Modified DBSCAN Algorithm on Oculomotor Fixation Identification

Beibin Li,* Quan Wang, Erin Barney
Logan Hart, Carla Wall, Katarzyna Chawarska
Child Study Center
Yale University

Irati Saez de Urabain, Timothy J. Smith
Department of Psychological Sciences
Birkbeck College
University of London

Frederick Shic†
Child Study Center
Yale School of Medicine
Yale University

Abstract

This paper modifies the DBSCAN algorithm to identify fixations and saccades. This method combines advantages from dispersion-based algorithms, such as resilience to noise and intuitive fixational structure, and from velocity-based algorithms, such as the ability to deal appropriately with smooth pursuit (SP) movements.

Keywords: DBSCAN, Fixation Identification, Saccade, Machine Learning, Clustering

Concepts: •**Theory of computation** → *Unsupervised learning and clustering;* •**Computing methodologies** → *Cluster analysis;*

1 Introduction

Scientists have become increasingly interested in oculomotor fixation identification algorithms because these properties of eye movements have been associated with visual cognition [Liversedge and Findlay 2000]. The goal of fixation identification is to reduce the complexity of eye-tracking data while maintaining the essential components for cognitive and visual processing analyses [Salvucci and Goldberg 2000; Shic et al. 2008]. Most fixation identification algorithms can be classified as either dispersion or velocity based; we will use the Distance Dispersion Algorithm (I-DD) [Salvucci and Goldberg 2000] and the Velocity Threshold Method (I-VT) [Sen and Megaw 1984] as representatives of these algorithms.

These classic algorithms have limitations. Distance-based algorithms rarely identify SP. However, SP movements might share underlying cognitive and neural processes with fixations, and being able to group such movements appropriately with fixations could be advantageous [Krauzlis and Miles 1996]. On the other hand, Velocity-based algorithms can be susceptible to noise. Researchers continue to seek better methods to address these flaws. Sun and colleagues integrated DBSCAN and mathematical morphology clustering (MMC) to group drivers' gaze fixations [Sun et al. 2015], but ignored the temporal dimension, which is a crucial property of eye-tracking data. This paper modifies the DBSCAN algorithm for fixation identification analysis and compares it with I-DD and I-VT in Section 4.

2 Modified DBSCAN

DBSCAN is more complex than traditional fixation identification algorithms because it distinguishes core points, border points, and noise in a dataset. Two parameters are required for it: a distance (ϵ) and a minimum number of points ($minPts$). Point p is a core point if at least $minPts$ points are within distance ϵ to it, and these $minPts$ points are directly reachable from p. Points p and q are density reachable if there is a chain of points $p_0 = p, p_1, p_2, \ldots, q = q_k$, where p_i is directly reachable from p_{i-1} for all $i > 0$. A core point forms a cluster with all of its density reachable points. The points not belonging to any clusters are considered noise.

Fixation identification relies on the temporal properties of gaze data in eye-tracking analysis. Participants may stare at the same location several times, and these fixations should be analyzed separately rather than as a whole. Moreover, if different saccades pass over the same region repeatedly, clustering algorithms that ignore the temporal dimension could mistake the intersection of these saccades as a fixation, leading to incorrect interpretations. Therefore, we modified the definition of core point in DBSCAN. Point p is a core point if: 1. at least $minPts$ points are within distance ϵ to point p; and 2. these points form a consecutive subsequence $p_0, p_1, \ldots, p_k$ of the dataset, where p_i and p_{i-1} are adjacent in time. The pseudo-code is provided below. More information can be found at *https://github.com/BeibinLi/MDBSCAN*

```
# eps = epsilon
func dbscan( vdata, eps, minPts):
  for p in data:
    if ( p is visited ): continue;
    neighbors = regionQuery(p, eps);
    if (neighbors.size < minPts): p is noise = true;
    else:
       C = expandCluster(p, neighbors, eps, minPts);
       Recognize C as one fixation;

func expandCluster(p, neighbors, eps, minPts):
  Set C = {p}
  for (Point p' in neighbors ):
     if ( p' is visited ): continue;
     if ( p' is not clustered ):
        C.add(p'); # add p' to a cluster
     neighbors2 = regionQuery(p', eps)
     if (neighbors2.size >= minPts):
         neighbors.union(neighbors2)
  return C

func regionQuery(p, eps):
  Array ngb; # neighbor
  ngb.push_back(p);
  p' = p;
  while( p' = p' next point ):
     if( distance(p, p') <= eps ): ngb.push_back( p' );
     else break;
  p' = p;
  while( p' = p' previous point ):
     if( distance(p, p') <= eps ): ngb.push_front( p' );
     else: break;
  return ngb;
```

This modification allows us to apply DBSCAN to fixation identification problems. The regionQuery function in the original DB-

*e-mail: beibin.li@yale.edu
†e-mail: frederick.shic@yale.edu

ETRA '16, March 14-17, 2016, Charleston, SC, USA
ISBN: 978-1-4503-4125-7/16/03
DOI: http://dx.doi.org/10.1145/2857491.2888587

SCAN algorithm was redesigned to query the adjacent neighbor points. The worst case complexity of this algorithm is $O(n^2)$. The two parameters, ϵ and $minPts$, are closely related to the density of eye-gaze points. Theoretically, the choice of ϵ and $minPts$ should be related to the visual angle and frequency of the eye tracker. Practically, the two parameters can be simplified into one because $minPts$ does not influence results significantly if it is not extremely small [Ester et al. 1996]. We observed that the results for MDBSCAN are similar if $100 > minPts > 10$.

3 Subjects and Stimuli

We applied fixation identification algorithms to two databases: a complex sample comprised of 2-year-old children with and without Autism Spectrum Disorder (ASD, n=38) and a large dataset with 179 subjects hand-verified for fixations and saccades through semi-manual coding program GraFix [de Urabain et al. 2014].

We set $minPts = 20$. We also set ϵ to half of the classical distance threshold d_{max}, i.e. $\epsilon = 0.5^\circ$ in our analysis. A distance threshold of 1° was used in I-DD based on prior recommendations for psychological research [Blignaut 2009]. A velocity threshold of $30^\circ/sec$ was used in I-VT. Fixations less than 100 ms were rejected in all these algorithms based on the evidence that most eye fixations last more than 100 ms [Salvucci and Goldberg 2000]. When comparing our results with GraFix data, we implemented GraFix's fixation merge and RMS rejection methods in all the three algorithms. It should be noticed that GraFix uses I-VT as the underlying fixation identification algorithm, which means I-VT should produce similar results for GraFix software.

4 Results and Analysis

I-DD cannot identify SPs, while I-VT might produce fixations with only a few consecutive points [Salvucci and Goldberg 2000]. Moreover, the eyeball's movement speed varies in a large range, which can cause difficulty in choosing a single velocity threshold (e.g. $30^\circ/sec$ for static image, but $50^\circ/sec$ for SP.

For ASD, we computed temporal fixation overlap statistics for repeated 5-point calibration and identified between-algorithm differences with linear mixed models (LMM) (Bonferroni corrected) [F(2,487.9)=129.0,$p < .001$]: MDBSCAN (M=4.1,SD=1.5) = I-DD (M=4.0, SD=1.5))>I-VT (M=2.2,SD=1.9)($p < .001$), suggesting default I-VT parameters were inadequate for identifying fixations in this challenging sample. For SP, we compared percent SP trajectory coverage [F=13.2,p<.001] finding (MDBSCAN (M=77%,SD=25%) = I-DD (M=73%,SD=26%))>I-VT (M=63%,SD=32%). We computed # of fixations [F=155,p<.001], finding MDBSCAN (M=7.5,SD=2.6)<I-VT (M=10.9,SD=6.8) < I-DD(M=21.2,SD=10.4), and coverage/fixation [F=135,p<.001], finding MDBSCAN (M=12.7%,SD=6.7%) < I-VT (M=7.1%,SD=2.4%)<I-DD(M=4.2%,SD=1.3%). This suggests MDBSCAN has advantages during SPs.

For the GraFix data, we used LMMs to compare MDBSCAN, DD, and I-VT algorithms using GraFIX as a baseline. Algorithms differed in mean fixation time [F(2,178)=5.4,p<.01; IVT closest to GraFIX], Number of fixations/sec [F=18.1,p<.001; MDBSCAN closest], and percentage time in fixations [F=10.3,p<.001; DD closest], suggesting unique features of each algorithm. While MDBSCAN was not always closest to ground truth, determining "what is best" would require algorithm parameter search based on experimental, subject, and outcome measure properties. So, I-DD, I-VT, and MDBSCAN identifies different properties in gaze data.

MDBSCAN closely resembles I-VT and I-DT. If we set $minPts$ to 3, the MDBSCAN becomes I-VT, where ϵ defines the velocity threshold. On the other hand, if we only take one core point in one fixation with its directly reachable points and disable the ExpandCluster function, the MDBSCAN becomes similar to I-DD.

5 Conclusion

Future studies can work on finding a golden standard to evaluate fixation algorithms. For instance, Komogortsev and colleagues proposed a qualitative and quantitative scoring system for eye movement classification algorithms [2010]. De Urabain and colleagues [2014] also presented an efficient two-step semiautomatic method, GraFIX, to assess and adjust Velocity Threshold Algorithms result, which provides reliable and stable measures on eye tracking data.

MDBSCAN is designed to identify fixations in eye-tracking data, combining advantages of classical fixation identification methods. Further studies will explore the utility of this approach for analyzing a variety of eye-tracking studies in practice.

Acknowledgements

Funding was provided by K01 MH104739, R21 MH102572, CTSA UL1 RR024139, R03 MH092618, NIH R01 MH100182, R01 MH087554, U19 MH108206; NSF #1139078, #0835767, DOD W81XWH-12-ARP-IDA, and the Nancy Taylor Foundation.

References

BLIGNAUT, P. 2009. Fixation identification: The optimum threshold for a dispersion algorithm. *Attention, Perception, & Psychophysics 71*, 4, 881–895.

DE URABAIN, I. R. S., JOHNSON, M. H., AND SMITH, T. J. 2014. Grafix: A semiautomatic approach for parsing low-and high-quality eye-tracking data. *Behavior research methods 47*, 1, 53–72.

ESTER, M., KRIEGEL, H. P., SANDER, J., AND XU, X. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. *Kdd 96*, 34 (August), 226–231.

KOMOGORTSEV, O. V., GOBERT, D. V., JAYARATHNA, S., KOH, D. H., AND GOWDA, S. M. 2010. Standardization of automated analyses of oculomotor fixation and saccadic behaviors. *Biomedical Engineering, IEEE Transactions 57*, 11, 2634–2645.

KRAUZLIS, R., AND MILES, F. 1996. Initiation of saccades during fixation or pursuit: evidence in humans for a single mechanism. *Journal of Neurophysiology 76*, 6, 4175–4179.

LIVERSEDGE, S. P., AND FINDLAY, J. M. 2000. Saccadic eye movements and cognition. *Trends in cog. sciences 4*, 1, 6 – 14.

SALVUCCI, D. D., AND GOLDBERG, J. H. 2000. Identifying fixations and saccades in eye-tracking protocols. *Proceedings of the 2000 symposium on Eye tracking research and applications. ACM.* (November), 71–78.

SEN, T., AND MEGAW, T. 1984. The effects of task variables and prolonged performance on saccadic eye movement parameters. *Advances in Psychology 22*, 103–111.

SHIC, F., SCASSELLATI, B., AND CHAWARSKA, K. 2008. The incomplete fixation measure. *ETRA 2008. ACM.*, 111–114.

SUN, W., YANG, Z., LI, S., XU, Y., GUO, M., AND X, W. 2015. Driver fixation area division oriented dbscan-mmc method. *Journal of Zhejiang University (Engineering Science)* (August).