

Detecting the Onset of Eye Fatigue in a Live Framework

Dillon J. Lohr, Evgeny Abdulin, and Oleg V. Komogortsev*
Computer Science Department, Texas State University

Abstract

This document describes a method for detecting the onset of eye fatigue and how it could be implemented in an existing live framework. The proposed method, which uses fixation data, does not rely as heavily on the sampling rate of the eye tracker as do methods which use saccade data, making it more suitable for lower cost eye trackers such as mobile and wearable devices. By being able to detect eye fatigue with such eye trackers, it becomes possible to react to the development of fatigue in virtually any environment, such as by alerting drivers that they appear fatigued and may want to pull over. It could also be used to aid in developing interfaces that are more user-friendly by noting at which point a user becomes fatigued while navigating the interface.

Keywords: eye fatigue, eye movements

Concepts: • Human-centered computing ~ HCI design and evaluation methods; *Usability testing*

1 Introduction

Detecting eye fatigue through eye movements is not a new concept; in fact, methods already exist for determining whether a user is fatigued by examining the saccadic information of his/her eye movements [Bahill and Stark 1975; Megaw and Sen 1983; Stasi et al. 2013]. However, these methods require an eye tracker with a high sampling rate of at least 250 Hz to accurately measure the quick movements of saccades [Abdulin and Komogortsev 2015] and so are not currently feasible for mobile and wearable devices.

To address this concern, we have created a method that determines if and when a user is fatigued by examining the fixation information of his/her eye movements [Abdulin and Komogortsev 2015]. Comparisons of our method to existing, saccade-based methods are detailed in their own paper, cited above; but this document focuses on the application of our method in a live framework.

The live framework we are using is part of a larger biometric framework that was designed to work with most regular eye trackers [Holland and Komogortsev 2014], making it appropriate for use in mobile and wearable devices. In this paper, we describe the implementation of our method for detecting the onset of eye fatigue using this live framework, but details are not framework-specific.

*e-mail: djl70@txstate.edu, abdul@txstate.edu, ok11@txstate.edu.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org. © 2016 Copyright held by the owner/author(s). Publication rights licensed to ACM.

2 Method Implementation

Our fixation-based method has three main components: determining the fatigue threshold, calculating the user's initial fixation qualitative score (FQIS), and comparing subsequent FQIS values to that baseline to see if the difference exceeds the threshold, indicating fatigue. For testing, users performed 100 horizontal saccadic jumps with 1 second between jumps at $\pm 15^\circ$ from the center of the screen for a total saccade amplitude of 30° , because 80 30° -degree saccades are enough to create symptoms of eye fatigue [Bahill and Stark 1975].

2.1 Determining the Fatigue Threshold

The fatigue threshold, T_F , is calculated using an empirical formula that depends on the average spatial accuracy in degrees, θ_{avg_acc} , of the eye tracker so that the threshold can scale with noisier signals. There are two constant numbers: A represents the difference in FQIS between the first fatigued group of data and the initial FQIS (from the first group of data), while μ represents the mean spatial accuracy of the data that led to finding A . These constants were found to be $A = 0.15^\circ$ and $\mu = 0.88^\circ$ from the results of the initial tests of the method in [Abdulin and Komogortsev 2015]:

$$T_F = A \times \frac{\theta_{avg_acc}}{\mu}$$

The average spatial accuracy is calculated during calibration by finding the mean gaze point, \bar{G}_i , for each calibration point, P_i , and then calculating the average distance in degrees, θ_i , between each calibration point and its resulting gaze point (xy-coordinates are in degrees from the top left of the display):

$$\theta_{avg_acc} = \frac{1}{n} \sum_{i=1}^n |P_i - \bar{G}_i|$$

2.2 Calculating the Initial FQIS

FQIS, or the fixation qualitative score, is a standardized metric that represents the distance between the fixation components of the eye movement signal and the stimulus [Komogortsev et al. 2010].

$$FQIS = \frac{1}{n} \sum_{i=1}^n |s_i - g_i|$$

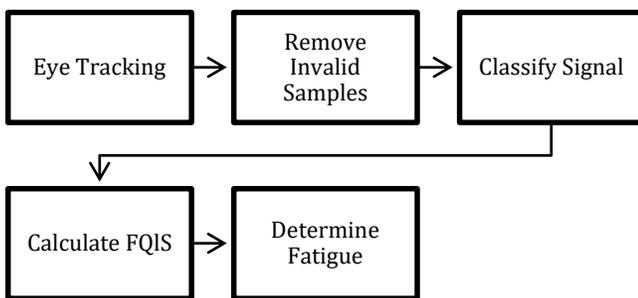
To calculate FQIS, the signal must first be classified. Once the signal is classified, its FQIS is calculated by averaging the distances between the stimuli, s_i , and the gaze points, g_i , as seen in the equation above. The classification method we chose was a simplified version of that found in [Nyström and Holmqvist 2010]. First, the velocity peaks above a threshold of $150^\circ/s$ were found to detect saccades. Then, the onsets and offsets of saccades were found as local minima to the left and to the right from the peak, accordingly.

$$Velocity = \frac{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}}{t_2 - t_1}$$

Saccades with an amplitude below 0.5° were merged with the surrounding fixation, and fixations with a duration below 100ms were merged with the surrounding saccade. Any invalid samples were simply removed from the signal. We chose this method for its accuracy relative to speed, which is desirable for a live framework that will be used on mobile and wearable devices. For the initial FQIS calculation, the first 10 seconds of data are taken into account.

2.3 Checking for Fatigue

To check for eye fatigue, every subsequent 10-second signal is classified and its FQIS calculated. Then, the difference between its FQIS and the initial FQIS is compared to the fatigue threshold; if the difference is larger than the fatigue threshold, the user is probably experiencing eye fatigue. This is because it would have become increasingly difficult for the user to: 1) perform the stimulus jumps as quickly as before, causing some of the recorded gaze points to be further away from the stimulus, leading to a higher FQIS, and 2) keep the eyes as precisely and accurately on the stimulus as before, causing the gaze points to be further away from the stimulus on average, leading to a higher FQIS. The flowchart below summarizes the process of detecting fatigue:



3 Conclusion

Our approach has some limitations. For example, it assumes the user is not fatigued initially. This can be problematic if the user is, in fact, fatigued at the beginning, because it is unlikely the fatigue threshold will be crossed. One way to address this may be to incorporate a saccade-based method, such as average saccade duration, on top of our fixation-based approach to determine whether the user is initially fatigued. Another possible way is analyzing glissades which sometimes occur after saccades as a lower velocity peak instead of FQIS, because there is evidence that glissadic behavior is affected by a person's fatigue level [Bahill and Stark 1975]

That said, our method is functional even in its current, early state, and the idea behind it has potential. With more work and further testing, the accuracy of our method would improve, making it reliable in real-world scenarios.

Acknowledgements

Thanks to Corey D. Holland for creating the framework we used. This work is supported in part by Google Research Award #2014_R1_308, NSF CAREER Grant #CNS-1250718, and NIST

Grant #60NANB15D325.

References

- ABDULIN, E., AND KOMOGORTSEV, O. User Eye Fatigue Detection via Eye Movement Behavior. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, 1265-1270 (2015).
- BAHILL, A. T., AND STARK, L. Overlapping saccades and glissades are produced by fatigue in the saccadic eye movement system. *Experimental Neurology* 48, 95-106 (1975).
- HOLLAND, C. D., AND KOMOGORTSEV, O. V. Software framework for an ocular biometric system. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, 365-366 (2014).
- KOMOGORTSEV, O. V., ET AL. Standardization of Automated Analyses of Oculomotor Fixation and Saccadic Behaviors. *IEEE Transactions on Biomedical Engineering* 57, 11, 2635-2645 (2010).
- MEGAW, T., AND SEN, T. Visual fatigue and saccadic eye movement parameters. In *27th Annual Meeting Human Factor Society*, 728-732 (1983).
- NYSTRÖM, M., AND HOLMQVIST, K. An adaptive algorithm for fixation, saccade, and glissade detection in eyetracking data. *Behavioral Research Methods* 42, 1, 188-204 (2010).
- STASI, L. L. D., ET AL. Saccadic peak velocity as an alternative index of operator attention: A short review. *European Review of Applied Psychology* 63, 6, 335-343 (2013).