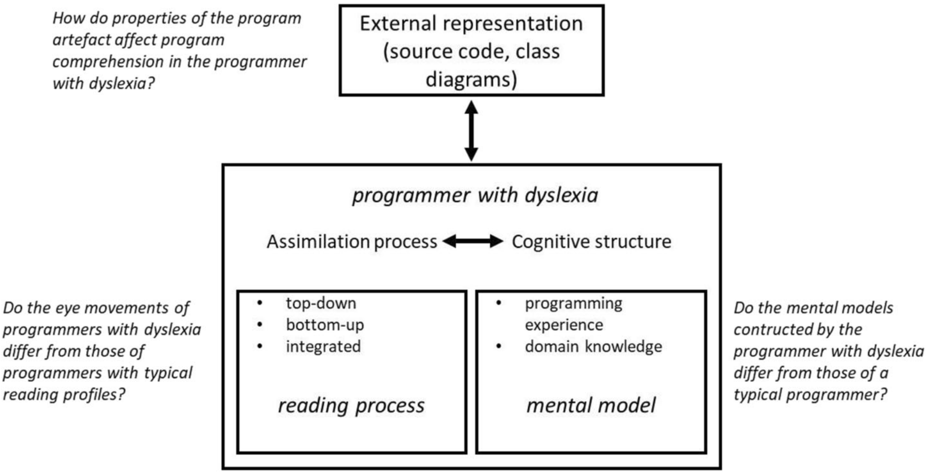

How do properties of the program artefact affect program comprehension in the programmer with dyslexia?
External representation (source code, class diagrams)
programmer with dyslexia
Assimilation process
Cognitive structure
• top-down
• bottom-up
• integrated
reading process
• programming experience
• domain knowledge
mental model
Do the eye movements of programmers with dyslexia differ from those of programmers with typical reading profiles?
Do the mental models contructed by the programmer with dyslexia differ from those of a typical programmer?

Fig. 1 Extension of Schulte et al.'s (2010) representation of key elements of program comprehension, annotated (italics) from the perspective of the programmer with dyslexia

an experienced programmer's approach to understanding was to read the code first, then confirm their mental model by running the visualization. Novice programmers had a greater reliance on the visualizer to aid understanding. Busjahn et al. (2011) conducted a comparison of reading natural text and reading program code using eye tracking. They observed some differences when reading normal text compared with reading program code, exhibited by differences in key gaze metrics such as mean fixation times and the number of regressions. Whereas reading natural text generally proceeds in a linear fashion, leading to serial-attention reading models such as the E-Z Reader Model (Reichle and Sheridan 2015), reading program code appears to be a mixture of linear and non-linear reading behaviour. The study described in Busjahn et al. (2015) further showed a combination of linear and non-linear behaviors, with notable differences between novice and expert programmers. Novice programmers showed a "fairly strong linear character" with 70% of their eye movements on source code being linear, compared with 60% for expert programmers. It is suggested this reflects the experts' ability to follow the execution order of a program and/or to seek out beacons in the code as an aid to understanding. Sharma et al. (2012) studied gaze transitions between the three program elements of identifiers, structural elements (e.g., loops) and expressions. Findings suggested that the gaze of those who understood a program was focused on transitions between identifiers and expressions, reflecting a control flow or execution-based reading of the code. Those who did not exhibit a good understanding of the program tended towards a systematic, structural reading of the code. Other work has also shown differences between reading natural text and program code; for example, in natural text reading, there is a correlation between first fixation duration and word frequency. The less frequent the word in the lexicon, the greater the first fixation duration. However, with respect to keywords in Java, keyword frequency is not a predictor of first fixation duration (Busjahn et al. 2014a). Jbara and Feitelson (2017) used eye tracking to compare the reading of regular code and non-regular code. They found that reading is done non-linearly using scan patterns such as scanning and jumping ahead. Binkley et al. (2013) report a series of experiments investigating the impact of identifier style on code comprehension. As part of this, they also considered the differences in reading natural text and program code. They concluded that reading natural text and reading code are fundamentally different processes – on the basis that the representational structure of code (such as indentation and white space) and code beacons enable programmers to assimilate and understand parts of a program quite quickly – a phenomenon less common in natural language texts.

The First International Workshop on Eye Movements in Programming Education (Bednarik et al. 2014) devised a coding scheme to describe gaze behaviour when reading program code. This scheme is useful for illustrating the ways in which reading code is different from reading natural text. The scheme includes the notion of gaze patterns to describe sequences of fixations. Patterns can be linear, for example *LinearHorizontal* (where a programmer reads elements in a whole line of code in an equally distributed time pattern), or non-linear, for example *Flicking* (where gaze moves back and forth between two related items), *JumpControl* (movement to the next line according to execution order), and *LinearVertical* (following the code line by line). The categories of the coding scheme provide a valuable vocabulary for describing the non-linear components of reading code at the program level (Busjahn et al. 2014b).

Other work has used eye tracking technology to study aspects of the software development process other than programming. Recognizing that most real-world software development involves complex programs spanning multiple screens and files, Sharif et al. (2016) describe iTrace, a tool for enabling the use of eye track technology when the software artifact is not a static representation on screen but rather a dynamic artifact such as a scrolling code listing or the folder structure in a code editor. Using iTrace, Kevic et al. (2017) investigated software change tasks. As well as eye tracking data, code editor interaction data was collected. They found that in a software change task, developers only looked at very few lines of code within a program subroutine. Also, developers "chase" variable flow (execution flow) within code. This is consistent with the patterns of expert gaze already mentioned. In their work, Rodeghero et al. (2014) seek ways to augment automated code summarization tools by using data from the programmer's gaze when performing summarization tasks (gaze time, number of fixations and regressions). Results include the observation that professional programmers exhibited a preference regarding the type of code regions they read. Rather than focusing on control flow (as suggested by Sharma et al. 2012), professional programmers tended to focus on method signatures and the code locations from where the methods were called. Ali et al. (2012) investigated the construction of requirements traceability links between requirements and source code. By identifying the sections of source code which developers focused on when verifying requirements, using metrics such as total fixation duration, they sought to find better ways of constructing accurate links between source code entities and their originating requirements. The use of eye tracking in software development research is not limited to studying gaze on source code. De Smet et al. (2014) describe three experiments investigating the impact of widely used program design patterns on the time and effort to perform maintenance and program comprehension tasks. Eye tracking technology was used to record participant's gaze behaviour (fixation duration) when looking at various types of program structure diagrams. In keeping with findings from the program comprehension work described earlier, novice programmers tended to browse structure diagrams systematically whereas experts used their experience to scan and gather the salient information more quickly.

2.3 Reading Models and Dyslexia

The reading of program code has similarities and differences to the reading of natural text. While it does have some linear characteristics, it is also characterized by scanning, jumping and regression. Nevertheless, the assimilation process does require a reading capability. Dyslexic readers exhibit deficiencies when they read and comprehend natural text. To paraphrase the research question from the introduction, do programmers with dyslexia read and comprehend program code differently from programmers who do not have dyslexia? Do programmers with dyslexia see things differently?

The Dual Route Model (DRM) of reading is a widely accepted abstraction of the reading process (Coltheart et al. 1993; Coltheart et al. 2001; Law and Cupples 2017). The first stage of reading is orthographic visual analysis and letter identification. The model describes the next stage of reading as taking place through two separate processes, or routes, from print to speech. The so-called direct or "lexical" route involves the reader, having visually acquired the word to be read, look up this word in her orthographic lexicon – the set of words she has previously recognized through reading. The indirect or "non-lexical" route involves the reader, having visually acquired the word to be read, applies explicit conversion rules for parsing the word into graphemes and their corresponding phonemes. These phonemes are combined to form the word. Both routes are active when reading is taking place. However, exception words (words that do not conform to standard phonetic rules, such as "tough" or "know"), are only processed through the lexical route as they do not conform to the reader's grapheme-phoneme mapping. Words which have not been encountered previously by the reader, i.e., are not part of her orthographic lexicon, are processed using the non-lexical route, leading to a successful or unsuccessful attempt at reading the new word.

Considering the Dual Route Model when reading program code, typical reading events would include reading familiar words, such as program language keywords, which according to the model, would be processed using the lexical route. This would include exception words such as new or byte. Words not previously encountered can be common in program source code, especially when reading code written by someone else. For example, the identifier name cakePriceArray would, according to the DRM, be processed through the non-lexical route, though because of its compliance with English grapheme-phoneme mapping rules, would typically be processed without difficulty at the word level.

As suggested by the Dual Route Model, dyslexia itself is a multi-faceted condition that has many subtypes which can be present to varying degrees in the reader. Friedmann and Coltheart (2016) provide a comprehensive summary of the types of dyslexia using the Dual Route Model as a reference framework. Deficits in the orthographic visual analysis stage of reading are examples of *peripheral dyslexia* (also known as visual dyslexia). These include letter position dyslexia, attentional (letter migration) dyslexia, letter identity dyslexia (the reader cannot abstract a letter), and neglect dyslexia (neglecting one side of a word). Deficits in the lexical and non-lexical routes of the model are described as *central dyslexia*. Examples include surface dyslexia which is a deficiency in the lexical route of the model. In such cases, the reader will have difficulty reading words such as "receipt", "new" or "gnu". Phonological dyslexia arises from a deficiency in the non-lexical (phonetic) route of the model whereby reading can only proceed via the lexical route, leading to a difficulty in reading new or non-words. Friedmann and Coltheart note in particular that readers with this type of dyslexia "usually encounter this severe difficulty again when they learn to read a new language". Other examples of central dyslexia relate to deficits in the "phonological output buffer" such that the reader cannot properly read, process or articulate long words. Deep dyslexia describes semantic errors or erroneous word associations such as reading, in a programming context, "variable" as "value", or "get" as "set".

There is an extensive body of work related to dyslexia and possible interventions, a review of which is beyond the scope of this paper. Refer to Pennington and Peterson (2015) for an overview.

2.4 Eye Movements in Reading

Reading models such as the Dual Route Model have been extended to take account of eye movements when reading. Schroeder et al. (2015) state that with respect to reading, monitoring eye movements is "an excellent tool to help us understand how comprehension during reading takes place via interactions between visual and language processing systems". Radach and Kennedy (2013) have noted three perspectives from which eye movement in reading research has been conducted. There is research which has focused on visual processing and sensori-motor control, for example, the relationship between vision, attention and saccade preparation. The second category of research is informed by cognitive science, focusing on reading as an information processing and word-level processing activity. The third category is research which has used direct measurement of eye gaze to develop and test hypotheses.

Certain types of gaze metrics can be associated with particular stages in the Dual Route Model. For example, first fixation duration measures can be associated with early stage orthographic processing; gaze duration can be associated with later stages of the model such as lexical access. Many eye movement measures used in the analysis of reading are temporal in nature. Early orthographic processing time can be inferred by first fixation duration on a word. Later stages of reading a word, including lexical analysis, can be related to total fixation duration on the word. Reading processes concerned with word integration and sentence semantics can be inferred from metrics such as total viewing time and regression path duration.

Other gaze metrics are spatial in nature. For example, the length of the target word, launch distance of a saccade, and position of the target word on the line of text. The extent to which such eye movement measures can infer cognitive processes when reading or can explain the essence of the reading process is the subject of much debate. For example, computational models of the Dual Route Model differ in their assumptions regarding reading as a sequential or parallel activity. In sequential attention shift models such as E-Z Reader (Reichle and Sheridan 2015), the processing window is one word wide. In a parallel processing model, perceptual spanning across a word boundary can be processed in parallel (Engbert et al. 2005). The case for sequential models includes the fact that, for example, attention is necessary to combine features of words into a unitary representation, that the sequential order of word recognition aligns with grammatical order (facilitating comprehension), and that the lexical processing of multiple words is not adequately described with any existing model. However, there is evidence that letter processing within words is conducted in parallel (Adelman et al. 2010).

2.5 Eye Movements Associated with Dyslexia

Eye movement data provides an insight into the reading process. Conceptual and computational reading models provide a theoretical framework in which this can be understood. It follows that eye movement data pertaining to the reading behaviour of dyslexic readers can provide some empirical basis for distinguishing this behaviour from that of typical readers.

Bellocchi et al. (2013) present a review of the literature pertaining to eye movement reading behaviour in developmental dyslexia. The nature of the link between dyslexia and eye movement is still under debate. Some of the observations are characteristics simply of younger readers and some characteristics disappeared when the task was not reading but rather requiring sequence or pattern recognition. Their review is presented in terms of research conducted in three broad areas. First, studies which have focused on visual motor behaviour have found that:

(A) At word, pseudoword or sentence level, dyslexic eye movements are characterized by more and longer fixations, shorter saccades, and more regressions. (e.g., Hawelka et al. 2010).

(B) Dyslexic eye movements show a smaller number of words that receive a single fixation or are skipped, a greater number of words with multiple fixations, a marked effect of word length on gaze duration, and prolonged gaze durations for singly fixated words (e.g., de Luca et al. 2002).

With reference to the Dual Route Model, these findings are interpreted as a failure of orthographic whole word recognition and an inefficient lexical route.

Second, several studies have found defective visio-attentional processes in dyslexia such that:

(C) Dyslexic readers are influenced more by crowding (visual distractions around the centre of the word target) (Spinelli et al. 2002) and that inter-letter, inter-word spacing improves legibility for dyslexics (e.g., Perea et al. 2012).

(D) However, crowding has a confounding effect. It affects some dyslexics more than others. Those with a moderate reading deficit tend to be sensitive to crowding. Those with a severe reading deficit tend not to be sensitive to crowding.

(E) The dyslexic reader exhibits sluggish attentional shifting associated with deficits in spatial position encoding, affecting phonological representation (e.g., Hari and Renvall 2001). There can be asymmetrical allocation of attention to the right visual field in dyslexia, resulting in a so called left mini-neglect phenomenon (Facoetti and Molteni 2001).

(F) Dyslexic readers can only process a few letters at each fixation, suggesting that a smaller visual-attention span prevents dyslexics from processing many letters simultaneously (Prado et al. 2007). However, this was not true for non-reading tasks such as visual search, leading to the conclusion that the observed differences between normal and dyslexic readers may apply only to text reading.

When considering eye movement behaviour relating to saccades, studies have shown the existence of an optimal viewing position (OVP) to maximize the efficiency of word recognition which, for normal readers, is slightly to the left of the word's centre, with recognition efficiency decreasing on both sides of this point.

(G) For dyslexic readers, there appears to be an absence of this left-right asymmetry in the OVP when initially fixating upon a word. Rather than the saccade landing on the OVP, it tends to land in the middle of the word, suggesting dyslexic readers are less able to focus on the OVP as the most information rich part of the word (Ducrot et al. 2003).

(H) Positioning errors are more frequent for the dyslexic reader, leading to more refixations (Hawelka et al. 2010).

Bellocchi et al. (2013) argue that dyslexia can be best observed and described using (a) characteristics of global eye movement measures (number of fixations, fixation duration) and (b) characteristics of specific eye movement measurements relating to OVP and saccade landing sites, as indicators of attention allocation during reading or word identification, notwithstanding the heterogeneous nature of dyslexia manifestations and causes.

2.6 Summary of Related Work

Reflecting on the work described in the previous sections, Fig. 2 summarizes the contribution of these research areas to the present study. The relationship between dyslexia and the programming task has been considered in other studies (see Section 1). However, there has been no empirical work on the gaze behaviour of programmers with dyslexia. The program comprehension and eye tracking literature has shown that reading code involves both sequential and scanning reading patterns. Scanning will typically be guided by the program structure and its control flow, with sequential reading taking place at the word and pseudoword level. Reading models provide a framework for understanding the different types of dyslexia, with deficiencies arising in different circumstances depending on the need to process, for example, exception words, new words or non-words – scenarios which are common when reading computer program code. The literature on the eye movement of dyslexic readers enables the formulation of hypotheses to test for differences in the reading behaviour of programmers with dyslexia compared with typical programmers.

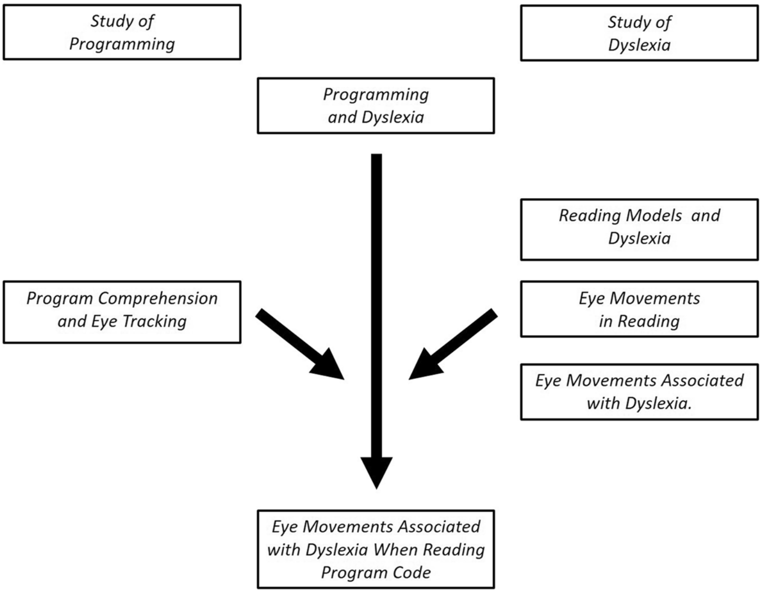

Study of Programming
Study of Dyslexia
Programming and Dyslexia
Reading Models and Dyslexia
Program Comprehension and Eye Tracking
Eye Movements in Reading
Eye Movements Associated with Dyslexia.
Eye Movements Associated with Dyslexia When Reading Program Code

Fig. 2 Summary of research areas contributing to study

However, there are limitations in taking findings from the realm of reading natural text and applying these to the reading of program code. Many of the studies investigating eye movement in dyslexia have been conducted under tightly controlled experimental conditions in terms of how gaze objects such as word lists, letters and rapid automatized naming (RAN) tasks can be manipulated. Also, many of the experiments in developmental dyslexia have been based on the observation of children's reading performance. In this study, the reading artifact (program code) is static in nature and the research is conducted using adults with dyslexia. Nevertheless, reading models and the related dyslexia research provide a reasonable starting position for exploring how programmers with dyslexia might read code. It enables the formulation of hypotheses regarding eye movement in order to explore potential differences in code reading behaviour amongst programmers with dyslexia and typical programmers. This study uses the observations A, B and E from section 2.5 above as the basis for formulating hypotheses testable using program code gaze metrics. Observations C,D,F,G and H are less straightforward in terms of their formulation into testable hypotheses given the experimental design described here. Exploration of these observations will require further study.

When measuring eye gaze activity of natural text reading, the unit of observation is typically the word, pseudoword or sentence construct. In terms of reading program code, the unit of observation adopted here is that of a program code feature, which may be an identifier, keyword or line of code, depending on context. The hypotheses of this study are formulated in terms of eye gaze behaviour in relation to such features and are presented in section 3.1 below.

3 Study Design

The following sub-sections present the hypotheses which have been formulated to guide the study and the experimental setting in which these were tested. In summary, the experiment involved a study group (14 programmers with dyslexia) and a control group (14 programmers without dyslexia). The participants were presented with three unseen Java programs, and in each case they were asked to read and describe the program's purpose. The experimental session was recorded using an eye tracking device. Before eye gaze recording commenced, participants completed a profiling questionnaire to capture details such as age, programming language experience, and whether or not they had dyslexia. The study was reviewed and approved by the university ethics filter committee and all participants were recruited according to the agreed protocol.[1]

3.1 Hypotheses

3.1.1 Hypothesis 1

Based on the observation (A) in Section 2.5 above, at the word, pseudoword or sentence level, dyslexia eye movements are characterized by more fixations:

> $H1_0$ – Programmers with dyslexia have the same number of fixations on program code features as the control group.
> $H1_1$ - Programmers with dyslexia have a greater number of fixations on program code features than the control group.

3.1.2 Hypothesis 2

Based on the observation (A) above, at the word, pseudoword or sentence level, dyslexia eye movements are characterized by longer fixations:

> $H2_0$ – Programmers with dyslexia have fixations on program code features of the same duration as the control group.
> $H2_1$ - Programmers with dyslexia have fixations on program code features with greater duration than the control group.

3.1.3 Hypothesis 3

From observation (B), when reading at the word, pseudoword or sentence level, dyslexia eye movements are characterized by more regressions:

> $H3_0$ - Programmers with dyslexia have the same number of gaze visits to program code features as the control group.
> $H3_1$ - Programmers with dyslexia have more gaze visits to program code features than the control group.

[1] An experiment description document, including the consent form, profiling questionnaire and eye tracker media files are available on Figshare. McChesney I, Bond R (2018) Experiment description for "Eye tracking analysis of computer program comprehension in programmers with dyslexia", https://doi.org/10.6084/m9.figshare.6267125

3.1.4 Hypothesis 4

From observation (B), when reading, dyslexic readers exhibit a smaller number of words that receive a single fixation or are skipped:

> $H4_0$ – For programmers with dyslexia, the number of program code features with a gaze visit count of [1|0] is the same as the control group.
> $H4_1$ – For programmers with dyslexia, the number of program code features with a gaze visit count of [1|0] is less than the control group.

3.1.5 Hypothesis 5

From observation (B), dyslexic readers spend more time on longer words and, there is a stronger correlation between word length and fixation duration:

> $H5_0$ – The correlation between identifier length and fixation duration is the same for programmers with dyslexia and the control group.
> $H5_1$ – The correlation between identifier length and fixation duration is stronger for programmers with dyslexia than the control group.

3.1.6 Hypothesis 6

From observation (E), dyslexic readers have an asymmetric visual attention gradient (fixation count), tending to an increased level of attention on the right-hand side (RHS) of a word:

> $H6_0$ – Programmers with dyslexia exhibit the same number of fixations on the RHS of a program code feature as the control group.
> $H6_1$ - Programmers with dyslexia exhibit a greater number of fixations on the RHS of a program code feature than the control group.

3.1.7 Hypothesis 7

From observation (E), dyslexic readers have an asymmetric visual attention gradient (fixation duration), tending to an increased level of attention on the right-hand side (RHS) of a word:

> $H7_0$ – Programmers with dyslexia exhibit the same fixation duration on the RHS of a program feature as the control group.
> $H7_1$ - Programmers with dyslexia exhibit a greater fixation duration on the RHS of a program feature than the control group.

In addition to investigating these hypotheses, the dataset from the experiment has also enabled exploratory data analysis of gaze behaviour across the two groups using metrics not immediately suggested by the dyslexia literature. The exploratory study examined behaviour such as time to first fixation and fixations before, to help identify possible differences in behaviour. This is discussed in section 4.3 below.

3.2 Participants

Participants were recruited from computing undergraduate programmes at Ulster University. A total of 30 participants were recruited for the study. Data was successfully collected from 28 (one study session was void due to computer failure during the session and one due to unsuccessful calibration). 14 participants were programmers with dyslexia (the dyslexia group), 14 did not have dyslexia (acting as the control group). Students were recruited to take part in the study through two types of invitation. One was an email invitation to all students on the institution's undergraduate computing programmes, explaining the need for participants with and without dyslexia. The second type was an email invitation to students registered as dyslexic with the university's student support department. Student support assembled the distribution list for this email from their own records and issued the invitation, requesting that replies be sent directly back to the student support department. They then returned the list of participants to the authors. As an incentive, participants were offered an online gift voucher for taking part. For the purposes of the study, students self-designated as having dyslexia or not on the profiling questionnaire administered at the beginning of each recording session. While the student support dyslexia register was useful in gauging if a sufficient number of students with dyslexia had responded, it was not known with certainty which participants were dyslexic until the study was underway.

Of the 14 participants with dyslexia, there were three female and 11 male. The mean age of the dyslexia group was 23.4 years (*SD* = 6.50). Of the 14 participants without dyslexia (the control group), there were similarly three female and 11 male, with a mean age 21.5 years (*SD* = 3.32).

Participants were asked how long they had been programming. In the dyslexia group, the mean duration was 3.32 years (*SD* = 2.44). For the control group, mean programming experience duration was 2.89 years (*SD* = 1.47).

Participants were also asked to rate as low, medium or high (i) their overall programming expertise and (ii) their programming expertise in Java. The responses are summarized in Table 1.

For the dyslexia group, the profiling questionnaire asked participants to rate their own dyslexia condition as mild, moderate or severe. As illustrated in Table 2, four participants reported their dyslexia as mild, seven moderate, and three severe. Participants were also asked to give an example of the most problematical aspect of their dyslexia.

The profiling questionnaire also asked each participant "How tired do you feel just now?" and to indicate their response of a scale of 1–10 where 1 = very tired and 10 = energetic. For the purpose of comparing the difference if any between the two groups, this was treated as an interval scale. When analyzed using a two-tailed independent samples t-test for equality of means at a significance level of 0.05, no significant difference in fatigue was found between the dyslexia and control group. For the fatigue scale value, dyslexia group mean = 6.93 (*SD* = 1.54) and the control group mean = 6.79 (*SD* = 0.98), $t = 0.29$, $p = .772$). Measurement of fatigue change during the session was not recorded.

Table 1 Self-assessment of programming expertise

	Number of responses			
	Dyslexia group		Control group	
Programming expertise (overall)	low	3	low	2
	medium	8	medium	11
	high	3	high	1
Programming expertise (Java)	low	7	low	6
	medium	6	medium	7
	high	1	high	1

3.3 Study Tasks

The experiment required participants to review three small Java programs and summarize their understanding of these. Each program was preceded by a screen containing brief instructions as to what the participant was required to do. For example, "Program 1 - On the next screen you will see a small Java program. Review this program with a view to understanding its overall function. As you read the program please think-aloud and let us know your thought process. Tell us when you are finished." The instruction screen was followed by a screen displaying the Java source code. This was followed by a screen asking the participant to verbally summarize the program and rate how confident she/he was in their understanding of the program, for example "Reflection – Now tell me about your understanding of this program. How confident are you in your understanding of this program: 10 = High confidence, 1 = low confidence?". This sequence was repeated for each program, so in total nine screens were displayed to each participant. The session also involved completion of a consent form and the profiling questionnaire mentioned above. A typical session lasted approximately 20 min. All sessions were conducted on an individual basis.

A number of factors were considered when choosing the programs to be used in the experiment. Depending on their original degree programme, participants had exposure to a range of programming languages, including Java, JavaScript, Visual Basic and PHP. Professional work experience also affected the language exposure of participants, including C#,

Table 2 Participant descriptions of the most problematical aspect of their dyslexia

Level of dyslexia	Description
Mild	Spelling and reading issues
Mild	Matching names or methods/classes without a predictive prompt
Mild	Reading long pieces of text and spelling
Mild	(none given)
Moderate	It takes me several attempts to read and understand some/most coding
Moderate	Writing, reading at times
Moderate	Reading, I have to re-read
Moderate	Understanding questions. Keeping up in lectures.
Moderate	Spelling and reading
Moderate	Reading, spelling and processing
Moderate	Keeping track of which line/word I am on when reading sets of text.
Severe	Understanding vocal commands, spelling, grammar
Severe	Slow reading and mixing up of words
Severe	Reading, understanding information within a short/time pressured period of time.

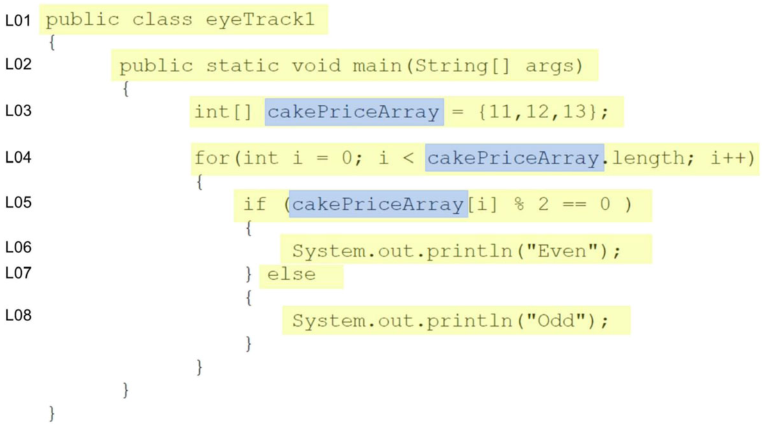
assembly language, and Objective C. For this study, Java was selected as a universal language for the sample.

As the purpose of the study was to examine the reading behaviour of programmers, rather than their programming proficiency, programs of high complexity were not required. Furthermore, it was highly likely that the programming experience of participants would be variable, as recruitment was from the first through to final year of study, with a subset of participants possibly having experience of programming before their studies. Three Java programs from a previous set of studies (Bednarik et al. 2014) were identified as meeting the study requirements. The programs cover a range of programming constructs, while being of sufficient simplicity to fit on a single screen with adequate point size and spacing for readability, and are of sufficient complexity to elicit meaningful gaze activity. In the EMIP'14 studies, the first program was presented as Java pseudocode and has been adapted here to an executable Java class "eyeTrack1", referred to in this paper as Program 1 or P1. EMIP'14 programs two and three have had a minor update, changing their class names to "eyeTrack2" and "eyeTrack3" respectively. In this paper, these are referred to as Program 2 or P2, and Program 3 or P3.

The programs are all beginner level programs, progressing in complexity from simple, through moderate to high complexity at this level. Program 1 iterates over an integer array of cake prices and prints "Even" or "Odd" depending on the current price. Program 2 prompts the user for two numbers and displays the average. Program 3 prints a three-row triangle pattern of asterisks using an inner and outer loop. The programs include features which we might expect to expose differences in gaze behaviour between the two groups, such as identifier names of varying length and a mix of sequence, selection and iteration constructs. The programs are shown in Figs. 3, 4 and 5. (Lines numbers are for reference only and were not in the original source code presented to participants. Similarly, highlighted areas of interest (AOIs) were not visible to the participants but are included here for subsequent discussion).

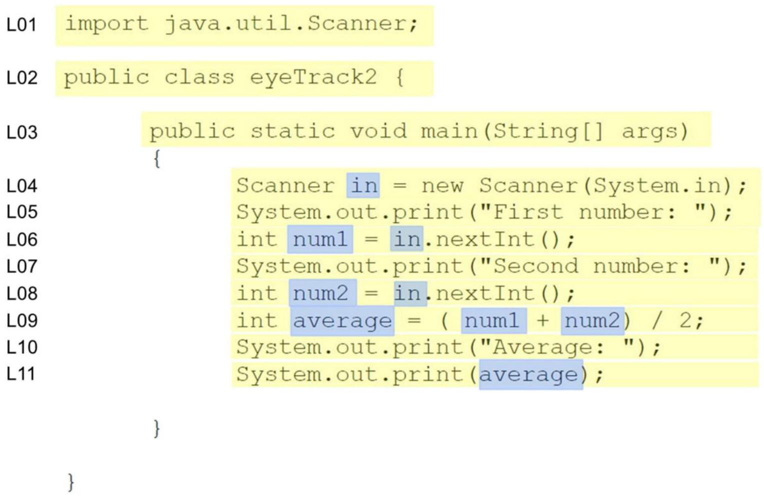
3.4 Instrumentation

Eye gaze recordings were taken using a Tobii X60 Eye Tracker, operating at a data rate of 60 Hz, with a typical accuracy of 0.5 degrees and typical spatial resolution of 0.2 degrees. The eye tracker was connected to a Windows Laptop and a Dell P2210 56 cm (22″) Flat Panel



```
L01 public class eyeTrack1
    {
L02         public static void main(String[] args)
            {
L03                 int[] cakePriceArray = {11,12,13};

L04                 for(int i = 0; i < cakePriceArray.length; i++)
                    {
L05                     if (cakePriceArray[i] % 2 == 0 )
                        {
L06                         System.out.println("Even");
L07                     } else
                        {
L08                         System.out.println("Odd");
                        }
                    }
            }
    }
```

Fig. 3 Program 1 (P1)

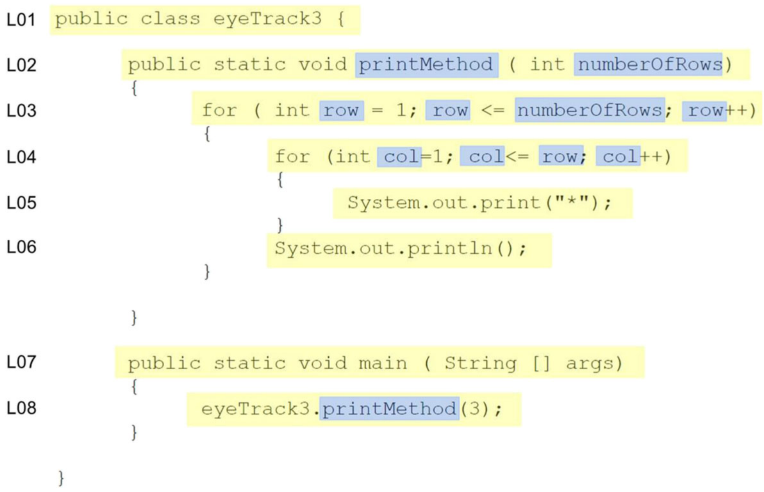


```
L01  import java.util.Scanner;

L02  public class eyeTrack2 {

L03          public static void main(String[] args)
             {
L04                  Scanner in = new Scanner(System.in);
L05                  System.out.print("First number: ");
L06                  int num1 = in.nextInt();
L07                  System.out.print("Second number: ");
L08                  int num2 = in.nextInt();
L09                  int average = ( num1 + num2) / 2;
L10                  System.out.print("Average: ");
L11                  System.out.print(average);

             }

     }
```

Fig. 4 Program 2 (P2)

Monitor operating at a resolution of 1280 × 800 at 60 Hz (Dell native resolution 1680 × 1050 at 60 Hz). The font size at which text and program code was displayed was 5/16 in. (22.5 point size). Participants were seated in front of the Tobii recorder at a distance of approximately 50 cm, with the distance from the Tobii lens to eye approximately 70 cm. For each participant, calibration was performed prior to formal recording. Calibration was not sufficiently accurate in one case and this participant's recording was not used. The audio was also recorded during the session, allowing any think-aloud comments made by the participants to be captured, along with their description and self-assessment of each program's purpose.

Tobii Studio software (version 3.4.5) was used to manage the recordings and to generate eye gaze metrics. The metrics used are defined in Table 3. The Tobii Fixation Filter was used in which a fixation is defined as an eye movement below the velocity threshold of 0.42 pixels/ms. A movement above this velocity is defined as a saccade.



```
L01  public class eyeTrack3 {

L02          public static void printMethod ( int numberOfRows)
             {
L03                  for ( int row = 1; row <= numberOfRows; row++)
                     {
L04                          for (int col=1; col<= row; col++)
                             {
L05                                  System.out.print("*");
                             }
L06                          System.out.println();
                     }

             }

L07          public static void main ( String [] args)
             {
L08                  eyeTrack3.printMethod(3);
             }

     }
```

Fig. 5 Program 3 (P3)

Table 3 Definition of eye gaze metrics from Tobii Studio (durations in seconds)

Metric	Abbreviation	Definition
Fixation count	FC	Number of times the participant fixates on the AOI.
Fixation duration	FD	Duration of an individual fixation within the AOI.
Total fixation duration	TFD	Sum of the duration for all participant's fixations within the AOI.
Fixations before	FB	The number of times participant fixates on the screen before fixating on the AOI for the first time.
Time to first fixation	TFF	Time taken for participant to fixate on the AOI for the first time.
First fixation duration	FFD	Duration of the first fixation on the AOI.
Visit count	VC	Number of visits within the AOI. (A visit is defined as the time interval between the first fixation on the AOI and the end of the last fixation within the same AOI where there have been no fixations outside the AOI.)
Visit duration	VD	Duration of each individual visit within an AOI.
Total visit duration	TVD	Duration of all visits within the AOI.

3.5 Areas of Interest for Gaze Analysis

Results from studies on dyslexia-related eye movements are typically reported at the word, pseudoword or sentence level. When framing the hypotheses, the notion of a "program feature" was adopted as the level of abstraction. This allows the data to be analyzed at various levels of detail, depending on the particular program feature of interest in the comprehension task. In terms of the experimental design, the feature of interest determines how AOIs should be defined within the program code screens. For this study, the AOIs specified were as follows.

3.5.1 Line of code

A line of code is an accepted level of abstraction when dealing with computer programs. High-level languages such as Java are based on a syntax in which one program statement is typically associated with one line of code. Despite the difference between reading text and program code, a line of code is proposed as a reasonable correspondence to a sentence in natural language. Braces are an important syntactical feature of Java programs. The coding convention in Java (Oracle 1997) is that there should be one statement per line and that braces delineating compound statements should typically be on a line of their own. As such, when referring to lines in the programs, only lines consisting of statements were defined as an AOI, disregarding lines containing braces only.

3.5.2 Identifier

If a line of code is the notional correspondence to a natural language sentence, then the program operators and operands which make up a statement have a notional correspondence to words in natural language. For Java programs, features such as keywords, mathematical and logical tokens are regarded as operators. Identifiers and constants are regarded as operands. Keywords and other operators are valid candidates for study. However, when reading program code, identifiers are arguably more aligned with the notion of words in natural text. Selecting AOIs at this level of abstraction is further supported by Binkley et al. (2013), where identifier style was studied as a feature in program comprehension, with factors such as identifier length and style considered influential. As such, identifiers are the second type of AOI in this study. Consideration of keywords and other operators as AOIs is postponed for further work.

3.5.3 Left-right split

The literature on eye gaze in reading identifies left-right asymmetry as a feature of dyslexic reading behaviour (see section 2.5 above). For the above program features - lines of code and identifiers - each is also considered in terms of its left-right split. As such, each line and identifier has an associated left hand side AOI and right hand side AOI (for example, see Fig. 6).

Finally, for summary purposes, the whole program is also defined as an AOI. Thus, the total number of areas of interest investigated was 139, as shown in Table 4. The full set of AOIs is shown in Table 5.

4 Analysis and Results

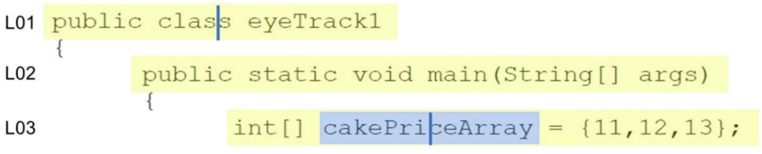
The hypotheses posit that the eye gaze behaviour of programmers with dyslexia exhibits performance deficiencies such that reading gaze is of longer duration or has more fixations. Hypotheses, therefore, have been tested using a one-tailed independent samples t-test for equality of means at a significance level *p* of 0.05. In performing the t-test analysis, inequality of variances in the sample values was accounted for using Levene's Test. Where equality of variance could not be assumed for the t-test, or where AOI recording samples were missing for some participants, the adjusted degrees of freedom is reported as *t* (df). When presenting significant differences in gaze metrics, effect size is reported using median difference as an intuitive indicator and Cohen's *d* metric as a standardized effect size.

4.1 Reading Time and Performance Overview

The time (seconds) spent reading each introduction screen and the subsequent program screen is shown in Table 6. The timings for the screen requesting a confidence rating in code understanding are not reported as that screen signaled a reflection and verbalization task rather than a reading task.

The "P3 introduction" reading time was the only value in which there was a significant difference between the dyslexia group and the control group, $t(17) = 2.49$, $p = .024$, median difference = 1.44, $d = 0.94$.

On completion of the reading task for each program, participants were asked to describe the purpose of the program. They were also asked to rate, on a scale of 1–10, where 1 represented low confidence and 10 represented high confidence, how confident they felt in their understanding of the code. After the experiment, the researchers graded each participant's



```
public class eyeTrack1
{
        public static void main(String[] args)
        {
                int[] cakePriceArray = {11,12,13};
```

Fig. 6 Left-right sub-areas for an AOI

Table 4 Summary of AOIs

AOI count	P1	P2	P3
Whole Program	1	1	1
Lines of code	8	11	8
Identifiers	3	9	11
Left-Right splits	24	36*	26**
Total	36	57	46

*because of its length of 2 characters, the identifier in was not left-right split

**because of their length of 3 characters each, identifiers col and row were not left-right split

comprehension on a scale of 1–10, where 1 represented a very poor understanding of the program's purpose and 10 represented a full understanding. The grading was performed together by the authors with discussion where necessary to reach a consensus value. The comprehension scores are shown in Table 7. There was no significant difference between the performance of the two groups, eliminating a potential confounding parameter concerning comprehension performance across groups.

The comprehension confidence levels are shown in Table 8. Again, there was no significant difference between the groups.

Table 5 Full set of AOIs used in data analysis

AOI count	P1	P2	P3
Whole Program	1	1	1
Lines of code	L01-L08	L01-L11	L01-L08
Identifiers	cakePriceArray-L03 cakePriceArray-L04 cakePriceArray-L05	average-L09 average-L11 in-L04 in-L06 in-L08 num1-L06 num1-L09 num2-L08 num2-L09	col-1-L04 col-2-L04 col-3-L04 numberOfRows-L02 numberOfRows-L03 printMethod-L02 printMethod-L08 row-1-L03 row-2-L03 row-3-L03 row-L04
Left-Right splits	Whole Program – L Whole Program – R L01-L, L01-R, ... L08-L, L08-R cakePriceArray-L03-L cakePriceArray-L03-R cakePriceArray-L04-L cakePriceArray-L04-R cakePriceArray-L05-L cakePriceArray-L05-R	Whole Program – L Whole Program – R L01-L, L01-R, ... L11-L, L11-R average-L09-L average-L09-R average-L11-L average-L11-R num1-L06-L num1-L06-R num1-L09-L num1-L09-R num2-L08-L num2-L08-R num2-L09-L num2-L09-R	Whole Program-L Whole Program-R L01-L, L01-R, ... L08-L, L08-R numberOfRows-L02-L numberOfRows-L02-R numberOfRows-L03-L numberOfRows-L03-R printMethod-L02-L printMethod-L02-R printMethod-L08-L printMethod-L08-R
Total	36	57	46

Table 6 Reading times for each screen

Reading times (s)	Dyslexia mean	Dyslexia *SD*	Control mean	Control *SD*	*t*(26)	p
P1 introduction	19.13	10.33	17.47	6.89	0.50	.620
P1 code	56.78	19.69	70.11	41.45	−1.09	.287
P2 introduction	32.62	15.19	26.01	10.94	1.32	.198
P2 code	71.62	36.33	59.63	26.15	1.00	.325
P3 introduction	14.74	5.68	10.70	2.20	2.49(17)	.024
P3 code	90.71	35.07	91.73	31.38	−0.08	.936

4.2 Hypothesis Testing

This section describes the operational definitions of the hypotheses. For each hypothesis, each AOI category is investigated – program, line of code, identifier and left-right split – using the eye gaze metric appropriate to the hypothesis under investigation.

> $H1_1$ - Programmers with dyslexia have a greater number of fixations on program code features than the control group

The fixation count metric was used to test this hypothesis. At the overall program level, the dyslexia group mean fixation count for Program 1 was 194.1 (*SD* = 68.1), for Program 2 was 227.5 (*SD* = 120.6) and for Program 3 was 304.7 (*SD* = 114.8). The control group mean fixation count for Program 1 was 206.5 (*SD* = 138.7), for Program 2 was 169.4 (*SD* = 72.8) and for Program 3 was 265.6 (*SD* = 108.6). The dyslexia group mean was lower for Program 1 but higher for Program 2 and Program 3. None of these differences were statistically significant.

Analysis was conducted for each AOI in each program. AOIs where the dyslexia group exhibited a significantly larger number of fixations than the control group are shown in Table 9. Only Program 2 contained AOIs where this was the case.

There were also three AOIs for which the difference in mean fixation count was significant but in the other direction from that predicted by the hypothesis, i.e. where the dyslexia group mean was less than the control group mean (see Table 10). For all other AOIs, there was no significant difference in mean fixation count between the two groups in either direction.

> $H2_1$ - Programmers with dyslexia have fixations on program code features with greater duration than the control group

This hypothesis was tested using the fixation duration (seconds) metric. At the program level, the dyslexia group mean fixation duration for Program 1 was 0.24 (*SD* = 0.04), for Program 2 was 0.23 (*SD* = 0.03) and for Program 3 was 0.22 (*SD* = 0.04). For the control group, mean

Table 7 Participant program comprehension scores

Performance (1–10)	Dyslexia mean	Dyslexia *SD*	Control mean	Control *SD*	*t*(26)	p
P1	7.71	3.00	8.21	2.64	−0.47	.643
P2	8.29	2.09	8.50	2.44	−0.25	.805
P3	4.50	2.24	4.41	1.70	0.47	.639

Table 8 Participant program comprehension confidence levels

Confidence (1–10)	Dyslexia mean	Dyslexia *SD*	Control mean	Control *SD*	*t*(26)	p
P1	7.79	2.36	7.57	2.34	0.24	.811
P2	7.43	2.34	7.86	2.51	−0.47	.644
P3	5.71	2.84	5.36	2.31	0.37	.718

fixation duration for Program 1 was 0.25 (*SD* = 0.08), for Program 2 was 0.22 (*SD* = 0.07) and for Program 3 was 0.23 (*SD* = 0.08). The dyslexia group fixation duration was lower for Program 1 and Program 3 and higher for Program 2. None of the differences at the program level were significant.

Again, analysis was conducted for each AOI in each program. Table 11 shows those AOIs where the dyslexia group exhibited a significantly longer fixation duration than the control group. There were no AOIs in Program 3 where this was the case.

As with hypothesis 1, there were AOIs for which the difference in fixation duration was significant but in the other direction from that expected (see Table 12).

> $H3_1$ - Programmers with dyslexia have more gaze visits to program code features than the control group

The visit count metric was used to test this hypothesis. A visit count greater than one indicates that the participant, having fixated on an AOI and saccaded to another program feature, has revisited the original AOI. Though not a meaningful measurement at the program level, it is the case that it was greater than one for some participants, indicating a looking away from the whole program screen and then returning. For the dyslexia group, the mean visit count for Program 1 was 2.29 (*SD* = 2.13), for Program 2 was 1.88 (*SD* = 0.95) and for Program 3 was 2.21 (*SD* = 1.76). For the control group the mean visit count for Program 1 was 2.14 (SD = 1.79), for Program 2 was 2.79 (*SD* = 3.64) and for Program 3 was 2.5 (*SD* = 2.18). There were no significant differences observed.

Those AOIs for which the dyslexia group did show a significantly larger number of visits are shown in Table 13. There were no AOIs from Program 1 where this was the case.

The AOIs for which the difference in visit count was significant but in the other direction from that suggested by the hypothesis were, in Program 1, cakePriceArray-L04-L (dyslexia mean = 1.43, *SD* = 1.95; control mean = 3.50; *SD* = 4.00; *t*(19) = −1.75, *p* = .049, median difference = −2, *d* = −0.66) and, in Program 3, numberOfRows-L03-R (dyslexia mean = 1.64, *SD* = 1.74; control mean = 3.86, *SD* = 4.17; *t*(26) = −1.84, *p* = .039, median difference = −1, *d* = −0.69).

Table 9 Mean fixation count, AOIs for which the dyslexia group value is significantly greater than the control group

Program	AOI	Dyslexia mean	Dyslexia *SD*	Control mean	Control *SD*	*t*(26)	p	median diff.	d
2	L06	26.29	16.04	16.64	9.29	1.95	.031	10	0.74
2	in-L06	2.29	1.33	1.29	1.20	2.09	.023	1	0.79
2	average-L09-R	2.07	2.02	1.00	0.78	1.85	.038	1	0.70
2	num1-L09-R	2.29	1.54	1.21	1.42	1.91	.034	1	0.72

Table 10 Mean fixation count, AOIs for which dyslexia group is significantly less than the control group

Program	AOI	Dyslexia mean	Dyslexia *SD*	Control mean	Control *SD*	*t*(26)	p	median diff.	d
1	cakePriceArray-L04-L	1.64	2.37	3.86	4.26	−1.71	.050	−2.50	−0.64
3	numberOfRows-L03-L	2.71	3.10	5.93	5.44	−1.92(21)	.034	−3.5	−0.73
3	numberOfRows-L03-R	1.71	1.77	4.50	4.55	−2.13(17)	.024	−1.5	−0.81

Table 11 Mean fixation duration (s), AOIs for dyslexia group mean significantly greater than the control group

Program	AOI	Dyslexia mean	Dyslexia *SD*	Control mean	Control *SD*	*t*(26)	p	median diff.	d
1	L07-R	0.26	0.07	0.18	0.09	1.80	.049	0.06	0.99
2	L04	0.22	0.06	0.17	0.07	1.79(23)	.043	0.05	0.71
2	L06-R	0.23	0.06	0.19	0.07	1.80	.042	0.07	0.68
2	L07-R	0.25	0.07	0.20	0.04	2.25(25)	.017	0.05	0.88

> $H4_1$ – For programmers with dyslexia, the number of program code features with a gaze visit count of [1|0] is less than the control group

This hypothesis relates to the ability to scan or skip words when reading, with the expectation that programmers with dyslexia will exhibit a smaller number of words that receive either a single fixation or are skipped. This hypothesis can be tested by identifying the number of AOIs with a visit count = 0, indicating that the AOI has been skipped, and with a visit count = 1, indicating that the AOI is viewed only once with no regressions.

First, consider the AOIs which were read with no regressions. The overall number of AOIs where visit count = 1 was 53 for the dyslexia group and 59 for the control group. Comparing this number between the groups, if the null hypothesis were true, then we would expect the number of AOIs with visit count = 1 to be the same in each group. The observed numbers are shown in Table 14. These values were analyzed using an independent samples Mann-Whitney U test. The total for Program 1, Program 2 and Program 3 combined shows that the dyslexic group “scanned” a smaller number of AOIs, however this difference is not statistically significant ($U = 92.5$, $p = .399$).

For Program 1 and Program 3, the dyslexia group has more AOIs with a visit count of one, and for Program 2 the dyslexia group has fewer AOIs with a visit count of one. In none of these cases is the difference statistically significant (for Program 1, $U = 104.5$, $p = .365$; Program 2, $U = 73.5$, $p = .124$; Program 3, $U = 103.0$, $p = .407$).

Second, consider AOIs which were not read. The overall number of AOIs where visit count = 0 was 74 for the dyslexia group and 75 for the control group (see Table 15). The data shows that the dyslexia group “skipped” only one less AOI than the control group – not statistically significant ($U = 100$, $p = .463$). For Program 1 there was no numerical difference between the groups, for Program 2 the dyslexia group had fewer skipped AOIs and for Program 3 the dyslexia group had more. For Program 2 and Program 3 neither of the differences were statistically significant (for Program 2, $U = 73.5$, $p = .120$; Program 3, $U = 111.5$, $p = .265$). In summary for hypothesis 4, there is no significant difference in the number of AOIs which receive either a single visit or are skipped.

> $H5_1$ – The correlation between identifier length and fixation duration is stronger for programmers with dyslexia than the control group

When reading normal text, it is observed that dyslexic readers spend more gaze time on longer words than non-dyslexic readers and that there is a stronger correlation between word length and fixation duration for dyslexic readers. In regard to program code, identifier length is the number of characters in the identifier name. In the three Java programs used, identifier length ranged from 2 (identifier in in Program 2) through to 14 (identifier cakePriceArray

Table 12 Mean fixation duration (s), AOIs for which dyslexia group significantly less than the control group

Program	AOI	Dyslexia mean	Dyslexia *SD*	Control mean	Control *SD*	*t*(26)	p	median diff.	d
3	L08	0.16	0.04	0.20	0.08	−1.77(19)	.046	−0.03	−0.68
3	numberOfRows-L03	0.18	0.04	0.21	0.05	−1.77(20)	.046	−0.03	−0.75
3	numberOfRows-L02-L	0.20	0.05	0.27	0.10	−2.07	.026	−0.04	−0.88
3	printMethod-L08-R	0.15	0.05	0.26	0.16	−1.99(11)	.036	−0.08	−0.93

Table 13 Mean visit count, AOIs for which the dyslexia value significantly greater than the control group

Program	AOI	Dyslexia mean	Dyslexia *SD*	Control mean	Control *SD*	*t*(26)	p	median diff.	d
2	L10	10.29	8.29	6.14	3.16	1.75	.046	4.5	0.66
2	in-L06	2.29	1.33	1.29	1.20	2.09	.023	1	0.79
2	average-L09-R	2.07	2.02	1.00	0.78	1.85	.038	1	0.70
2	L03-L	2.93	2.84	1.43	0.76	1.91(15)	.038	1	0.72
2	L06-R	8.57	4.93	5.00	3.55	2.20(24)	.018	4.5	0.83
2	L09-R	7.14	5.14	3.79	2.26	2.24(18)	.019	3	0.84
2	L10-R	9.36	7.68	5.07	2.90	1.95	.031	2	0.74
2	num1-L09-R	2.21	1.53	1.14	1.29	2.00	.028	1	0.76
3	L02-L	9.86	8.66	5.36	3.15	1.83	.040	3.5	0.69
3	L02	13.07	9.98	7.64	3.05	1.95	.031	2.5	0.74

in Program 1). For each identifier instance, the mean fixation duration was calculated across all participants and correlated with identifier length. For the dyslexia group, the Pearson Correlation between identifier length and mean fixation duration was $r = 0.083$ ($p = .354$) and for the control group $r = -0.342$ ($p = .055$). The correlation is weak in both groups, slightly positive for the dyslexia group and more strongly negative for the control group.

To test the hypothesis, it is necessary to test whether the difference between these correlations is significant. A univariate analysis of variance was conducted. Data was prepared for the mean fixation duration for each identifier AOI per subject. (46 cases in total, 23 identifier instances and their mean fixation duration for the dyslexia group and similarly for the control group). Using the method described in (Howell 2012), the identifier length and mean fixation duration were standardized to Z scores so that tests on the effects of identifier length will be on within-group correlations, without distortion from differences in the variance of mean fixation duration across the groups.

Using the univariate analysis of variance (at 0.1 confidence level) with mean fixation duration as the dependent variable, the between groups effect of identifier length is not significant, $F(1, 42) = 2.019$, $p = .163$, rejecting the alternate hypothesis.

> $H6_1$ - Programmers with dyslexia exhibit a greater number of fixations on the RHS of a program code feature than the control group

Hypotheses 6 and 7 are derived from dyslexia research relating to asymmetrical allocation of attention in the visual field, typically associated with the linear reading of natural text at the word level. When considered in terms of program code features, this is most relevant to gaze behaviour when reading identifiers. Hypothesis testing was extended to include the larger program features of lines of code and the whole program, to investigate if any enhanced

Table 14 Number of AOIs with visit count = 1

Number of AOIs	P1	P2	P3	Total
Dyslexia group	7	19	27	53
Control group	6	27	26	59

Table 15 Number of AOIs with Visit Count = 0

Number of AOIs	P1	P2	P3	Total
Dyslexia group	5	23	46	74
Control group	5	32	38	75

parafoveal acuity of a programmer with dyslexia leads to different gaze patterns. At the line level, left-hand side/right-hand side distinction is important because OVP in natural text dyslexia is based on the notion of a linear left-to-right reading model. When reading code, given there is a mixture of linear and non-linear reading patterns, we are interested if there is a tendency to land on the right-hand side or left-hand side of a line. Hypothesis testing for the whole program follows from the tendency for (some) programs to be right-hand side dominant, due to indentation, and whether this might lead to differences in how a programmer with dyslexia reads the code.

The fixation count metric was used as an indicator of visual attention. As described in section 3.5, each AOI, including the whole program, is evenly split such that there is a corresponding left AOI and right AOI. For example, the AOI L01 in Program 1 has two associated AOIs, L01-L and L01-R. For an AOI, the number of fixations on its right-hand side is expressed as a percentage of the number of fixations on the full AOI.

First, considering the whole program as an AOI, separating the left-right fixation counts led to the data shown in Table 16.

At the program level, the dyslexia group gaze was effectively symmetrical for Program 1, and showed a slight RHS bias for Program 2 and Program 3. The control group showed a slight RHS bias for each program, higher than the dyslexia group for Program 1 and Program 3 and lower for Program 2. None of these differences were significant. The results are not surprising as the spatial structure of the programs, due to indentation, is towards the right-hand side of the program.

Table 17 shows those AOIs where the dyslexia group fixation count for the RHS of an AOI was greater than 50% of the full AOI and compares the dyslexia group percentage with the control group.

For this subset of AOIs, there is only one instance where the dyslexia RHS% is significantly greater than the control group (Program 1, cakePriceArray-L04, $d = 1.02$), most likely due to the length property immediately following. There are four AOIs in Table 17 where the dyslexia group RHS% is lower than that of the control group (Program 1, L08; Program 2, L01; Program 3, L01; and Program 3, L08) but none of these differences were significant. The data suggest a rejection of the alternate hypothesis.

> $H7_1$ - Programmers with dyslexia exhibit a greater fixation duration on the RHS of a program feature than the control group

Table 16 Percentage mean fixation count on RHS of whole program

Program	AOI	Dyslexia mean RHS%	Dyslexia *SD*	Control mean RHS%	Control *SD*	*t*(26)	p
1	P1-R	50.8	8.3	54.4	7.9	−1.19	.494
2	P2-R	57.9	10.5	55.7	12.38	0.52	.305
3	P3-R	56.8	8.2	57.1	9.6	−0.08	.471

Table 17 Mean fixation count percentage for AOIs where dyslexia group values greater than 50%

Program	AOI	Dyslexia mean RHS%	Dyslexia *SD*	Control mean RHS%	Control *SD*	*t*(26)	p
1	cakePriceArray-L04	67.2	36.5	36.8	21.2	2.44(15)	.014
1	cakePriceArray-L05	50.4	34.8	34.5	29.1	1.29	.105
1	L06	51.0	20.1	39.6	32.9	1.07(20)	.149
1	L07	90.0	19.2	82.5	32.2	0.52	.307
1	L08	50.9	32.3	60.5	31.2	−0.75	.230
2	average-L11	50.9	28.4	44.00	28.5	0.63	.268
2	L01	56.9	29.9	58.1	26.5	−0.10	.459
2	L08	53.0	13.7	52.7	23.21	0.04	.483
2	L10	63.7	18.2	58.0	17.9	0.83	.208
2	L11	71.9	12.1	65.6	10.7	1.44	.081
2	num1-L09	58.0	31.6	37.6	34.6	1.60	.061
3	L01	64.2	36.4	74.7	31.2	−0.77	.226
3	L06	71.1	18.9	67.1	22.2	0.51	.309
3	L08	53.7	19.5	62.6	24.3	−1.04	.154
3	numberOfRows-L03	51.1	34.2	40.4	12.1	1.02(14)	.163
3	printMethod-L08	62.4	30.9	59.7	33.9	0.20	.423

In this case visual attention is considered in terms of total fixation duration. Considering the whole program as an AOI, separating the left-right total fixation duration led to the data shown in Table 18.

The pattern of RHS bias is similar to that indicated by fixation count – greater than 50% for each of the two groups on each program, with the dyslexia group showing a RHS percentage greater than the control group only in Program 2, and vice versa in Program 1 and Program 3. None of the differences were significant.

AOIs where the dyslexia group total fixation duration on the right-hand side was greater that 50% are shown in Table 19.

The only AOI where the dyslexia RHS% is significantly larger than the control group is again Program 1, cakePriceArray-L04 ($d = 0.93$). In Table 19 AOIs for which the dyslexia group RHS% is lower than that of the control group are, for Program 2, L01, and L08. For Program 3, this is the case for L01, L08 and printMethod-L08. None of these differences were significant. Overall these figures suggest a rejection of the alternate hypothesis.

4.3 Exploratory Data Analysis

In addition to the selected gaze metrics used for hypothesis testing, a range of other gaze metrics was analyzed to identify AOIs where there were significant differences in gaze activity between the groups. In testing these differences, the independent samples t-test for equality of means was again used, at a significance level of 0.05. A one-tail test was used for those gaze

Table 18 Percentage mean total fixation duration on RHS of whole program

Program	AOI	Dyslexia mean RHS%	Dyslexia *SD*	Control mean RHS%	Control *SD*	*t*(26)	p
1	P1	52.2	10.8	57.7	7.8	−1.52	.070
2	P2	58.5	12.6	56.8	13.9	0.33	.372
3	P3	58.7	8.6	59.9	10.0	−0.33	.371

Table 19 Mean total fixation duration for AOIs where dyslexia group values greater than 50%

Program	AOI	Dyslexia mean RHS%	Dyslexia *SD*	Control mean RHS%	Control *SD*	*t*(26)	p
1	cakePriceArray-L04	64.9	39.0	35.1	22.7	2.23(15)	.015
1	L06	56.1	20.2	41.2	35.4	1.33(19)	.100
1	L07	95.3	8.1	82.8	36.5	0.89(11)	.196
2	L01	56.1	31.0	60.9	29.8	−0.39	.351
2	L06	50.1	17.0	49.7	22.6	0.06	.477
2	L08	51.5	13.4	57.1	22.7	−0.78	.222
2	L10	69.3	17.7	61.1	19.4	1.17	.126
2	L11	72.2	16.0	68.7	9.9	0.71	.244
2	num1-L09	54.7	34.8	37.0	35.3	1.32	.100
3	L01	61.9	37.5	74.4	33.3	−0.87(23)	.196
3	L05	75.5	16.4	67.3	20.6	−0.94	.178
3	L08	55.3	20.0	67.1	23.1	−1.42(25)	.085
3	printMethod-L08	61.2	32.2	64.9	34.1	−0.26(21)	.399

metrics concerned with number of fixations on a AOI (e.g., minimum visit count) and fixation duration on an AOI (e.g., minimum visit duration), in keeping with the approach adopted during hypothesis testing. When exploring metrics for which there is no anticipation of the direction of significance, a two-tail test was used. Example metrics in this category include fixations before and time to first fixation. The significant findings for each program are described below. The analysis showed that AOIs already identified as being of interest from hypothesis testing, were similarly of interest when examined using other gaze metrics, which is not surprising. However, in some cases, new AOIs have emerged based on differences in gaze metrics such as fixations before and first fixation duration. These are described for each program in turn.

4.3.1 Program 1

Considering identifiers, for cakePriceArray-L03, the first occurrence of the cakePriceArray identifier, the dyslexia group showed a significantly greater number of fixations before. For the dyslexia group, mean fixations before = 44.8 (*SD* = 37.2), for the control group mean = 20.5 (*SD* = 15.9), *t*(17) = 2.14, *p*(2-tail) = .048, median difference = 18, *d* = 0.85.

For cakePriceArray-L04, the minimum fixation duration metric shows a significant difference. The minimum is the lowest fixation duration (seconds) value in the dataset of fixations for each participant. For the dyslexia group, the mean minimum fixation duration = 0.14 (*SD* = 0.04), and control group mean = 0.10 (*SD* = 0.03), *t*(22) = 2.08, *p* = .025, median difference = 0.05, *d* = 1.13. This can be interpreted as, for all of the shortest glances on this AOI, the control group's were significantly shorter than those of the dyslexia group. Also for cakePriceArray-L04, the dyslexia group mean first fixation duration = 0.20 (*SD* = 0.06), and the control group mean = 0.15 (*SD* = 0.06), *t*(22) = 1.86, *p* = .038, median difference = 0, *d* = 0.83.

Table 20 shows the lines of code in Program 1 which exhibited significant differences in various time-based metrics. For L04, the dyslexia group mean first fixation duration was significantly greater than the control group. On the other hand, L08 showed a dyslexia group mean first fixation duration significantly lower than the control group. For L04, the mean minimum visit duration is significantly greater for the dyslexia group than the control group. For L08, the dyslexia mean visit duration is significantly lower than the control group mean.

Table 20 Lines of code in Program 1 with significant difference in various time-based metrics

AOI	Metric	Dyslexia mean	Dyslexia *SD*	Control mean	Control *SD*	*t*(26)	p	median diff.	d
L04	first fixation duration	0.28	0.21	0.17	0.10	1.79	.042	0.08	0.67
L08	first fixation duration	0.12	0.05	0.18	0.09	2.08(23)	.025	−0.06	−0.83
L04	minimum visit duration	0.15	0.06	0.10	0.03	2.62	.008	0.05	1.05
L08	visit duration	0.20	0.08	0.31	0.19	0.81(17)	.044	−0.05	−0.75
L05	maximum fixation duration	0.64	0.28	1.00	0.68	1.8(17	.045	−0.2	−0.69

Comparing L04 and L08, this might suggest some inversion of gaze behaviour between top and bottom of the program. For L05, the dyslexia group's mean maximum fixation duration was significantly less than the control mean.

4.3.2 Program 2

For Program 2 there were a number of AOIs showing differences in behaviour in both directions. First considering the identifiers, num1-L09 shows a dyslexia group mean first fixation duration = 0.16 (*SD* = 0.07) significantly lower than the control group (*M* = 0.26, *SD* = 0.09), *t*(23) = −2.92, *p* = .004, median difference = −0.08, *d* = −1.24). Also for this AOI, the dyslexia group mean minimum visit duration of 0.13 (*SD* = 0.05) is significantly lower than the control group (*M* = 0.18, *SD* = 0.09), *t*(23) = −1.81, *p* = .042, median difference = −0.05, *d* = −0.69.

For num2-L08, the gaze activity before fixation on the AOI is significantly greater for the dyslexia group. Its mean fixations before = 88.2 (*SD* = 45.8) compares with the control group (*M* = 54.4, *SD* = 22.2), *t*(18) = 2.37, *p*(2 tail) = .029, median difference = 41.5, *d* = 0.94.

Considering the lines in Program 2, the testing of hypothesis 2 showed that, for L04, the dyslexia group had significantly longer mean fixation duration. Other metrics reinforce this for L04 as shown in Table 21, suggesting this line as a significant differentiator in how the two groups read the Program 2 code.

For L06, the dyslexia group also showed significantly larger values for mean total fixation duration (*M* = 5.87, *SD* = 3.17) than the control group (*M* = 3.51, *SD* = 2.21), *t* = 2.29, *p* = .015, median difference = 3.27, *d* = 0.86. Also, considering mean total visit duration, dyslexia group mean = 6.80 (*SD* = 3.56) and control group mean = 4.43 (*SD* = 2.54), *t* = 2.03, *p* = .026, median difference = 2.72, *d* = 0.77. The mean fixation count for the dyslexia grou*p* = 26.29 (*SD* = 16.04), and the control group mean = 16.64 (*SD* = 9.29), *t* = 1.95, *p* = .031, median difference = 10, *d* = 0.74.

Table 21 Other dyslexia group metrics showing longer fixation and visit durations for Program 2-L04

Metric	Dyslexia mean	Dyslexia *SD*	Control mean	Control *SD*	*t*(26)	p	median diff.	d
maximum fixation duration	0.39	0.20	0.24	0.10	2.35(18)	.015	0.07	0.95
mean total fixation duration	2.78	3.68	0.74	0.67	2.05(14)	.030	0.53	0.77
mean first fixation duration	0.27	0.14	0.15	0.09	2.30(23)	.016	0.1	1.02
mean total visit duration	3.38	4.91	0.91	0.85	1.86(14)	.042	0.48	0.70

Noting L07, it is of interest not because the dyslexia group spent more time on it, but they exhibited more fixations elsewhere on the program before fixating on that line; mean fixations before = 56.4 (*SD* = 33.6) compared with control group mean = 32.7 (*SD* = 21.0), *t*(25) = 2.18, *p*(2-tail) = .039, median difference = 13.50, *d* = 0.85. When the dyslexia group did first fixate on this AOI, their initial gaze duration was lower; dyslexia group mean first fixation duration = 0.18 (*SD* = 0.10), control group mean = 0.26 (*SD* = 0.14), *t*(25) = −1.71, *p* = .050, median difference = −0.07, *d* = −0.66.

The dyslexia group also showed a "delay" in fixating on L08, with mean fixations before = 78.1 (*SD* = 40.3), compared with the control group (*M* = 48.7, *SD* = 20.3), *t*(19) = 2.43, *p*(2 tail) = .025, median difference = 31, *d* = 0.92.

For L10, the dyslexia group's mean visit count was larger (*M* = 10.29. *SD* = 8.29) than the control group (*M* = 6.14, *SD* = 3.16), *t* = 1.75, *p* = .046, median difference = 4.5, *d* = 0.66.

4.3.3 Program 3

As noted when testing hypothesis 2, the dyslexia group gaze on numberOfRows-L03 exhibited significantly shorter durations than the control group. This is also apparent from other related metrics as shown in Table 22.

The identifier row-2-L03 also showed, for the dyslexia group, fixation durations and visit durations significantly lower than the control group (see Table 23).

The first fixation duration on row-1-L03 and row-3-L03 were significantly higher for the dyslexia group. Row-1-L03 mean first fixation duration = 0.27 (*SD* = 0.19), control group mean = 0.16 (*SD* = 0.07), *t*(16) = 1.83, *p* = .043, median difference = 0.07, *d* = 0.77. row-3-L03 mean first fixation duration = 0.37 (*SD* = 0.19), control group mean = 0.19 (*SD* = 0.08), *t*(8) = 2.31, *p* = .025, median difference = 0.17, *d* = 1.23.

For row-L04, the dyslexia mean visit duration of 0.19 (*SD* = 0.06) was significantly lower than the control group (*M* = 0.27, *SD* = 0.11), *t*(18) = −1.82, *p* = .043, median difference = −0.09, *d* = −0.90.

The gaze pattern to emerge for col-1-L04 is that for the dyslexia group, it took longer "to get there" and when they did, gaze durations were shorter. (See Table 24).

Regarding Program 3 lines, it was the case that those shown in Table 25 had significantly less gaze time from the dyslexia group in terms of the metrics shown in the table.

For L02, the dyslexia group had mean visit count = 13.07 (*SD* = 9.98), significantly greater than the control mean (*M* = 7.64, *SD* = 3.05), *t*(26) = 1.95, *p* = .031, median difference = 2.5, *d* = 0.74.

L07 had a dyslexia group mean minimum visit duration = 0.23 (*SD* = 0.19) which was significantly greater than the control group (*M* = 0.11, *SD* = 0.04), *t*(14) = 2.37, *p* = .016, median difference = 0.05, *d* = 0.87.

5 Discussion

The significant metrics from hypothesis testing and the exploratory analysis are shown in Figs. 7, 9 and 11. These figures are the focus for subsequent discussion in this section. (In Figs. 7, 9, 11, ">" indicates metrics for which the dyslexia group values were greater than the control group, "<" indicates metrics for which the values were less than the control group). Heat maps are shown in Figs. 8, 10 and 12 as a visual guide to the attention distribution within

Table 22 Program 3 numberOfRows-L03 significant metrics

Metric	Dyslexia mean	Dyslexia *SD*	Control mean	Control *SD*	t	p	median diff.	d
maximum visit duration	0.40	0.27	0.98	0.70	−2.59 (13)	.011	−0.38	−1.09
sum visit duration	1.12	0.90	3.06	2.32	−2.59 (13)	.011	−1.50	−1.10
maximum fixation duration	0.26	0.07	0.41	0.18	−2.56 (13)	.012	−0.12	−1.10
sum fixation duration	0.98	0.62	2.71	2.14	−2.58 (12)	.012	−0.90	−1.10
mean visit duration	0.24	0.09	0.39	0.19	−2.44 (14)	.014	−0.09	−1.01
mean total visit duration	0.88	0.92	2.40	2.42	−2.21 (17)	.021	−1.07	−0.83
mean total fixation duration	0.77	0.69	2.13	2.21	−2.21 (16)	.021	−0.93	−0.83
mean fixation duration	0.18	0.04	0.21	0.05	−1.77 (20)	.046	−0.03	−0.66
mean fixation count	4.50	4.40	9.50	8.73	−1.91 (19)	.035	−6.00	−0.72

Table 23 Program 3 row-2-L03 fixation duration and visit duration values

Metric	Dyslexia mean	Dyslexia *SD*	Control mean	Control *SD*	t	p	median diff.	d
mean total fixation duration	0.29	0.42	0.84	1.04	−1.82 (17)	.043	−0.36	−0.69
sum fixation duration	0.58	0.44	1.30	1.03	−1.89 (11)	.043	−0.68	−0.91
mean total visit duration	0.31	0.45	0.87	1.09	−1.77 (17)	.047	−0.36	−0.67
sum visit duration	0.62	0.47	1.35	1.10	−1.80 (11)	.049	−0.68	−0.86

Table 24 col-1-L04 significant metrics

Metric	Dyslexia mean	Dyslexia *SD*	Control mean	Control *SD*	t	p	median diff.	d
mean fixations before	133.5	77.3	63.5	38.9	2.70 (21)	.013(2 tail)	61	1.14
mean time to first fixation	37.6	20.7	19.8	13.1	2.43 (21)	.024(2 tail)	15.24	1.03
sum visit duration	0.54	0.48	1.32	1.15	−2.08 (13)	.029	−0.64	−0.89
sum fixation duration	0.54	0.48	1.26	1.09	−2.01 (13)	.032	−0.64	−0.85
mean visit duration	0.21	0.11	0.30	0.11	−1.87 (21)	.038	−0.08	−0.82

Table 25 Program 3 lines where metrics indicated significantly less time spent by the dyslexia control group

AOI	Metric	Dyslexia mean	Dyslexia *SD*	Control mean	Control *SD*	*t*(26)	p	median diff.	d
L02	minimum visit duration	0.10	0.03	0.18	0.16	−1.75(26)	.046	−0.05	−0.69
L03	mean visit duration	0.51	0.20	1.10	0.89	−2.44(14)	.014	−0.38	−0.91
L04	minimum fixation duration	0.07	0.04	0.10	0.03	−2.69	.006	−0.03	−0.85
L04	mean visit duration	0.53	0.20	0.87	0.57	−2.15(16)	.023	−0.21	−0.80
L04	maximum visit duration	2.07	1.04	3.72	2.93	−1.99(16)	.032	−0.92	−0.75
L06	maximum visit duration	0.74	0.45	1.30	0.83	−2.21	.018	−0.66	−0.84
L08	mean fixation duration	0.16	0.04	0.20	0.08	−1.77(19)	.046	−0.03	−0.63
L08	maximum fixation duration	0.28	0.15	0.56	0.57	−1.76(15)	.050	−0.02	−0.67

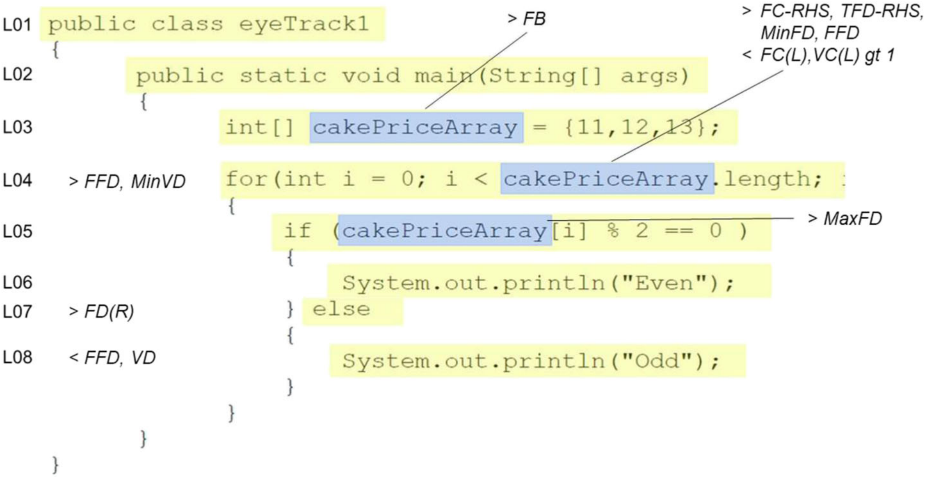
each of the two groups. The heat maps were produced using Tobii Studio, displaying the accumulated fixation durations per group on the program code. Red represents the points of maximum accumulated fixation duration, through yellow, to green for the lowest accumulated fixation duration. At a point radius of 50 pixels, the maximum duration values represented by the red coloring are shown in the respective figure captions.

5.1 Program 1

Figure 7 shows those metrics from Program 1 for which there was a significant difference in gaze behaviour between the groups. Figure 8 shows the heat map (fixation duration) for reference. Referring back to Table 6, the heat map is not suggesting that the dyslexia group spent more time viewing the program, but that their gaze intensity was more evenly spread across the program than was the case for the control group.

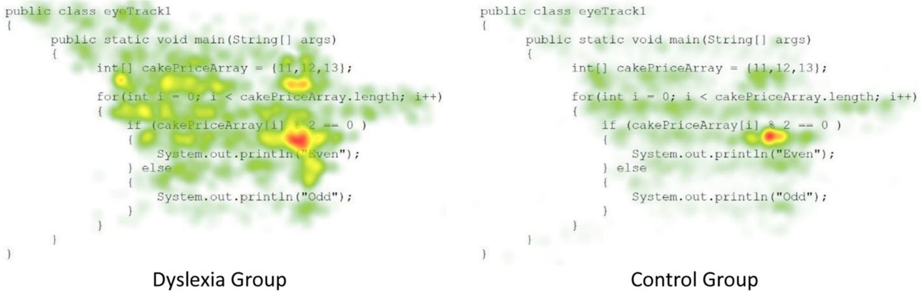
Hypothesis 1 and 2 suggest that programmers with dyslexia will pay more attention to, and spend longer on, program features than those in the control group. However, there were no AOIs in Program 1 were this was the case. One of the AOIs, cakePriceArray-L04-L, actually showed the reverse, with the dyslexia group fixation count significantly lower than the control group. Hypothesis 3 suggests that the dyslexia group would rescan a program feature to a greater extent than the control group, and again the reverse effect was observed here for cakePriceArray-L04-L, with the dyslexia visit count lower than the control.

Line 04 and cakePriceArray-L04 show other distinct differences in gaze activity between the two groups. Line 04 could be regarded as one of the more complex lines in the program – it is the header of the for loop which iterates over the cake price array and it has the largest number of characters (46). Is it the case that the relative complexity of Line 04 causes the programmer with dyslexia to actually "avoid" its details, rather than dwell longer on the line as suggested by hypothesis 1–3? Literature has already been noted which would suggest that "crowding" can lead to a deficiency in the visual attention of dyslexic readers. The exact mechanism at work in the crowding phenomenon is not clear. For any type of reader, visual feature analysis can be impaired by distractors masking the reading target. This distraction appears to be enhanced for dyslexic readers (Bellocchi 2013). Adjustment to word and letter



```
L01   public class eyeTrack1
       {
L02            public static void main(String[] args)
                {
L03                  int[] cakePriceArray = {11,12,13};

L04                  for(int i = 0; i < cakePriceArray.length;
                      {
L05                        if (cakePriceArray[i] % 2 == 0 )
                            {
L06                             System.out.println("Even");
L07                         } else
                            {
L08                             System.out.println("Odd");
                            }
                      }
                }
       }
```

Fig. 7 Significant gaze metrics for Program 1



```
public class eyeTrack1
{
        public static void main(String[] args)
        {
                int[] cakePriceArray = {11,12,13};

                for(int i = 0; i < cakePriceArray.length; i++)
                {
                        if (cakePriceArray[i] % 2 == 0 )
                        {
                                System.out.println("Even");
                        } else
                        {
                                System.out.println("Odd");
                        }
                }
        }
}
```

Dyslexia Group

```
public class eyeTrack1
{
        public static void main(String[] args)
        {
                int[] cakePriceArray = {11,12,13};

                for(int i = 0; i < cakePriceArray.length; i++)
                {
                        if (cakePriceArray[i] % 2 == 0 )
                        {
                                System.out.println("Even");
                        } else
                        {
                                System.out.println("Odd");
                        }
                }
        }
}
```

Control Group

Fig. 8 Heatmap (fixation duration) for Program 1. Red represents maximum duration values, dyslexia group = 10.44 s, control group = 28.89 s

spacing can improve reading performance. However, the multi-layered nature of dyslexia is such that crowding can impact different subgroups in different ways. Nevertheless, as a broad concept, it may help explain some of the observed differences for the program.

For the dyslexia group, line 04 and cakePriceArray-L04 have significantly larger first fixation durations. This could suggest that the complexity of this line requires more initial attention from the programmer with dyslexia, and having encountered this complexity, leads to subsequent inattention. The dyslexia group fixations before metric on the cakePriceArray-L03 is significantly larger than the control group, also supporting the observation that initial attention in the dyslexia group is drawn to the area of Line 04. The fixation count and total fixation duration for cakePriceArray-L04-R are greater for the dyslexia group (from testing hypothesis 6 and 7 respectively), consistent with a mini-left neglect phenomenon on this AOI.

When fixating on cakePriceArray-L05, the dyslexia group's longest fixations are significantly longer than those of the control group. As can be seen from the heat map, there is an overall high level of gaze activity around the modulus operator adjacent to this identifier, especially for the dyslexia group, which could be an explanation for this metric.

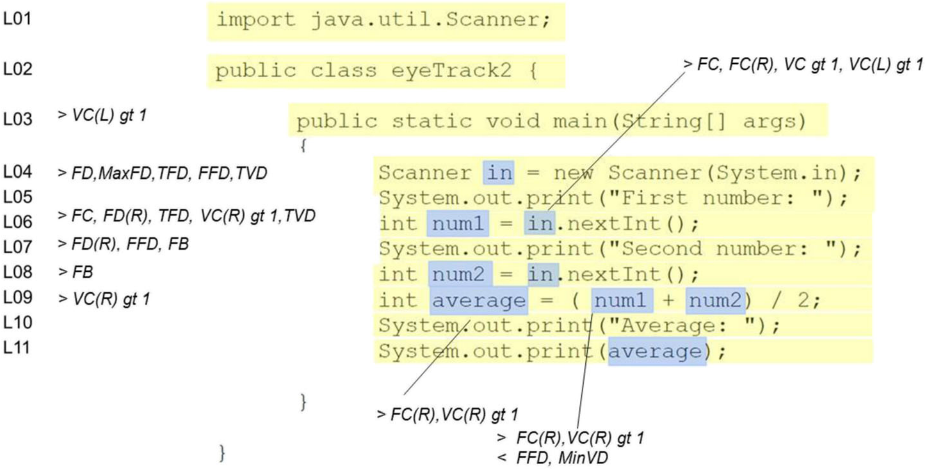
The other observation of note is the dyslexia group's significantly lower visit duration and first fixation duration for the last line of the program, suggesting a relative inattention to this part of the code.

In summary, it would appear that Line 04 is a differentiator between the two groups. Line 04 is also the most complex line in the program, both in terms of its role within the algorithm and also in terms of character length. Hypothesis 5 does suggest that the longer the program feature, the longer the dyslexia group's fixation duration. As has been shown, for all of the programs, this correlation is weak and is not significantly different from that of the control group. The difference in gaze performance may not arise from the length of the line as such, but possibly because its length and its location relative to other lines of the program (i.e. in the middle), serves to "crowd" or "hide" other important program features, the comprehension of which, is necessary for the overall comprehension of the program.

5.2 Program 2

Figure 9 shows those metrics from Program 2 for which there was a significant difference between the groups. Figure 10 shows the heat map (fixation duration) for reference, indicating that within both groups there is a similar distribution of attention across the AOIs.

The two groups show distinct behaviour in relation to a number of features in Program 2. In testing hypothesis 1, line 06 and the identifier in-L06 receive significantly more attention as

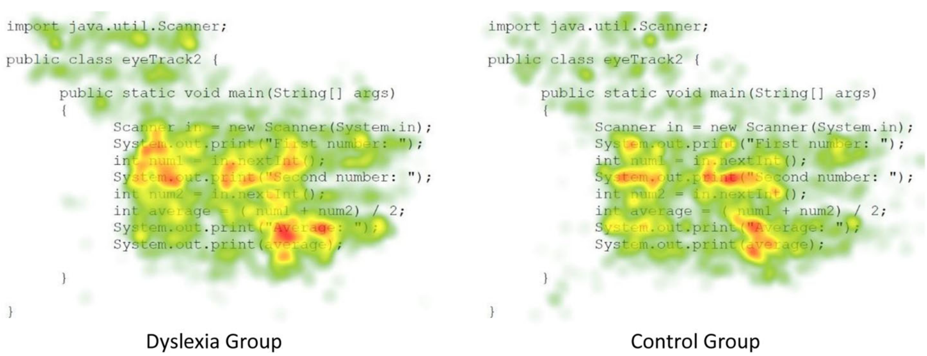


```
L01          import java.util.Scanner;
L02          public class eyeTrack2 {
L03              public static void main(String[] args)
                 {
L04                  Scanner in = new Scanner(System.in);
L05                  System.out.print("First number: ");
L06                  int num1 = in.nextInt();
L07                  System.out.print("Second number: ");
L08                  int num2 = in.nextInt();
L09                  int average = ( num1 + num2) / 2;
L10                  System.out.print("Average: ");
L11                  System.out.print(average);
                 }
             }
```

Fig. 9 Significant gaze metrics for Program 2

measured by fixation count. When considering fixation duration (hypothesis 2), significantly more time is spent on line 04, as well as the right-hand side AOIs of lines 06 and 07. Unlike Program 1, there were no AOIs for which the reverse phenomenon (i.e. significantly less attention) was observed.

Other metrics of significance for line 06 are that the dyslexia group showed greater total fixation duration and total visit duration, along with evidence of more regressions on the right hand side of the line. Within the main code section, line 06 and line 08 are the shortest statements (24 characters each), yet are associated with quite different gaze behaviors from the two groups. Line 08, for example, does not show any of the differences of line 06 regarding fixation counts and duration. Lines 07 and 08, however, seem to be an area of the program for which the dyslexia group takes longer to arrive at before reading – both show a significantly greater number of fixations before compared with the control group. In terms of text density, Program 2 is different from Program 1 and 3. Since the main code section is a block of sequential statements, there is not the horizontal or vertical spacing associated with the code layout compared to the other programs. Line 06, average-L09 and num1-L09 have fixation



```
import java.util.Scanner;

public class eyeTrack2 {

    public static void main(String[] args)
    {
        Scanner in = new Scanner(System.in);
        System.out.print("First number: ");
        int num1 = in.nextInt();
        System.out.print("Second number: ");
        int num2 = in.nextInt();
        int average = ( num1 + num2) / 2;
        System.out.print("Average: ");
        System.out.print(average);

    }

}
```

```
import java.util.Scanner;

public class eyeTrack2 {

    public static void main(String[] args)
    {
        Scanner in = new Scanner(System.in);
        System.out.print("First number: ");
        int num1 = in.nextInt();
        System.out.print("Second number: ");
        int num2 = in.nextInt();
        int average = ( num1 + num2) / 2;
        System.out.print("Average: ");
        System.out.print(average);

    }

}
```

Fig. 10 Heatmap (fixation duration) for Program 2. Red represents maximum duration values, dyslexia group = 7.71 s, control group = 6.97 s

counts and regression measures which stand out from other lines. It may be the case that a crowding effect is such that line 07 and 08 are “hidden” from the dyslexia programmer?

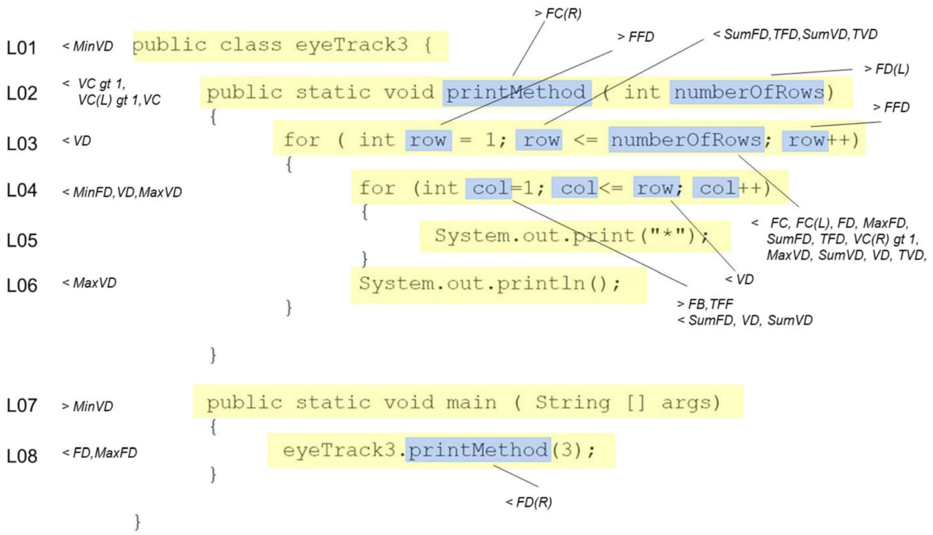
Line 06 appears to be a differentiator between the two groups. Fixation count, fixation duration, visit count and visit duration are all significantly longer for the dyslexia group. Lines 05 and 06, and lines 07 and 08, are effectively the same code but for different inputs. Maybe by understanding lines 05 and 06, an understanding of lines 07 and 08 follow but, in the case of the dyslexia group, this requires more time overall to identify and recognize the similarities. An alternative explanation is that lines 07 and 08 are receiving less attention due to crowding.

5.3 Program 3

Figure 11 shows those metrics from Program 3 for which there was a significant difference between the groups. Figure 12 shows the heat map (fixation duration) for reference, indicating in broad terms more dispersed areas of relatively high gaze intensity exhibited by the dyslexia group across the program.

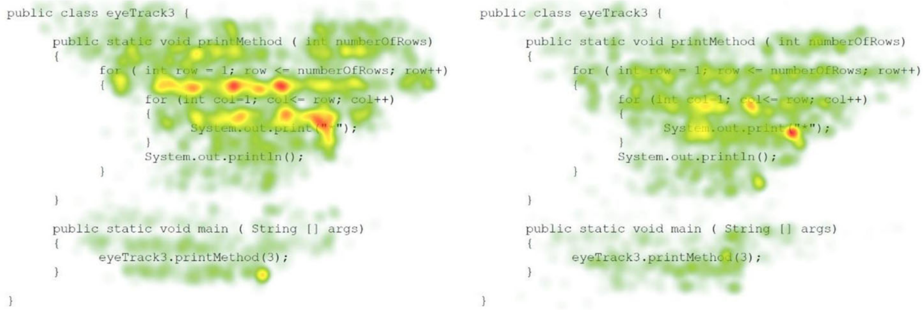
Program 3 is the most algorithmically complex of the three programs, consisting of a method declaration (printMethod) and a call to this from the main method. In addition, printMethod involves a nested for loop. When testing hypothesis 1 and 2, Program 3 did not show any AOIs where the predicted dyslexia gaze behaviour occurred.

Gaze activity for both groups was mainly focused on the header lines of the two for loops (lines 03 and 04). This is expected behaviour when seeking to understand such a program. Within these lines there are some notable differences in behaviour. Metrics related to fixation duration and visit duration (specifically visit duration for lines 03 and 04, and minimum fixation duration and maximum visit duration for line 04), are all significantly lower for the dyslexia group. This is effected by, and related to, similar gaze metrics on the loop variables therein. For example, for the second occurrence of the row identifier in line 03, the dyslexia group has significantly lower values for total fixation and total visit duration. Also on line 03, numberOfRows-L03 has significantly lower values on a range of count and duration metrics



```
L01  public class eyeTrack3 {
L02      public static void printMethod ( int numberOfRows)
         {
L03          for ( int row = 1; row <= numberOfRows; row++)
             {
L04              for (int col=1; col<= row; col++)
                 {
L05                  System.out.print("*");
                 }
L06              System.out.println();
             }

         }
L07      public static void main ( String [] args)
         {
L08          eyeTrack3.printMethod(3);
         }

     }
```

Fig. 11 Significant gaze metrics for Program 3



```
public class eyeTrack3 {

    public static void printMethod ( int numberOfRows)
    {
        for ( int row = 1; row <= numberOfRows; row++)
        {
            for (int col=1; col<= row; col++)
            {
                System.out.print("*");
            }
            System.out.println();
        }

    }

    public static void main ( String [] args)
    {
        eyeTrack3.printMethod(3);
    }

}
```

Dyslexia Group

```
public class eyeTrack3 {

    public static void printMethod ( int numberOfRows)
    {
        for ( int row = 1; row <= numberOfRows; row++)
        {
            for (int col=1; col<= row; col++)
            {
                System.out.print("*");
            }
            System.out.println();
        }

    }

    public static void main ( String [] args)
    {
        eyeTrack3.printMethod(3);
    }

}
```

Control Group

Fig. 12 Heatmap (fixation duration) for Program 3. Red represents maximum duration values, dyslexia group = 10.52 s, control group = 14.12 s

(fixation count, fixation duration, regression, total fixation duration, visit duration, maximum visit duration, total visit duration, maximum fixation duration). These values are contrary to what is predicted by hypothesis 1, 2 and 3. It is also the case that in line 03, for the first and third instance of the row identifier, the dyslexia group shows a significantly higher value for first fixation duration. The pattern here seems to be that the "middle" of the program feature (line 03) attracts less gaze from the dyslexia group, and the "edges" of the feature (the first and last occurrences of row), attract a significantly longer initial gaze (first fixation duration) compared to the control group. Is it again the case that an AOI which is "crowded" is less cognitively accessible for dyslexia programmers, leading to code reading deficiency?

In the case of line 04, for the dyslexia group, the first instance of the col. identifier and the first instance of the row identifier have significantly lower values for visit duration. Also for the dyslexia group, attention on the first instance of col. is delayed, with significantly larger values for fixations before and time to first fixation.

Line 02 is the signature of the printMethod code. In making sense of how this method is called and the role of its parameters, some regression of gaze is to be expected (*Flicking* according to the EMIP coding scheme). The dyslexia group shows an overall visit count which is significantly lower than the control group, and also significantly fewer regressions. For the identifiers in line 02, the dyslexia group has gaze metrics significantly higher than the control group - printMethod-L02-R fixation count is higher and numberOfRows-L02-L fixation duration is higher. The printMethod method is called from line 08, a line for which the dyslexia group has significantly lower fixation duration metrics.

Overall for Program 3, the hypothesized behaviour is not exhibited. Spatially, there is some pattern apparent whereby the dyslexia group payed more attention to the "edges" of the source code and less attention to the "middle" of the printMethod code. This is again broadly consistent with the tentative notion of crowding as mentioned above, but the dynamics would appear to be complex, and further investigation will require careful consideration of experimental design.

6 Threats to Validity and Study Limitations

Before the conclusions are presented, threats to validity and limitations of the study are outlined. Siegmund and Schumann (2015) present a comprehensive set of confounding parameters to be considered in program comprehension experiments. The study described here was able to control for a number of these as follows:

6.1 Participant Factors

All participants in the study were undergraduate students. This was not considered to be a threat to validity given that the program comprehension tasks undertaken were straightforward and also that the programming experience was beyond novice level (see Section 3.2). Furthermore, the effect, if any, of dyslexia would not be expected to change on transition to becoming a professional programmer.

Participant intelligence was not explicitly controlled for. However, all participants were students on the same level of computing degree, with similar entry requirements. Individual technical ability was indirectly measured by scoring each participant on their comprehension task. Between group analysis showed that there was no significant difference in comprehension performance between the groups.

In terms of familiarity with the Java programming language, while all participants had a familiarity with programming in general, not all used Java as their main programming language. However, the level of Java experience was consistent across the two groups (see Section 3.2 above). The period of programming experience was broadly similar across the two groups (dyslexia group mean = 39.86 months (*SD* = 29.31), control group mean = 34.71 months (*SD* = 17.64), *t*(26) = 0.56, *p*(2-tailed) = .579.

6.2 Experimental Factors

Participant related experimental parameters were controlled through the experiment protocol in a number of ways. Apprehension arising from participants being "evaluated" was mitigated by ensuring anonymity and that outcomes had no bearing on the individual as a student. The Hawthorne effect was minimized as the hypotheses were not revealed to participants – they were simply informed that the experiment was concerned with identifying if there were differences in program code reading behaviour between programmers with dyslexia and those without dyslexia. There was consistency in compliance with the experimental process through the issue of clear instructions as well as observation throughout. There was no time limit set for completing tasks; participants were reminded that they could take as long as necessary to complete the comprehension task, reducing the performance impact of any perceived time pressure.

Technical experimental parameters were managed in a number of ways. It was assumed that the eye tracker device was a novel instrument for all participants. Consequently, at the outset of the experiment, participants were shown a sample video replay of how eye movements could be recorded and measured, assuring the participants that no direct "video" of the session was being recorded. The potential for mono-operation bias was addressed through asking participants to complete the comprehension task for three programs rather than just one.

The spatial resolution of the Tobii X60 eye tracker could be a threat to validity with respect to vertical accuracy when a small number of readings are used. For the Tobii device, the accuracy specification is 0.5 degrees. This source of random error was mitigated in a number of ways. First, built in calibration was applied for each participant. This resulted in one participant not being able to proceed with the recording due to an unsuccessful calibration attempt. Second, large fonts (22.5 pt) were used when displaying the source code, maximizing the granularity of the areas of interest within the constraints of the program size and screen size. Third, the collection of a large number of gaze points (on average 7500 gaze points per participant per program) helps to minimize the effect of the random error. The error is also consistent across recordings, with each participant being recorded under the same conditions.

Context-related parameters were also considered. Any learning effects due to completion of the tasks amounted to the increasing familiarity with the style of code presented as the participant progressed from Program 1 through to Program 3. The programs were always presented in the same order and was therefore consistent across all participants. Related to this, the same instructions were used in each experiment, as well as a standard participant information sheet issued in advance of the study.

6.3 Study Limitations

Dyslexia is a multi-faceted condition, manifesting in different people in different ways. The experiment described does not account for different types such as peripheral dyslexia or central dyslexia. For example, if the dyslexia group was characterized mainly by deficits in visual processing such as letter position or letter identity dyslexia, we might expect different gaze behaviors than if the group was primarily characterized by phonological dyslexia. Associated with this is the difficulty in interpreting what the gaze metrics mean in this context. The actual cognitive process of understanding program code is potentially a confounding parameter. A large value for, say, fixation duration, could be due to semantic complexity of the programming construct or it could be due to the lexical complexity of an identifier or keyword. However, if dyslexia is a contributing factor to this, it is in part addressed by the control group comparison.

Analysis has focused on the subset of program features consisting of lines and identifiers. Given that reading program code is known to consist of both a sequential and scanning component, it is possible that other, less linear, AOIs could shed light of gaze difference between the groups. For example, AOIs could be designated consisting of method signatures and method body, or loop headers and loop body. There are also other subsets of program features still to be analyzed such as keywords or operators.

A further limitation arises from the statistical techniques used. Gaze recordings were only used where calibration had been successful. However, not all recordings achieved a 100% sampling rate. The weighted gaze sample values (the ratio of eye tracking samples that were correctly identified to the number of attempts) ranged from 98 to 63% with an average value of 88%. Consequently for individual gaze metrics on an AOI, data from the full set of participants was not always available. However, an average sample value of 88% is considered satisfactory. Where the degree of freedom is less than 26, this is reported in the affected statistics.

7 Conclusions and Further Work

The study described is the first to investigate, using eye tracking technology, how programmers with dyslexia read program code. The primary research question was, when reading program code for the purpose of comprehension, do the eye movements of programmers with dyslexia differ from those of programmers with typical reading profiles? The literature describing the eye movement behaviour of dyslexic readers when reading text suggested that there were particular behaviors which might be expected when reading program source code. These expected behaviors have been set out in the experiment's hypotheses. Work which has been done elsewhere using eye tracking technology to investigate how programmers read code, both expert and novice, has shown that reading code is not like reading text (Section 2.2 above). There are syntactic, semantic and structural differences which would suggest that differences between programmers with dyslexia and typical programmers may not be evident when reading code.

The results presented in this paper would suggest that observations and theories regarding dyslexic readers and their eye gaze behaviour do not necessarily map onto the gaze behaviour of programmers with dyslexia when reading program code. For each of the hypotheses, there was no convincing data to support the alternate hypothesis – there was no evidence of gaze behaviour relating to the selected program features which consistently showed a difference between programmers with dyslexia and typical programmers. This is not to say that a programmer with dyslexia does not exhibit some deficiencies when reading code, but that any such deficiencies may be quite different in nature from those experienced when reading natural text. Since we did not distinguish between different types of dyslexia which might have been present within the study group, we cannot determine if, for example, a programmer with letter position dyslexia will experience problems in, say, tracing data flow across identifiers, in a way that other programmers would not.

There are a number of features of program source code which by their nature might ameliorate dyslexia reading deficiency, features which would not typically be found in normal text. These include spacing arising from indentation, line breaks and the use of braces. For example, the inherent reading behaviour of scanning when comprehending program code would, by its nature, reduce any left mini-neglect behaviour evident when reading normal text.

When the small number of differences observed from hypothesis testing were interpreted in conjunction with those identified through the exploratory data analysis, some observations regarding differences in gaze behaviour did begin to emerge. Notably, programmers with dyslexia tended to show some inattention to those areas of the program code which, arguably, exhibited crowding – a known obstacle for dyslexic readers. The tentative observation that crowding could play a role when reading program code is a significant area for further study. It can be explored experimentally by more formally controlling the level of crowding in a number of programming scenarios, as proposed in (McChesney and Bond 2018).

In addition to investigating crowding, there is scope for other work arising from this study:

- Further testing of the current hypotheses using other AOIs, such as keywords, operators, or program feature abstractions, could provide additional data to expose any difference between the groups.
- For the programmer with dyslexia, there is scope to explore what is the essential nature of reading code and its linear and non-linear characteristics, for example, through the analysis of saccade data and related metrics for regression.
- One aspect of reading code which was not explored in this study was the effect of code comments. These are typically composed of natural text and are intended to assist with code understanding. How programmers with dyslexia read and process comments merits further study, especially as code comments introduce what would be expected to be a linear dimension to the reading process.
- It is worth noting that, though there were some tentative differences in gaze behaviour between the two groups, this did not lead to differences in program comprehension. Whatever the differences in reading behaviour, the ability and time to comprehend the program code was not statistically different across the two groups. However, an interesting question to pursue is if there is any correlation between dyslexia group gaze metrics, such as fixation count on particular AOIs, and program comprehension time or comprehension confidence.

- A further question which arises is whether gaze metrics could help identify a programmer with dyslexia. Since dyslexia is a complex and multi-faceted condition, it is very unlikely that, as a diagnostic device, eye gaze metrics when programming could serve this purpose. There are also ethical considerations in this regard. However, if the effects identified in this experiment could be replicated and standardized to other program scenarios, then it could be a useful insight to a programmer with dyslexia if their particular type of dyslexia is associated with distinct gaze behaviors when reading or writing code.

The exploratory nature of the study is such that, while of value in exploring dyslexia as an aspect of neurodiversity in programming, it is not sufficient to enable changes to theory relating to eye movement effect as proposed by theories of reading natural text. Traditional reading experiments which have informed theories about dyslexia have typically been conducted in a way which precisely controls the target text, for example, using RAN techniques. In terms of reading program code, the level of granularity required and operational definition of the dependent variables is beyond that which has been configured in the current experimental design.

If definite gaze patterns associated with dyslexia are identified through further work, then code style guidelines could be revised and features of integrated development environments (IDEs) could be enhanced to better support the programmer with dyslexia. For example, if the phenomenon of crowding were to be identified as a significant feature, then IDEs could support, for example, increased inter-word spacing, autoformatting to minimize crowding, or enhanced use of color. It so happens that many modern IDEs already support some level of interface customization, such as color highlighting and screen magnification, such that the programmer with dyslexia may already be finding ways of ameliorating any difficulties they encounter.

Finally, rather than programmers with dyslexia being considered to have a deficiency when reading and comprehending program code, it could be that they have some advantage. Recall that for some of the hypotheses, the dyslexia group had shorter fixation counts and fixation duration than the control group. As discussed in the introduction, previous work (Powell et al. 2004; Coppin 2008) has referred to the advantages which programmers with dyslexia might have when developing software, arising from their enhanced spatial awareness and visual learning style. Programmers with dyslexia should not necessarily assume that any deficiencies they experience when reading natural text will impair their ability to program, at least with respect to comprehension, as we did not find any significant difference in comprehension between the two groups. Further work would be valuable in exploring the advantages which a programmer with dyslexia might have when developing software.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

Adelman JS, Marquis SJ, Sabatos-DeVito MG (2010) Letters in words are read simultaneously, not in left-to-right sequence. Psychol Sci 21(12):1799–1801

Ali N, Sharafl Z, Gueheneuc Y, Antoniol G (2012) An empirical study on requirements traceability using eye-tracking. In: 28th IEEE International Conference on Software Maintenance (ICSM) pp. 191–200

Armstrong D, Squires G (2014) Key perspectives on dyslexia: an essential text for educators. Routledge

Bednarik R, Tukiainen M (2006) An eye-tracking methodology for characterizing program comprehension processes. In: proceedings of the 2006 symposium on eye tracking research & applications pp. 125–132

Bednarik R, Busjahn T, Schulte C (2014) Eye movements in programming education: analyzing the Expert's gaze: Proc First Int Workshop. Joensuu

Bellocchi S (2013) Developmental dyslexia, visual crowding and eye movements. In: Stewart LC (ed) Eye movement: developmental perspectives, dysfunctions and disorders in humans. Nova Science Publishers, pp.93–110

Bellocchi S, Muneaux M, Bastien-Toniazzo M, Ducrot S (2013) I can read it in your eyes: what eye movements tell us about visuo-attentional processes in developmental dyslexia. Res Dev Disabil 34(1):452–460

Binkley D, Davis M, Lawrie D, Maletic JI, Morrell C, Sharif B (2013) The impact of identifier style on effort and comprehension. Empir Softw Eng 18(2):219–276

Bishop-Clark C (1995) Cognitive style, personality, and computer programming. Comput Hum Behav 11(2): 241–260

Brooks R (1978) Using a behavioral theory of program comprehension in software engineering. In: proc of the 3rd Int Conf on Softw Eng, pp. 196–201

Brooks R (1983) Towards a theory of the comprehension of computer programs. Int J Man Mach Stud 18(6): 543–554

Busjahn T, Schulte C, Busjahn A (2011) Analysis of code reading to gain more insight in program comprehension. In: proc 11th Koli calling international conference on computing Education Research, pp. 1–9

Busjahn T, Bednarik R, Schulte C (2014a) What influences dwell time during source code reading?: analysis of element type and frequency as factors. In: Proc of the Symposium on Eye Tracking Research and Applications, pp 335–338

Busjahn T, Schulte C, Sharif B, Begel A, Hansen M, Bednarik R, Orlov P, Ihantola P, Shchekotova G, Antropova M (2014b) Eye tracking in computing education. In: proc 10th annual Conf Int computing Education Research, pp 3–10. https://doi.org/10.1145/2632320.2632344

Busjahn T, Bednarik R, Begel A, Crosby M, Paterson JH, Schulte C, Sharif B, Tamm S (2015) Eye movements in code reading: relaxing the linear order. In: 23rd IEEE International Conference on Program Comprehension (ICPC), pp 255–265

Coltheart M, Curtis B, Atkins P, Haller M (1993) Models of reading aloud: dual-route and parallel-distributed-processing approaches. Psychol Rev 100(4):589

Coltheart M, Rastle K, Perry C, Langdon R, Ziegler J (2001) DRC: a dual route cascaded model of visual word recognition and reading aloud. Psychol Rev 108(1):204

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

Coppin P, Hockema SA (2009) Learning from the information workspace of an information professional with dyslexia and ADHD. In: Int Conf on Science and Technology for Humanity (TIC-STH), pp 801–807

Cruz S, da Silva FQ, Capretz LF (2015) Forty years of research on personality in software engineering: a mapping study. Comput Hum Behav 46:94–113

De Luca M, Borrelli M, Judica A, Spinelli D, Zoccolotti P (2002) Reading words and pseudowords: an eye movement study of developmental dyslexia. Brain Lang 80(3):617–626

De Smet B, Lempereur L, Sharafi Z, Guéhéneuc YG, Antoniol G, Habra N (2014) TAUPE: visualizing and analyzing eye-tracking data. Sci Comput Program 79:260–278

Ducrot S, Lété B, Sprenger-Charolles L, Pynte J, Billard C (2003) The optimal viewing position effect in beginning and dyslexic readers. Curr Psychol Lett. Behaviour, Brain & Cognition, 10(1) https://journals.openedition.org/cpl/99

Engbert R, Nuthmann A, Richter EM, Kliegl R (2005) SWIFT: a dynamical model of saccade generation during reading. Psychol Rev 112(4):777

Facoetti A, Molteni M (2001) The gradient of visual attention in developmental dyslexia. Neuropsychologia 39(4):352–357

Friedmann N, Coltheart M (2016) Types of developmental dyslexia. In: Bar-On A, Ravid, D (eds) Handbook of communication disorders: Theoretical, empirical, and applied linguistics perspectives. De Gruyter, pp 721–752

Hari R, Renvall H (2001) Impaired processing of rapid stimulus sequences in dyslexia. Trends Cogn Sci 5(12): 525–532

Hawelka S, Gagl B, Wimmer H (2010) A dual-route perspective on eye movements of dyslexic readers. Cognition 115(3):367–379

Howell D (2012) Statistical methods for psychology (8/e), Wadsworth Publishing

Jbara A, Feitelson DG (2017) How programmers read regular code: a controlled experiment using eye tracking. Empir Softw Eng 22(3):1440–1477

Kevic K, Walters BM, Shaffer TR, Sharif B, Shepherd DC, Fritz T (2017) Eye gaze and interaction contexts for change tasks–observations and potential. J Syst Softw 128:252–266

Law C, Cupples L (2017) Thinking outside the boxes: using current reading models to assess and treat developmental surface dyslexia. Neuropsychol Rehabil 27(2):149–195

Maalej W, Tiarks R, Roehm T, Koschke R (2014) On the comprehension of program comprehension. ACM Trans Softw Eng Methodol (TOSEM) 23(4):31

McChesney I, Bond R (2018) Gaze Behaviour in Computer Programmers with Dyslexia - Considerations Regarding Code Style, Layout and Crowding. In: EMIP'18: Symposium on Eye Movements in Programming, June 14–17, 2018, Warsaw, Poland. https://doi.org/10.1145/3216723.3216724

Morris MR, Begel A, Wiedermann B (2015) Understanding the challenges faced by neurodiverse software engineering employees: towards a more inclusive and productive technical workforce. In: proc of 17th Int ACM SIGACCESS Conf on computers & accessibility, pp 173–184

Oracle (1997) Java Code Conventions http://www.oracle.com/technetwork/java/codeconventions-150003.pdf Accessed 17 May 2018

OUP (2015) "dyslexia." In: Wallace, S (ed) A Dictionary of Education, Oxford University Press

Pennington N (1987) Stimulus structures and mental representations in expert comprehension of computer programs. Cogn Psychol 19(3):295–341

Pennington BF, Peterson RL (2015) Development of dyslexia. In: Pollatsek A, Treiman R (eds) The Oxford handbook of Reading, Oxford University Press, pp 361–376

Perea M, Panadero V, Moret-Tatay C, Gómez P (2012) The effects of inter-letter spacing in visual-word recognition: evidence with young normal readers and developmental dyslexics. Learn Instr 22(6):420–430

Powell N, Moore D, Gray J, Finlay J, Reaney J (2004) Dyslexia and learning computer programming. Innovation in Teaching and Learning in Information and Computer Sciences 3(2):1–12. https://doi.org/10.11120/ital.2004.03020005

Prado C, Dubois M, Valdois S (2007) The eye movements of dyslexic children during reading and visual search: impact of the visual attention span. Vis Res 47(19):2521–2530

Radach R, Kennedy A (2013) Eye movements in reading: some theoretical context. Q J Exp Psychol 66(3):429–452

Rayner K (2009) Eye movements and attention in reading, scene perception, and visual search. Q J Exp Psychol 62(8):1457–1506

Reichle ED, Sheridan H (2015) EZ reader: an overview of the model and two recent applications. In: Pollatsek A, Treiman R (eds) The Oxford handbook of Reading, Oxford University Press, pp 277–290

Rodeghero P, McMillan C, McBurney PW, Bosch N, D'Mello S (2014) Improving automated source code summarization via an eye-tracking study of programmers. In: Proc 36th Int Conf on Softw Eng, pp 390–401

Salleh N, Mendes E, Grundy J (2014) Investigating the effects of personality traits on pair programming in a higher education setting through a family of experiments. Empir Softw Eng 19(3):714–752

Schroeder S, Hyönä J, Liversedge SP (2015) Developmental eye-tracking research in reading: introduction to the special issue. J Cogn Psychol 27(5):500–510

Schulte C, Clear T, Taherkhani A, Busjahn T, Paterson JH (2010) An introduction to program comprehension for computer science educators. In: Proc 2010 ITiCSE Working Group Reports, pp 65–86

Sexton CC, Gelhorn HL, Bell JA, Classi PM (2012) The co-occurrence of reading disorder and ADHD: epidemiology, treatment, psychosocial impact, and economic burden. J Learn Disabil 45(6):538–564

Shaft TM, Vessey I (1995) The relevance of application domain knowledge: the case of computer program comprehension. Inf Syst Res 6(3):286–299

Sharif B, Shaffer T, Wise J, Maletic JI (2016) Tracking Developers' eyes in the IDE. IEEE Softw 33(3):105–108

Sharma K, Jermann P, Nüssli MA, Dillenbourg P (2012) Gaze Evidence for different activities in program understanding. In: Proc 24th Annual Conf of Psychology of Programming Interest Group (No. EPFL-CONF-184006)

Shneiderman B, Mayer R (1979) Syntactic/semantic interactions in programmer behavior: a model and experimental results. Int J Parallel Prog 8(3):219–238

Siegmund J, Schumann J (2015) Confounding parameters on program comprehension: a literature survey. Empir Softw Eng 20(4):1159–1192

Spinelli D, De Luca M, Judica A, Zoccolotti P (2002) Crowding effects on word identification in developmental dyslexia. Cortex 38(2):179–200

Von Mayrhauser A, Vans AM, Howe AE (1997) Program understanding behaviour during enhancement of large-scale software. J Softw-Evol Proc 9(5):299–327

Ian McChesney is a Senior Lecturer in Computing Science. His first degree is in computer science and he has a PhD in software engineering. He is also a Member of The British Computer Society and is a Chartered IT Professional and Chartered Engineer. His research interests are in the areas of software engineering teamwork, process modeling, open data and sensor ontologies, and computer science education.

Raymond Bond is a Senior Lecturer in Data Analytics. He has research interests within biomedical informatics, clinical decision making, data science and human-computer interaction. He has ~180 publications and been a grant holder on research projects funded by EPSRC, ESRC, AHRC, FP7, H2020, InvestNI, Innovate UK, Samaritans Ireland, InterTrade Ireland and the Royal Irish Academy. Raymond obtained his BSc and PhD in the School of Computing and Mathematics (Ulster University). Before his academic appointment, he worked in the IT industry as a Web Programmer and later held Research Associate positions in the subject areas of Connected Health and Computerized Electrocardiography.