Journal of Neuroscience Methods 227 (2014) 121–131

Contents lists available at ScienceDirect

Journal of Neuroscience Methods

journal homepage: www.elsevier.com/locate/jneumeth

A nonparametric method for detecting fixations and saccades using cluster analysis: Removing the need for arbitrary thresholds

Seth D. König[a,b,e], Elizabeth A. Buffalo[b,c,d,*]

[a] Wallace H. Coulter Department of Biomedical Engineering at the Georgia Institute of Technology and Emory University, 313 Ferst Drive, Atlanta, GA 30332, USA
[b] Yerkes National Primate Research Center, 954 Gatewood Road, Atlanta, GA 30329, USA
[c] Department of Neurology, Emory University School of Medicine, 1440 Clifton Road, Atlanta, GA 30322, USA
[d] Department of Physiology and Biophysics, University of Washington, Seattle, WA 98195, USA
[e] Graduate Program in Neurobiology and Behavior, University of Washington, Seattle, WA 98195, USA

HIGHLIGHTS

- Standard saccade detection algorithms are insufficient for complex viewing tasks.
- Cluster Fix detects saccades with *k*-means cluster analysis in scan path state space.
- Global and local criteria increase the accuracy of saccade detection.
- A gold standard is required to compare viewing behavior across experiments.

ARTICLE INFO

Article history:
Received 11 August 2013
Received in revised form 17 January 2014
Accepted 20 January 2014

Keywords:
Saccade detection
Fixations
Eye tracking
Cluster analysis
Viewing behavior

ABSTRACT

Background: Eye tracking is an important component of many human and non-human primate behavioral experiments. As behavioral paradigms have become more complex, including unconstrained viewing of natural images, eye movements measured in these paradigms have become more variable and complex as well. Accordingly, the common practice of using acceleration, dispersion, or velocity thresholds to segment viewing behavior into periods of fixations and saccades may be insufficient.
New method: Here we propose a novel algorithm, called Cluster Fix, which uses *k*-means cluster analysis to take advantage of the qualitative differences between fixations and saccades. The algorithm finds natural divisions in 4 state space parameters—distance, velocity, acceleration, and angular velocity—to separate scan paths into periods of fixations and saccades. The number and size of clusters adjusts to the variability of individual scan paths.
Results: Cluster Fix can detect small saccades that were often indistinguishable from noisy fixations. Local analysis of fixations helped determine the transition times between fixations and saccades.
Comparison with existing methods: Because Cluster Fix detects natural divisions in the data, predefined thresholds are not needed.
Conclusions: A major advantage of Cluster Fix is the ability to precisely identify the beginning and end of saccades, which is essential for studying neural activity that is modulated by or time-locked to saccades. Our data suggest that Cluster Fix is more sensitive than threshold-based algorithms but comes at the cost of an increase in computational time.

1. Introduction

Rigorous analysis of eye movements dates back to the seminal work of Alfred Yarbus (Yarbus, 1967). Today, eye tracking is used to determine the location of visual attention (Duchowski, 2002; McAlonan et al., 2008; Lee et al., 2011), measure memory (Smith et al., 2006; Smith and Squire, 2008; Hannula and Ranganath, 2009; Jutras et al., 2009; Richmond and Nelson, 2009; Hannula et al., 2010; Jutras and Buffalo, 2010; Hannula et al., 2012; Killian et al., 2012), detect cognitive impairments (Crutcher et al., 2009; Lagun et al., 2011; Zola et al., 2013), and evaluate visual search strategies (Najemnik and Geisler, 2005; Dewhurst et al., 2012). The development of non-invasive infrared eye-tracking

* Corresponding author. Present address: Washington NPRC, Buffalo Lab, 1959 NE Pacific Street, Box 357290, Seattle, WA 98195, USA. Tel.: +1 404 712 9431.
E-mail address: ebuffalo@uw.edu (E.A. Buffalo).

0165-0270/$ – see front matter © 2014 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.jneumeth.2014.01.032

technologies has further enhanced the value and feasibility of collecting viewing behavior across a large range of experimental tasks.

Commonly, viewing behavior, represented as a scan path, is parsed into periods of fixations and saccades using a variety of algorithms. The most widely used algorithms employ velocity and/or acceleration thresholds to detect the occurrences of saccades because the velocity and acceleration of the eye are much greater during a saccade than during a fixation (Otero-Millan et al., 2008; Nystrom and Holmqvist, 2010; Kimmel et al., 2012). Threshold-based algorithms have the benefit of being intuitive, quick, and easy to implement. Other popular algorithms use density or dispersion and areas of interest (Tatler and Gilchrist, 2005; Ito et al., 2011). Variants and combinations of these algorithms include mechanisms to correct for errors in eye tracking such as blinks and other temporary losses of signal (Wass et al., 2013).

Despite a significant increase in the use of eye movements in neuroscience, there have been very few advances in the algorithms used to detect fixations and saccades (Salvucci and Goldberg, 2000). We could only find a few instances of algorithms that deviated significantly from the most widely used algorithms. Unfortunately, many of these alternative algorithms still employ a velocity threshold to detect potential saccades, followed by additional techniques including principal component analysis to distinguish between smooth pursuit, saccades, and noise (Berg et al., 2009; Liston et al., 2012). One exception is (Urruty et al., 2007) which used dispersion and projection clustering into arbitrary subspace to detect fixations. To the best of our knowledge, these algorithms have not been adopted in subsequent studies.

Algorithms employing velocity and acceleration thresholds for saccade detection may be sufficient for simple tasks in which subjects make predictable saccades toward a stationary target; however, more complex oculomotor tasks such as unconstrained viewing of natural scenes or dynamic stimuli may produce more variable eye movements (Andrews and Coppola, 1999; Hayhoe and Ballard, 2005; Berg et al., 2009; Rayner, 2009). A major source of this variability arises from the variability in saccade amplitude which is strongly correlated with the peak velocity of the saccade (Otero-Millan et al., 2008; Martinez-Conde et al., 2009, 2013). Velocity thresholds may not accurately parse highly variable scan paths into periods of fixations and saccades since saccade amplitudes and thus their peak velocity are not constrained in free viewing. Further, many algorithms employ arbitrary thresholds based on qualitative human observations which can vary across research laboratories and even from one experiment to the next within a laboratory. Finally, computed viewing behavior statistics including saccade rate, fixation duration, saccade duration, and saccade amplitude vary not only according to experimental variables but also by the method used to calculate them (Duchowski, 2007; Shic et al., 2008; Nystrom and Holmqvist, 2010). Therefore, there exists a need for a more accurate, sensitive, non-arbitrary, and completely automated saccade detection algorithm. Such an algorithm could constitute a "gold standard" for detecting fixations and saccades from scan paths so that viewing behavior could be accurately compared across experiments, laboratories, and algorithms.

Here we present a novel algorithm, called Cluster Fix, which applies *k*-means cluster analysis to parse scan paths into fixations and saccades. There are several clear qualitative differences between fixations and saccades—saccades are temporally short with a high velocity whereas fixations are longer in duration with a slower velocity. These qualitative differences translate into quantitative differences and the occupation of different regions in state space. Cluster Fix makes no assumptions about the arrangement of scan paths in state space, requires no human inputs, and includes only duration thresholds as free parameters.

2. Methods

2.1. Eye tracking

Scan paths were obtained at 200 Hz using an infrared eye tracker (ISCAN) from rhesus macaques freely viewing 288 images of natural scenes. Eye tacking data were collected from 4 adult male macaques seated head-fixed in a dimly illuminated room 60 cm away from a 19 in. CRT monitor with a refresh rate of 120 Hz. Images of natural scenes were 600×800 pixels large and subtended $25^\circ \times 33^\circ$ of visual angle (dva). Experimental control software (CORTEX http://dally.nimh.nih.gov/) displayed images for 10 s each. Initial calibration of the infrared eye tracking system consisted of a 9-point calibration task. Drift was tracked throughout the experiment by presenting additional calibration trials between image viewing trials. We excluded from further analysis any eye tracking data more than 50 pixels (2 dva) outside of the image. Blinks were rarely observed in our data so we did not make any corrections other than the exclusion of data outside of the image. Standard blink correction techniques should work with Cluster Fix if scan paths are evaluated in a piece-wise manner ignoring blinks and as long as at least one fixation is present in each evaluated portion of the scan path (Supplementary Fig. 1). All experiments were carried out in accordance with the National Institutes of Health guidelines and were approved by the Emory University Institutional Animal Care and Use Committee and Emory Institutional Review Board.

2.2. Cluster Fix algorithm

The Cluster Fix algorithm was written in MATLAB and is available as supplemental material. Table 1 contains the procedural outline detailing the major processes achieved by the algorithm. To avoid filtering artifacts, eye traces were buffered prior to filtering, filtered, and then the buffers were removed. First, horizontal and vertical eye traces from the viewing of each image were individually pre-processed using a polyphase implementation (MATLAB function RESAMPLE) to up-sample the data from 200 Hz to 1000 Hz

Table 1
Cluster Fix procedural outline.

1.	Pre-process and filter a. Up-sample horizontal and vertical eye traces from 200 Hz to 1000 Hz b. Low pass filter with a cutoff frequency of 30 Hz
2.	Calculate distance, velocity, acceleration, and angular velocity for every time point
3.	Move outliers and normalize a. Move outliers greater than the mean + 3 × std to the mean + 3 × std b. Individually normalize the 4 state space parameters to be from 0 to 1
4.	Global clustering a. Determine the number of clusters b. Cluster using *k*-means c. Determine fixation clusters and saccades clusters d. Reclassify fixations with durations less than 25 ms as saccades
5.	Local re-clustering a. Compare detected fixations to adjacent portions of the scan path b. Determine the number of clusters c. Cluster using *k*-means d. Determine fixation clusters and saccade clusters
6.	Reclassify global fixation time points that were locally determined to be saccades
7.	Consolidate using duration thresholds a. Classify fixations with durations less than 5 ms as saccades b. Reclassify saccades with durations less than 10 ms as fixations c. Reclassify fixations with durations less than 25 ms as saccades
8.	Post-processing a. Down-sample to acquisition frequency of 200 Hz