# Color Perception in Programming

Sohini Mazumder
Clemson University
Clemson, SC, USA
sohinim@clemson.edu

Mary Walker Felder
Clemson University
Clemson, SC, USA
mwfelde@clemson.edu

Connie Ku
Clemson University
Clemson, SC, USA
ku@clemson.edu

## ABSTRACT

Abstract . . . Placeholder Text

## CCS CONCEPTS

• ;

## KEYWORDS

eye tracking, visual attention, syntax highlighting, error finding

**ACM Reference Format:**
Sohini Mazumder, Mary Walker Felder, and Connie Ku. 2024. Color Perception in Programming. In . ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

## 1 INTRODUCTION

Placeholder Text

## 2 BACKGROUND

*Eye Tracking in Software Engineering*

Eye tracking is often used to understand and quantify the way the participant interacts with visual information. While there are other options to learn about the choices a participant makes, such as asking them to explain their actions and thoughts, participants' perceptions do not always align with their underlying process [Sharafi et al. 2020]. Eye tracking allows for an understanding of how the participants act without the interference of their perception and quantifiable information on how that interaction occurs. Eye tracking can show where individuals found areas of interest, where they fixated on the image or text, and how long each of these fixations lasted. This allows for a quantitative understanding of participants' attention and effort. According to [Sharafi et al. 2020], eye tracking in software engineering often occurs in a few typical types of studies. These include program comprehension, diagram comprehension, code reviews, traceability, and code summation. This understanding can be applied to many areas of software engineering to better understand how programmers interact with the code that they work with. In this particular case, this paper is focused on code reading and the possible impacts that syntax highlighting might have on how individuals read code.

*Syntax Highlighting*

Syntax highlighting might enhance error detection as color helps users to differentiate code elements. In Gestalt psychology, which is a theory of mind that focuses on how people perceive and process information, the human brain naturally organizes similar visual stimuli (e.g., color) into groups (perceptual grouping; [Wertheimer 1938]; [Koffka 2013]). This innate perceptual grouping process, reduces users cognitive workload as they can simplify information processing by reading them in as a group, rather than a single word or item. In the context of detecting errors in programming, highlighted syntax prompts the brain to group those elements into a certain category which promotes readability. This could result in fewer fixation counts because users are able to process the grouped element more efficiently. For example, syntax highlighting may allow users to understand the structure of the code by just scanning it once instead of fixating their gaze across individual words. Similarly, fixation duration may also be reduced as a result of perceptual grouping stimulated by highlighted syntax. When similar elements are grouped together by color, users may need less time to read and process the code because they are able to process the grouped element as a whole.

The findings of [Sarkar 2015] support the benefits of syntax highlighting by examining its effect on program comprehension. In their experiment, ten graduate computer science students were asked to read three pairs of Python code. Each pair included one code with syntax highlighting and another without it. The participants' goal was to determine the output of the code they viewed. Variables such as fixation count, fixation duration, context switches, and completion time were measured and analyzed. The study found that participants completed the task faster when syntax was highlighted compared to when it was not. Eye-tracking data also showed that the number of context switches was lower with syntax highlighting than without. However, the study did not find any significant effect on fixation counts and durations, which could be attributed to the small sample size. As mentioned by the authors, among the ten sets of data, three of them were excluded due to poor fixation data caused by participants wearing glasses. This further reduced sample size, threatening the internal validity; therefore, a significant effect on fixation counts and durations were not observed.

*Impact of syntax highlighting in Code reading*

The study conducted in [Beelders and du Plessis 2016] tried to gauge a broad understanding of the impact of color coding when a programmer reads a piece of code. The snippets provided to the participants were written in C#, a language understood by everyone in the group and multiple metrics were used to test the impact.


*Conference'17, July 2017, Washington, DC, USA*

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

The metrics used were: Fixation count, fixation count per sentence, fixation count per word, fixation durations and regressions. Data collected was visualized using heatmaps (the same one used in (Busjahn, Shulte, & Busjahn, 2011)), and overall, the major hypotheses derived from the fixation metrics was: *There is no difference between the number of fixations per line/AOI or in total fixation count between the black-and-white and colored code snippets.* Moreover, they also hypothesized that: *There is no difference in the number of regressions when reading code in black-and-white versus color.*

The colored code was noted to impact the readability of the code - as participants noted that it was easier to read the color coded code over the black-and white. Another observation made during the experiment was that words in a warmer color were fixated on for longer than words in cooler colors, though the difference was not overly significant to create a benchmark for any results. The percentage of regressions was marginally less than previous findings by (Busjahn, Shulte, & Busjahn, 2011) - which was attributed to the subjective way of counting the regressions. Despite almost identical results for both the color coded and black and white snippets, throughout all the metrics, the values for the color coded snippets remained lower. This proved that even though not as significant, color coded code posed comparatively less difficulty to the reader.

## 3 EMPIRICAL VALIDATION

Placeholder Text

### 3.1 Experimental Design

Placeholder Text

### 3.2 Participants

Placeholder Text

### 3.3 Procedure

Placeholder Text

### 3.4 Apparatus

Placeholder Text

## 4 DISCUSSION

Placeholder Text

## 5 LIMITATIONS & FUTURE WORK

Placeholder Text

## 6 CONCLUSION

Placeholder Text

## REFERENCES

Beelders, T. and du Plessis, J.-P. L. (2016). Syntax highlighting as an influencing factor when reading and comprehending source code. *Journal of Eye Movement Research*, 9(1).

Koffka, K. (2013). *Principles of Gestalt psychology.* routledge.

Sarkar, A. (2015). The impact of syntax colouring on program comprehension.

Sharafi, Z., Sharif, B., Guéhéneuc, Y.-G., Begel, A., Bednarik, R., and Crosby, M. (2020). A practical guide on conducting eye tracking studies in software engineering. *Empirical software engineering : an international journal*, 25(5).

Wertheimer, M. (1938). Laws of organization in perceptual forms. *Psycologische Forschung, 4, 301-350.*