



Contents lists available at ScienceDirect

Omega

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

Outlier-Aware, density-Based gaze fixation identification[☆]

Wen Liu^{a,*}, Andrew C. Trapp^{a,b}, Soussan Djamasbi^b

^a Data Science Program, Worcester Polytechnic Institute, 100 Institute Road, United States

^b Robert A. Foisie School of Business Worcester Polytechnic Institute 100 Institute Road, Worcester, MA, United States

ARTICLE INFO

Article history:

Received 15 May 2019

Accepted 22 June 2020

Available online xxx

Keywords:

Eye tracking

Fixation identification

Outlier detection

Fixation inner-Density

Mixed-Integer optimization

Time series

ABSTRACT

Eye tracking is an advancing technology holding significant promise to improve our understanding of human behavior and decision making. Gaze data gathered by eye trackers contain events known as *fixations*. Fixations indicate visual attention and awareness, and are identified by algorithms that parse eye-tracking data into a sequence of gaze point clusters. While great potential exists, eye-tracker imprecision often results in noisy gaze data, such as what arises from calibration errors, erratic eye movements, or other system noise. Noise can cause inaccurate identification of fixations in eye-tracking applications, resulting in misleading behavioral interpretations and conclusions. Therefore, fixation identification algorithms should be robust against data noise. To resolve such inaccuracies, we propose FID⁺: outlier-aware fixation identification via fixation inner-density. We represent the problem of detecting outliers in fixation gaze data through a novel mixed-integer optimization formulation, and subsequently strengthen the formulation using two geometric arguments to provide enhanced bounds. We show that neither bound dominates the other, and that both are effective in reducing the overall solution runtime. Our experiments on real gaze recordings demonstrate that accommodating for the reality of fixation outliers enhances the ability to identify fixations with greater density in reasonable runtime.

© 2020 Elsevier Ltd. All rights reserved.

1. Introduction

Eye-tracking technologies are an increasingly powerful tool for analyzing human behavior and visual attention patterns. An eye-tracking device provides objective, quantitative data concerning human gaze, which can be used to analyze focus of attention and awareness under variable visual stimuli. Eye-trackers can be readily attached to computer devices. Fig. 1 depicts such a screen-based configuration. The eye tracker uses infrared light illuminators and cameras to identify light source reflection patterns on the eyes of a user. The captured patterns are used to algorithmically estimate a consecutive stream of (x, y) gaze point positions on the computer monitor.

The proliferation of eye-tracking devices on personal computers [1] offers great potential in many practical applications, such as analysis of user experience [2–4] and enhancement of multimedia learning experience [5]. In management science research, many studies analyze customer decision-making via visual attention information collected by eye-tracking devices. Eye-tracking technology is used for learning information acquisition patterns in cus-

tom shopping environments [6], for studying the efficiency of decision processes in conjoint choices [7], and for the evaluation of behavior attention in retail category management [8,9]. In health-care studies, eye-tracking technology has been employed for researching human cognition and decision making [10], experimental psychology studies [11] and attentional neuroscience [12] investigations. Eye-tracking technology has particularly prominent uses as a supportive diagnostic tool for monitoring vision health [13] and mental health [14]; Augmentative and Alternative Communication (AAC) devices commonly adapt eye tracking technologies to substitute for more traditional human computer interaction tools such as keyboard and touch screen. AAC assists individuals with disabilities like autism [15–17], muscular dystrophy [18,19], and cerebral palsy [20] to more easily use technology.

The foundation of all of the aforementioned eye-tracking applications is a system that can accurately process gaze data and correctly identify human visual attention. For superior performance, such systems require both high-quality gaze data, as well as efficient and effective translation of raw gaze data into behavioral indicators.

While high-quality gaze data is a prerequisite for information acquisition among all eye-tracking recordings, a variety of factors in real-world settings can adversely affect gaze data quality. These include system issues such as sensor noise and data loss from the eye tracker [21], calibration errors prior to the start of an experi-

[☆] This manuscript was processed by Associate Editor Dmytro Matsypura.

* Corresponding author.

E-mail addresses: wliu3@wpi.edu (W. Liu), atrapp@wpi.edu (A.C. Trapp), djamasbi@wpi.edu (S. Djamasbi).

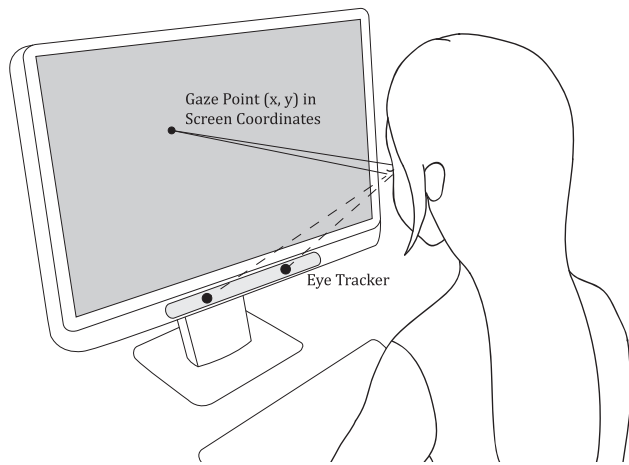


Fig. 1. An eye-tracking device mounted to a computer monitor, recording eye movement positions over time.

ment [22], gaze data processing algorithms [22], participant characteristics [22], eye-tracking experiments design [23], and poor recording environments and low-skilled operators [22]. The reality of eye movement mismeasurement and data noise in eye tracking recordings ensures that *outliers* exist in gaze data. While outliers have been studied in a variety of settings [24–30], when left unremedied in accuracy-dependent contexts, outliers can distort downstream processing and analysis, ultimately leading to inaccurate and less useful research.

Technically speaking, gaze data is categorized into two primary types: *fixations* are clusters of points that are adjacent in proximity and time, whereas *saccades* are higher velocity gaze points that occur between fixations. Because fixations represent visual attention, the accurate classification of eye gaze data into its constituent categories is a must for researchers to precisely understand focus of attention in meta-analysis, which is the most critical issue in eye-tracking research and development. The process of categorizing fixation and saccade eye movements is known as *fixation identification* [31–33] or *event detection* [34,35]. While the velocity-based I-VT filter [31] and the dispersion-based I-DT filter [31] serve as two foundations upon which many fixation identification methods are built, each suffers from limited precision that skews fixation properties [34,35] and hinders downstream research that relies on these essential properties.

Trapp et al. [36] advance the state-of-the-art in fixation identification through the notion of fixation *inner-density*, which addresses some limitations of existing methods including a lack of sensitivity to peripheral fixation points, as well as possible misrepresentation of fixation properties. They introduce the FID filter [37] which uses integer optimization techniques to identify fixations in a sequence of gaze points by optimizing for inner-density. The benefits of the FID filter can be seen in Fig. 2, where it can eliminate extraneous gaze points #1 and #9 that are at the boundaries of the fixation – technically under the velocity threshold, but likely not belonging to the fixation. Computational results demonstrated that the FID filter is efficient and effective in identifying denser fixations than the current I-VT method.

There are opportunities to improve the FID filter, especially its sensitivity to handle occasional noise and erratic eye movements within gaze data. The optimization model in [36] enforces that within a single fixation, all fixation points must be temporally adjacent; this can result in overly strict interpretations of fixations, whereby some small aberrations which should be otherwise ignored, may force fixations to terminate early. Thus, it is worthwhile to allow for some small deviations in the sequence, for ex-

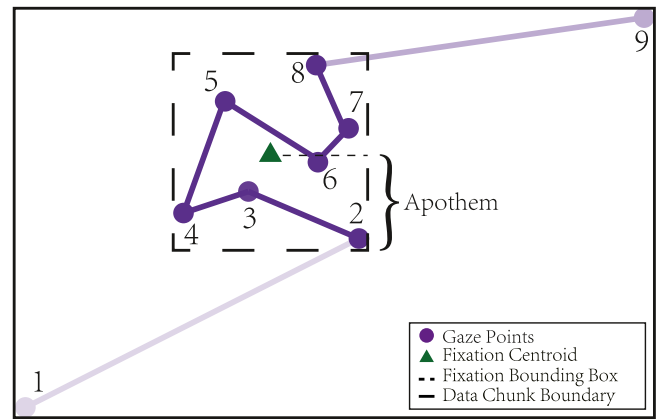


Fig. 2. Illustration of fixation and its apothem (side half-length) identified in a gaze data chunk; FID filter: minimizing apothem of fixation bounding box.

ample if a stray gaze point exists between two larger clusters of gaze points in the same region. In this case, it may be preferable to allow for the facility to simply omit this gaze point.

We contribute to the eye-tracking fixation identification literature by creating the first density-based method for detecting fixations that is outlier-aware. The FID filter is introduced in [36] and employs optimization-based approaches to find the densest fixations, but is otherwise silent with respect to outliers. The work presented in this study augments the FID filter by enabling the detection and elimination of certain outlier points within the fixation. Our work can significantly improve results in identifying fixations within noisy gaze data. This is particularly important for eye-tracking experiments where the understanding of human visual attention is of central importance, such as healthcare applications.

We propose an enhanced mathematical optimization formulation – FID⁺ – to account for this *outlier sensitivity*. To the best of our knowledge, this paper and [36] are the only approaches to identify fixations in gaze data by optimizing for density. The addition of a new set of budget-constrained binary variables accounts for the condition of where a gaze point is labeled as an outlier. In conjunction with the existing binary variables that indicate whether a gaze point is labeled as a fixation point, we introduce two new constraint sets that together represent time consistency in light of outlier gaze points. While the new formulation accurately remedies the aforementioned limitation, it does so at the cost of additional complexity. Thus, we present two algorithmic techniques to tighten lower bounds on the size of the apothem (which is minimized) to improve the computational performance.

The remainder of this paper is organized in the following manner. In Section 2 we provide background on fixation identification algorithms for analyzing eye-tracking data, including classical methods, as well as the more recent FID filter. In Section 3 we present FID⁺, a novel mixed integer programming (MIP) formulation for detecting fixations with outlier sensitivity. We subsequently provide two geometric arguments to strengthen the optimization formulation by enhancing the lower bounds on the apothem of the bounding box, and demonstrate that both are advantageous (we show that neither technique dominates the other). Section 4 details the computational experiments on real eye-tracking data, including a discussion on its observed performance. Finally, we conclude the paper and discuss future work in Section 5.

2. Background on eye-Tracking technologies

Gaze data is recorded as a sequence of (x, y, t) triplets, often referred as the point of regard (POR) in eye-tracking literature, where

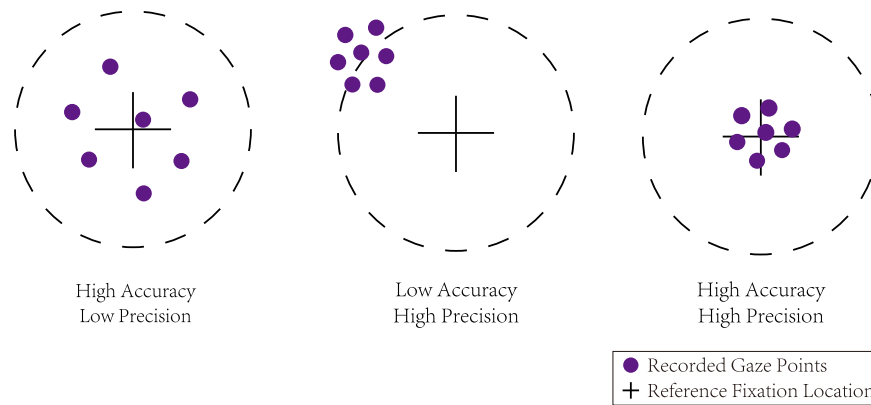


Fig. 3. Illustration of accuracy and precision for measuring gaze data quality, reproduced from [39]. Accuracy is the difference between the centroid of grouped recorded gaze points, and an actual reference fixation location. Precision is the variance of the gaze point dispersion in a fixation.

(x , y) attributes represent eye movement position on 2D stimuli, such as static computer displays. The third attribute t is the timestamp that represents when the corresponding position is recorded by the eye-tracking device. The sampling rate of commercial eye-tracking devices commonly ranges from 30 Hz to 1,000 Hz, or even higher.

Eye movements can be separated into two common types of events: fixations and saccades. The purpose of eye movement classification is to isolate eye movements within the gaze data stream into distinct time intervals that correspond to oculomotor responses or cognitive properties towards visual stimuli [35]. Fixations are clusters of gaze points that occur near in both time and location. This is because the act of fixating maintains visual gaze on a single location while cognitive processing occurs. On the other hand, saccades are the rapid movements between fixations. Notably distinct from saccades are *smooth pursuit* [38] eye movements, which allow the eyes to follow a moving visual stimulus. Similar to many of the current classification algorithms that identify fixations and saccades [35], we also limit our discussion to fixation identification, that is, gaze points that are not fixations are not further classified as saccades or smooth pursuits.

The stability of fixation identification is highly influenced by gaze data quality, which has long been discussed in eye-tracking research. We now review the key aspects of data quality and the actual impacts for fixation metrics.

2.1. Data quality and fixation outliers

High-quality gaze data is the foundation of generating valid and reproducible behavioral research results. As illustrated in Fig. 3, *Accuracy* and *precision* are the two highlighted aspects measured for eye-tracking data quality. The reference location, denoted with a “+”, is where the participant is asked to fixate. *Accuracy*, also called offset, refers to the shift between the recorded gaze position location, and the actual reference location. *Precision* refers to the variance of the recorded positions to the reference location [22,29,39].

Inaccuracy and imprecision can be attributed to multiple factors: eye-tracking cameras [39], algorithms for capturing eye movements [39], experimental design [23], system issues (such as sensor noise, data loss) [21], and various participant characteristics (such as glasses, astigmatism, eye color, head movements) [39]. Poor data precision leads to noisy gaze samples, which can challenge the reliability of fixation identification algorithms.

Fig. 4(a) illustrates a raw gaze sequence with 425 points collected by a Tobii Pro-TX300 [40] eye-tracking device, while Fig. 4(b) shows a noisy raw gaze sequence with the same length also from the same device. Gaze points in Fig. 4(a) show ex-

plicit clusters at the location of fixations. However, the clusters in Fig. 4(b) contain multiple stray points, and those points appear to drift to the same direction from their temporally adjacent points. The fixation patterns in Fig. 4(b) will inevitably contain some noise points in a long fixation gaze point sequence. Such noise points should be viewed as *Fixation Outliers*, and subsequently be eliminated from fixations.

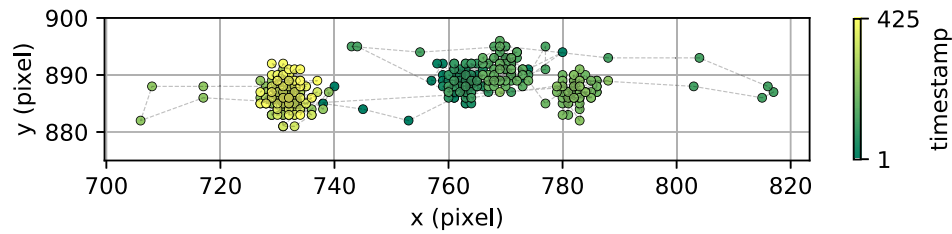
Fixation outliers can have substantial effects on the precision of fixation metrics, such as the number, and duration, of fixations [22]. Also impacted is *dwelt time*, a commonly used measurement of gaze duration in eye-tracking research for entering and remaining in an area of interest [41]. As illustrated in Fig. 5(a), when the point C is included as a fixation point, the square fixation bounding region increases significantly and the fixation centroid shifts away from its original position. Fig. 5(b) shows an actual example of possible fixation outliers appearing in real gaze data.

2.2. Common algorithms for fixation identification

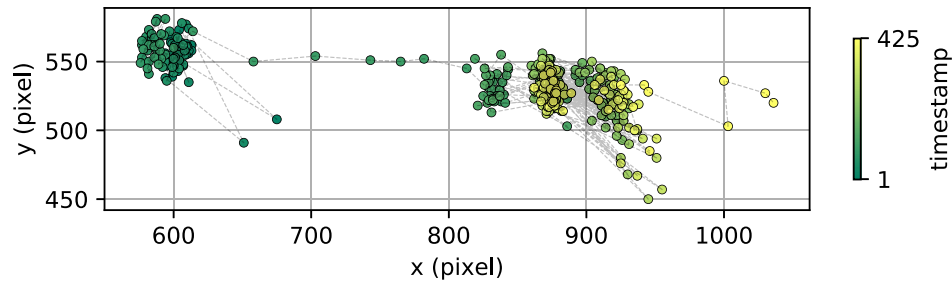
Fixation identification is closely related to cluster analysis. Because there is inherent ambiguity in assessing the quality of grouped objects, formal evaluation of fixation identification algorithms is challenging and lacks standardization. While it is commonly agreed upon that all existing algorithms for event detection have limitations [35], it is valuable to examine those that exist because they form the foundation of the state-of-the-art and offer insights into how to approach solving the fixation identification problem. In particular, a recently developed algorithm known as the *fixation identification (FID) filter* was the first to incorporate an optimization-based approach to identify fixations, optimizing for *fixation inner-density* [36]. We now review key existing methods.

2.2.1. Velocity-based algorithms

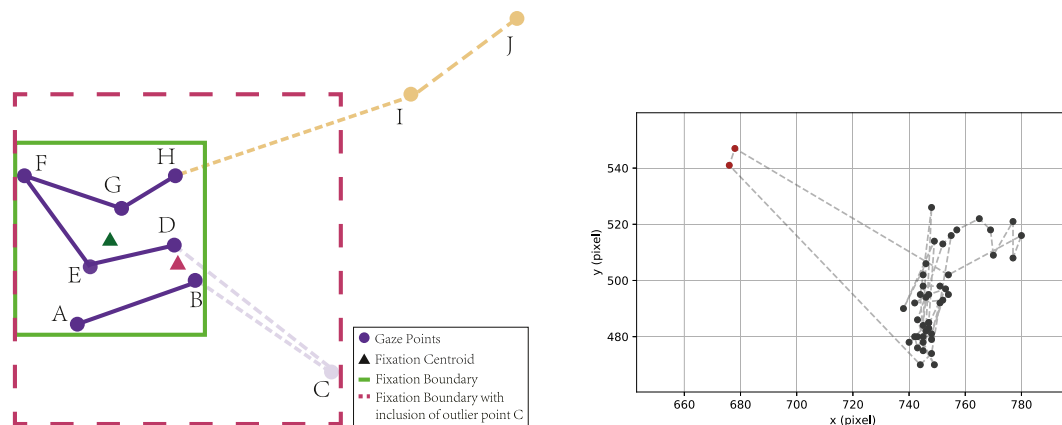
In velocity-based algorithms for fixation identification, the classical approach is the *Identification by Velocity-Threshold* (I-VT) filter [31]. This algorithm sequentially separates gaze points into fixations and saccades based on point-to-point velocity. Points with velocity exceeding that of a predefined velocity threshold V are categorized as saccade points. This process naturally separates gaze points into distinct fixations. This algorithm is fairly accurate in saccade detection, easy to implement, and robust for a variety of practical uses for eye-tracking devices. However, a significant drawback is that the I-VT filter may result in misclassifying gaze points that, while having a velocity technically below the threshold, are locationally separate from adjacent gaze points. This shortcoming can skew fixation metrics such as fixation centroid location, which is an important representation of visual location for



(a) Well-calibrated gaze data in two dimensions recorded by eye-tracking device.



(b) Noisy raw gaze data in two dimensions recorded by eye-tracking device.

Fig. 4. Comparison between normal gaze data and noisy data.

(a) When excluding fixation outlier point C, we get a tighter, denser fixation, with improved metrics: the center shifts from the red triangle to the green; the density increases as the bounding region decreases.

(b) Example in raw gaze data: the intermediate red gaze points (top left) are far from the main cluster of gaze points, indicating the potential to be fixation outliers.

Fig. 5. Influence of fixation outliers on fixation metrics.

user attention in behavioral studies. Another drawback is that constant velocity thresholds are not suitable for gaze stream data with substantial noise. Some recent studies [34,42–44] enhance the basic I-VT filter by designing an adaptive velocity threshold that provides greater flexibility with event classification at different noise levels. Even so, there exists an inherent challenge: the I-VT filter does not consider the compactness of constituent gaze points.

2.2.2. Dispersion-based algorithms

The *Identification by Dispersion-Threshold* (I-DT) filter [31] is a classical dispersion-based method. The I-DT filter identifies fixations using two predefined thresholds: the minimum fixation duration, and the maximum fixation dispersion threshold D . It uses a fixed-size sliding window to sequentially examine data. To constitute a fixation, the length of the gaze sequence should meet or

exceed the minimum duration, while its dispersion should not exceed D . The dispersion of the gaze sequence is measured using gaze point location. One implementation is to set a threshold for fixation radius. It also can be defined as a dispersion threshold D that equals to the sum of the length and width of the window covering a minimum amount of consecutive points. The main limitation of the I-DT filter is that D is a constant parameter, which may result in misclassifying gaze points and a lack of sensitivity in actual implementation. Some research has been done to further investigate dispersion-based algorithms. Blignaut [32] suggested that the correct setting of dispersion threshold for fixation radius was found in the range of 0.7° to 1.3° . Veneri et al. [33] propose an algorithm with improved dispersion criterion that is based on the analysis of fixation variance using covariance thresholds and F -tests.

2.2.3. Density-based algorithms

One recent study that identifies fixations by density-based clustering is the modified DBSCAN algorithm proposed in [45]. Traditional DBSCAN requires two parameters: the minimum distance ϵ between two points, and the minimum number of points $minPts$ to form a dense region. It then categorizes points into *core points*, *border points*, and others (which are known as noise points). The modified DBSCAN algorithm in [45] adds an additional requirement while evaluating the number of points within the dense region: the points within distance ϵ should be temporally adjacent. In consequence, the core points and border points constitute the fixations, whereas the other points are classified as saccades.

2.2.4. Fixation inner-Density-based algorithms

Trapp et al. [36] introduced a new fixation identification method known as fixation inner-density (FID). It combines both temporal and spatial aspects of the fixation. Together, these aspects are used to evaluate the compactness of a fixation, which has been shown to be positively correlated with user attention [46]. Inner-density overcomes several limitations of existing methods, such as a lack of sensitivity to peripheral points of a fixation, as well as the misrepresentation of fixation properties. The FID filter inherently differs from [45] in two aspects: methodology and the interpretation of density. Two mixed-integer optimization approaches were developed to identify fixations in a sequence of gaze points by optimizing for inner-density. The key novelty is the guarantee that there is no better gaze point identification according to the objective function of optimizing for inner-density, modulo the parameter α . This parameter is a predetermined value (e.g., via expert judgment) that enables decision-makers to have fine-tuned control over the inner-density.

Simultaneously identifying all fixations in the entire gaze stream is computationally prohibitive. We exploit the fact that saccades are natural separators of fixation to decompose the entire gaze stream into a series of data chunks for efficient processing. This decomposition principle, together with the optimization approach applied over all chunks, constitutes the FID filter. The experimental results on real datasets demonstrate that the FID filter with optimization formulation (13a)–(13f) in [36] is efficient and effective, averaging under one second per chunk to identify the α -densest fixation among the constituent gaze points. The identified fixations exhibit greater density than the existing I-VT filter, reflecting the ability to refine fixations, as well as more accurately represent gaze metrics such as fixation duration and center. The improved gaze metrics can form a more precise representation of attention and awareness for further analysis in eye-tracking studies.

While we have addressed the benefits of eliminating fixation outliers, such as illustrated in Fig. 5, the FID filter is limited in its ability to account for fixation outliers due to the overly strict nature of the constraint set outlined in Proposition 1 of Section 3.2.2 of [36] that requires every fixation to contain only consecutive gaze points in time. Therefore, to enable the FID filter to account for outlier sensitivity, we extend the approach in [36].

3. Mathematical developments

From a gaze sequence $\mathcal{S}$ with $\mathcal{T}$ points (x^t, y^t) , $t = 1, \dots, \mathcal{T}$, we seek to identify fixation points to constitute $\mathcal{F}$ fixations. The fixation identification problem discussed in [36] requires each fixation to contain at least $\mathcal{N}$ points for information processing to occur, and those points must be temporally adjacent. Define $\mathcal{TF}$ binary variables z , with $z_{tf} = 1$ if gaze point t is included in fixation f , and 0 otherwise. Of the two formulations presented in [36] for FID filter in finding dense fixations, we focus on the latter, *Minimize*

Square Area of Fixations [36, formulation (13a) – (13f)]. The formulation bounds each fixation with a two-dimensional square box of minimal area; it achieves a minimum area by equivalently minimizing the apothem of the square, r_f . The model incorporates a non-negative parameter α into the objective function that balances the trade-off between the inclusion of additional gaze points and the compactness of the fixation region. For the sake of completeness, we include this formulation in (1a)–(1h).

$$\text{minimize } \sum_{f=1}^{\mathcal{F}} \left[r_f + \alpha \sum_{t=1}^{\mathcal{T}} (1 - z_{tf}) \right], \quad (1a)$$

$$\text{subject to } \sum_{f=1}^{\mathcal{F}} z_{tf} \leq 1, \quad t = 1, \dots, \mathcal{T}, \quad (1b)$$

$$\sum_{t=1}^{\mathcal{T}} z_{tf} \geq \mathcal{N}, \quad f = 1, \dots, \mathcal{F}, \quad (1c)$$

$$\sum_{j=t+1}^{\mathcal{T}} z_{jf} \leq (\mathcal{T} - t)(1 - z_{tf} + z_{t+1,f}), \quad t = 1, \dots, \mathcal{T} - 1; f = 1, \dots, \mathcal{F}, \quad (1d)$$

$$x_f - r_f - \mathcal{M}_x(1 - z_{tf}) \leq x^t \leq x_f + r_f + \mathcal{M}_x(1 - z_{tf}), \quad t = 1, \dots, \mathcal{T}, \quad (1e)$$

$$y_f - r_f - \mathcal{M}_y(1 - z_{tf}) \leq y^t \leq y_f + r_f + \mathcal{M}_y(1 - z_{tf}), \quad t = 1, \dots, \mathcal{T}, \quad (1f)$$

$$l_x \leq x_f \leq u_x, \quad l_y \leq y_f \leq u_y, \quad f = 1, \dots, \mathcal{F}, \quad (1g)$$

$$r_f \geq 0, x_f \geq 0, y_f \geq 0, \quad f = 1, \dots, \mathcal{F}; \quad z_{tf} \in \{0, 1\}, \quad t = 1, \dots, \mathcal{T}, \quad f = 1, \dots, \mathcal{F}. \quad (1h)$$

Objective function (1a) contains two terms, the first minimizes the sum of apothems, and the second provides incentive to label additional points as fixation points. Constraint set (1b) represents that a point can be assigned to at most one fixation. Constraint set (1c) ensures that each fixation contains at least $\mathcal{N}$ points. Constraint set (1d) ensures gaze points identified in one fixation are temporally adjacent. Constraint sets (1e)–(1f) are box constraints to guarantee that when time point t is assigned to fixation f , it lies in the square with center (x_f, y_f) and apothem r_f . Bounds for x_f and y_f are $l_x = \min_{t=1, \dots, \mathcal{T}} x^t$, $u_x = \max_{t=1, \dots, \mathcal{T}} x^t$, $l_y = \min_{t=1, \dots, \mathcal{T}} y^t$, and $u_y = \max_{t=1, \dots, \mathcal{T}} y^t$. Then, the values of $\mathcal{M}_x$ and $\mathcal{M}_y$ are calculated by $\mathcal{M}_x = \max \{|x^t - l_x|, |u_x - x^t|\}$ and $\mathcal{M}_y = \max \{|y^t - l_y|, |u_y - y^t|\}$. Variable definitions and bounds are listed in (1g)–(1h).

3.1. Decomposition principle

The gaze sequence length $\mathcal{T}$ can easily reach the hundreds of thousands gaze points, and the number of fixations can likewise be in the thousands. Formulation (1a)–(1h) is valid for any number of gaze points $\mathcal{T}$ and fixations $\mathcal{F}$. This includes subsequences obtained after applying the decomposition principle discussed in [36]. This process separates a gaze data sequence into distinct data chunks $\mathcal{C}^k$, $k = 1, \dots, \mathcal{K}$, with data chunk separated by one or more saccade points as identified by benchmark filters such as the I-VT filter. After the decomposition, a minimal number of fixations remain within each data chunk, and formulation (1a)–(1h) can identify α -densest fixations efficiently in each chunk. Again, we term this approach the *FID filter*. We also apply this decomposition principle in the FID⁺ filter.

3.2. FID⁺ Filter: Detecting fixation outliers in gaze data

In this section we present the insights for extending the mathematical formulation to identify fixations with outlier sensitivity.

3.2.1. New variables for outlier detection

We extend formulation (1a)–(1h) to additionally classify a small portion of gaze points within the identified fixations as *fixation outliers*. Although they lie in the interior of a fixation time sequence, they are not identified as fixation points (i.e., $z_{tf} = 1$). Define $\mathcal{TF}$ binary variables q , with $q_{tf} = 1$ if gaze point t is an outlier in fixation f , and 0 otherwise.

3.2.2. Fixation outlier budget

We propose a budget $\mathcal{P}$ to allow some small number of outlier points. One reasonable value for $\mathcal{P}$ is a percent p of the total number of gaze points $\mathcal{T}$ in the chunk, so that $\mathcal{P} = \lceil p\mathcal{T} \rceil$. Hence, the sum of outlier points over all fixations should be less or equal to $\mathcal{P}$:

$$\sum_{f=1}^{\mathcal{F}} \sum_{t=1}^{\mathcal{T}} q_{tf} \leq \mathcal{P}. \quad (2)$$

Alternatively, $\mathcal{P}$ can be set to any user-defined, positive integer.

3.2.3. Relaxation from absolute time consistency

Proposition 1 in [36, Section 3.2.2] introduces the following constraint set:

$$\sum_{j=t+1}^{\mathcal{T}} z_{jf} \leq (\mathcal{T}-t)(1-z_{tf}+z_{t+1,f}), \quad t = 1, \dots, \mathcal{T}-1; \quad f = 1, \dots, \mathcal{F}. \quad (3)$$

This constraint set ensures the included points within each fixation must be consecutive in time. Fixation f terminates once a consecutive time pair $(z_{tf}, z_{t+1,f})$ appears as (1,0) among all the possible values $\{(0,0), (0,1), (1,1), (1,0)\}$. When $(z_{tf}, z_{t+1,f})$ equals to (1,0), the right-hand side becomes zero, ensuring that $z_{jf} = 0$, for all $j : t+1 \leq j \leq \mathcal{T}$. It guarantees that the remainder of the points in the chunk are not included in this fixation. For the other possible values of $(z_{tf}, z_{t+1,f})$, the right-hand side is either $(\mathcal{T}-t)$ or $2(\mathcal{T}-t)$, so the constraint set becomes vacuous. Thus, for a fixation f , a starting gaze point at time a and an ending point at time b , constraint set (3) ensures z_{tf} is assigned in the following fashion:

$$\text{i) } z_{tf} = 0, \quad \text{ii) } z_{tf} = 1.$$

However, when a set of outlier gaze points $\mathcal{E} \subset \{a+1, \dots, b-1\}$ appears between the starting and ending fixation points, as indicated by $q_{tf} = 1$, the corresponding z_{tf} should be assigned to zero. The assignment ii) changes to $z_{tf} = 0$ and

$z_{tf} = 1$. The consecutive pair $(z_{tf}, z_{t+1,f})$ equals to (1,0) not only happens at the termination of f , but can also occur when point $t+1$ is identified as an outlier, i.e., $q_{t+1,f} = 1$. When fixation f terminates, $(z_{tf}, z_{t+1,f})$ is (1,0) and $q_{t+1,f}$ should be assigned as zero. Following this interpretation, we extend the constraint set from (3) to (4) by relaxing the assumption that fixation points must be consecutive in time:

$$\sum_{j=t+1}^{\mathcal{T}} z_{jf} \leq (\mathcal{T}-t)(1-z_{tf}+z_{t+1,f}+q_{t+1,f}), \quad t = 1, \dots, \mathcal{T}-1; \quad f = 1, \dots, \mathcal{F}. \quad (4)$$

When $q_{t+1,f} = 0$, indicating point $t+1$ is not an outlier for fixation f , the right-hand side in (4) equals zero when consecutive

time pair $(z_{tf}, z_{t+1,f})$ equals (1,0). Thereby it ensures the following variable z_{jf} , for all $j : t+1 \leq j \leq \mathcal{T}$ must be zero, which means fixation f terminates as it may no longer include any gaze points. Therefore, when $q_{t+1,f} = 0$, the constraint set has the same impact as constraint set (3). However when $q_{t+1,f} = 1$, the constraint set induces no restrictions under any alternatives of $(z_{tf}, z_{t+1,f})$, because the right-hand side is always at least $(\mathcal{T}-t)$. Thus, the consecutive variables z_{jf} , for all $j : t+1 \leq j \leq \mathcal{T}$ may still be assigned to one. Therefore, the subsequent gaze points from $t+1$ to $\mathcal{T}$ can be included in fixation f and the assignment of (1, 0) to the pair $(z_{tf}, z_{t+1,f})$ no longer delineates the end of the fixation.

3.2.4. Controlling the position of outliers

While constraint set (4) generalizes the condition of strict time consistency, there is no implication on the values that points z_{jf} , for all $j : t+1 \leq j \leq \mathcal{T}$ can take when $q_{t+1,f} = 1$. In the absence of any other constraints, this may cause a fixation to be decomposed into multiple components. To ensure that every fixation f has consecutive gaze points formed by only fixation points ($z_{tf} = 1$) and outlier points ($q_{tf} = 1$), the following set of constraints can be incorporated:

$$q_{tf} \leq q_{t+1,f} + z_{t+1,f}, \quad t = 1, \dots, \mathcal{T}-1, \quad f = 1, \dots, \mathcal{F}. \quad (5)$$

Constraint set (5) ensures that if $q_{tf} = 1$, the next gaze point at $t+1$ must be classified as a fixation point ($z_{t+1,f} = 1$) or a fixation outlier ($q_{t+1,f} = 1$). When $q_{tf} = 0$, the constraint is always valid. While this constraint set technically allows both $z_{t+1,f} = 1$ and $q_{t+1,f} = 1$, there are scarce outlier points available by (2), and so gaze points are classified as outliers only when it is beneficial for the objective, that is, when subsequent gaze points are classified as fixation points. Constraint set (5) introduces $\mathcal{TF} - \mathcal{F}$ additional constraints.

3.3. Minimizing square area of fixations with outlier sensitivity

We now present the final MIP formulation for FID⁺: outlier-aware fixation identification via density optimization. Note that the extensions discussed in Section 3.2 can also be applied to *Minimize Average Intra-Fixation Sum of Distances* [36, formulation (12a) – (12f)].

$$\text{minimize } \sum_{f=1}^{\mathcal{F}} \left[r_f + \alpha \sum_{t=1}^{\mathcal{T}} (1 - z_{tf}) \right], \quad (6a)$$

$$\text{subject to } \sum_{f=1}^{\mathcal{F}} z_{tf} \leq 1, \quad t = 1, \dots, \mathcal{T}, \quad (6b)$$

$$\sum_{t=1}^{\mathcal{T}} z_{tf} \geq \mathcal{N}, \quad f = 1, \dots, \mathcal{F}, \quad (6c)$$

$$\sum_{j=t+1}^{\mathcal{T}} z_{jf} \leq (\mathcal{T}-t)(1-z_{tf}+z_{t+1,f}+q_{t+1,f}), \quad t = 1, \dots, \mathcal{T}-1, \quad f = 1, \dots, \mathcal{F}, \quad (6d)$$

$$q_{tf} \leq q_{t+1,f} + z_{t+1,f}, \quad t = 1, \dots, \mathcal{T}-1, \quad f = 1, \dots, \mathcal{F}, \quad (6e)$$

$$\sum_{f=1}^{\mathcal{F}} \sum_{t=1}^{\mathcal{T}} q_{tf} \leq \mathcal{P}, \quad (6f)$$

$$x_f - r_f - \mathcal{M}_x(1 - z_{tf}) \leq x^t \leq x_f + r_f + \mathcal{M}_x(1 - z_{tf}), \quad t = 1, \dots, \mathcal{T}, \quad (6g)$$

$$y_f - r_f - \mathcal{M}_y(1 - z_{tf}) \leq y^t \leq y_f + r_f + \mathcal{M}_y(1 - z_{tf}), \quad t = 1, \dots, \mathcal{T}, \quad (6h)$$

$$r_f \geq 0, \quad l_x \leq x_f \leq u_x; \quad l_y \leq y_f \leq u_y, \quad f = 1, \dots, \mathcal{F}, \quad (6i)$$

$$z_{tf} \in \{0, 1\}, \quad q_{tf} \in \{0, 1\}, \quad t = 1, \dots, \mathcal{T}, \quad f = 1, \dots, \mathcal{F}. \quad (6j)$$

Formulation (6a)–(6j) uses binary variables z_{tf} to assign time point t to fixation f . It incorporates binary variables q_{tf} to identify outlier points in each fixation f . Objective function (6a) minimizes the sum of fixation square apothems, penalizing the number of excluded points with parameter α . Constraints (6b) and (6c) are the fundamental constraints indicating that a time point can be assigned to at most one fixation, and each fixation contains at least $\mathcal{N}$ points. Constraint set (6d) relaxes fixation point assignment from absolute time consistency, while constraint set (6e) ensures points identified as outlier points are succeeded by either outlier or fixation points. Constraint set (6f) ensures the number of identified outlier points is within the fixation outlier budget $\mathcal{P}$. Constraints (6g)–(6h) ensure that the identified points in fixation f present in the fixation bounding box with center (x_f, y_f) and apothem r_f . Variable definitions and bounds are listed in (6i)–(6j).

While formulation (6a)–(6j) is correct and detects fixation and outlier points, initial computational testing on larger instances revealed that, while strong feasible solutions were quickly found, the MIP solver Gurobi [47] experienced difficulty proving optimality.

3.4. Deriving lower bounds on r_f

Objective function (6a) minimizes the apothem r_f of the bounding box encompassing the fixation points. While feasible solutions to (6a)–(6j) representing strong upper bounds are quickly computed using the MIP solver Gurobi [47], the lower bounds often exhibit only gradual progress toward convergence, likely due to poor relaxation strength from constraint set (6d).

To accelerate the computational proof of optimality, we present geometric arguments that can strengthen lower bounds on r_f . We algorithmically preprocess the gaze point sequences to identify lower bounds ℓ on r_f , $f = 1, \dots, \mathcal{F}$.

3.4.1. Deriving lower bounds on r_f via sliding windows

Consider identifying $\mathcal{F}$ fixations from a gaze sequence with $\mathcal{T}$ total points, each of which requires at least $\mathcal{N}$ fixation points to ensure cognitive processing occurs [1]. Further, suppose the entire budget of $\mathcal{P}$ outlier points is used in a fixation with the minimum number of points $\mathcal{N}$. Lemma 1 states that there will be at least one subsequence separated by outlier points that contains at least $\lfloor \frac{\mathcal{N}}{\mathcal{P}+1} \rfloor$ consecutive gaze points.

Lemma 1. Suppose for fixation f , the fixation point sequence s_f has length $\mathcal{N}_f$, and it is decoupled into subsequences by $\mathcal{P}_f$ fixation outliers. There always exists a subsequence s of s_f with length of at least $\lfloor \frac{\mathcal{N}}{\mathcal{P}+1} \rfloor$ points.

Proof. The average length of all subsequences in fixation f is $\frac{\mathcal{N}_f}{\mathcal{P}_f+1}$, hence there is at least one subsequence s whose length is greater than or equal to $\frac{\mathcal{N}_f}{\mathcal{P}_f+1}$. Because $\mathcal{N}_f \geq \mathcal{N}$ and $\mathcal{P}_f \leq \mathcal{P}$ by (6c) and (6f), this implies $\frac{\mathcal{N}_f}{\mathcal{P}_f+1} \geq \frac{\mathcal{N}}{\mathcal{P}+1} \geq \lfloor \frac{\mathcal{N}}{\mathcal{P}+1} \rfloor$. Thereby the length of s is also greater than or equal to $\lfloor \frac{\mathcal{N}}{\mathcal{P}+1} \rfloor$. $\square$

For fixation f , the apothem r_f represents a minimum bounding box covering all included fixation points, starting from a gaze point at time a to an ending gaze point at time b . The apothem of the bounding box must satisfy $r_f \geq \frac{1}{2} \max_{i,j} \{|x^i - x^j|, |y^i - y^j|\}$ for all

the point pairs (i, j) : $a \leq i < j \leq b$. The apothem r_f of the bounding box is monotonically nondecreasing as the number of points in the range $[a, b]$ increases. Thus, a conservative global lower bound ℓ_1 on r_f can be derived from the individual lower bounds originating from the distance arising from t , to t shifted by the minimum number of consecutive gaze points, $\lfloor \frac{\mathcal{N}}{\mathcal{P}+1} \rfloor$. By considering all pairs of points $(t, t + \lfloor \frac{\mathcal{N}}{\mathcal{P}+1} \rfloor - 1)$ for $t = 1, \dots, \mathcal{T} - \lfloor \frac{\mathcal{N}}{\mathcal{P}+1} \rfloor + 1$, we obtain a lower bound on r_f . Finding ℓ_1 can be accomplished in polynomial time. For each begin-end point pair, we compute the corresponding minimum bounding length ℓ'_1 :

$$\ell'_1 = \frac{1}{2} \max_{i,j} \left\{ |x^i - x^j|, |y^i - y^j| \mid t \leq i < j \leq t + \left\lfloor \frac{\mathcal{N}}{\mathcal{P}+1} \right\rfloor - 1 \right\}. \quad (7)$$

When a smaller ℓ'_1 is found, we update ℓ_1 to be ℓ'_1 . The cost of this method is $\mathcal{O}(\mathcal{T} - \lfloor \frac{\mathcal{N}}{\mathcal{P}+1} \rfloor)$, that is, it is linear in the number of gaze points $\mathcal{T}$. This method is summarized in Algorithm 1.

Algorithm 1 Determine Valid Lower Bound ℓ_1 .

Input: Gaze sequence $\mathcal{S}$ with length $\mathcal{T}$; fixation outlier budget $\mathcal{P}$; minimum number of fixation points $\mathcal{N}$.

Output: Lower bound ℓ_1 on the fixation apothem r_f .

- 1: Set $\ell_1 \leftarrow \max \{|u_x - l_x|, |u_y - l_y|\}$.
- 2: **for** $t = 1, \dots, \mathcal{T} - \lfloor \frac{\mathcal{N}}{\mathcal{P}+1} \rfloor + 1$ **do**
- 3: Calculate the minimum bounding length
 $\ell'_1 = \frac{1}{2} \max_{i,j} \left\{ |x^i - x^j|, |y^i - y^j| \mid t \leq i < j \leq t + \left\lfloor \frac{\mathcal{N}}{\mathcal{P}+1} \right\rfloor - 1 \right\}$.
- 4: **if** $\ell'_1 < \ell_1$ **then**
- 5: Set $\ell_1 \leftarrow \ell'_1$.
- 6: **return** ℓ_1 .

Theorem 1. For a gaze sequence $\mathcal{S}$, ℓ_1 is a valid lower bound for r_f , $f = 1, \dots, \mathcal{F}$, i.e. $\ell_1 \leq r_f$.

Proof. Suppose there exists $\ell_1 > r_f$ for fixation f from Algorithm 1. By Lemma 1, we can find a subsequence s of fixation f with a length of at least $\lfloor \frac{\mathcal{N}}{\mathcal{P}+1} \rfloor$. We further truncate s by sequentially eliminating points from either the beginning or the end, until the remaining sequence s' is exactly $\lfloor \frac{\mathcal{N}}{\mathcal{P}+1} \rfloor$ points. The remaining sequence constitutes a new sequence s' , and let the apothem of the minimal bounding box be ℓ'_1 . Because s' is contained in s , it has fewer fixation points than fixation f . The lower bound on the bounding box apothem, by the construction in (7), is a nondecreasing function in the number of points in the fixation, thus we conclude that $\ell'_1 \leq r_f$. This implies that $\ell'_1 < \ell_1$, which contradicts the fact that ℓ_1 is the minimal bounding box apothem for all the consecutive gaze subsequences with length of $\lfloor \frac{\mathcal{N}}{\mathcal{P}+1} \rfloor$. Thus, the original statement holds. $\square$

3.4.2. Deriving lower bounds on r_f via smallest enclosing squares

For a gaze sequence of $\mathcal{T}$ points, the apothem length of the smallest enclosing square covering $\mathcal{N}$ points, irrespective of temporal adjacency, is a valid lower bound ℓ_2 for r_f , $f = 1, \dots, \mathcal{F}$. We adapt Algorithm 2 from [48] for finding the smallest square bounding box of $\mathcal{N}$ points for each input gaze sequence. Algorithm 2 first sorts the gaze points at x-decreasing order and sweeps each point. Hence, the algorithm sweeps points from right to left. When sweeping at point t , the current x^t is recorded as p_1 . From the points lying to the right of the vertical line drawn by p_1 , it finds a set of points V whose x-axis value is in the range of $[x_t, x_t + \ell_2]$, y-axis value is in the range of $[y_t - \ell_2, y_t + \ell_2]$, where ℓ_2 is the smallest apothem of the enclosing square identified thus far. It then finds the squares that exactly cover $\mathcal{N}$ points and their left side is on the vertical line through p_1 and bottom side is on the

Algorithm 2 Determine Valid Lower Bound ℓ_2 .

Input: Gaze sequence point set $\mathcal{S}$ with length $\mathcal{T}$; minimum number of gaze points $\mathcal{N}$.

Output: Lower bound ℓ_2 on the fixation apothem r_f

- 1: Sort points in $\mathcal{S}$ at x -decreasing order.
- 2: Set $\ell_2 \leftarrow \max\{|u_x - l_x|, |u_y - l_y|\}$.
- 3: Set $P \leftarrow$ empty balanced binary search tree.
- 4: **for** $t = 1, \dots, \mathcal{T}$ **do**
- 5: $p_1 = x^t$.
- 6: $xMax = x^t + \ell_2$
- 7: $yMax = y^t + \ell_2$.
- 8: $yMin = y^t - \ell_2$.
- 9: Insert a new node into P , key= y^t , value= (x^t, y^t) .
- 10: Set $V \leftarrow \emptyset$.
- 11: **for** node $p \in P$ **do**
- 12: **if** $x^p \leq xMax$ **then**
- 13: **if** $yMin \leq y^p \leq yMax$ **then**
- 14: Add (x^p, y^p) to V .
- 15: **else**
- 16: Delete p from P , i.e., $P = P \setminus p$.
- 17: **if** $|V| \geq \mathcal{N}$ **then**
- 18: Sort points in V at y -decreasing order.
- 19: Set $A \leftarrow$ empty balanced binary search tree.
- 20: Set $B \leftarrow$ empty balanced binary search tree.
- 21: **for** $i = 1, \dots, |V|$ **do**
- 22: Select $q = V[i] = (x^q, y^q)$ from V .
- 23: Set $q_2 = y^q$.
- 24: Insert a new node into A , key= x^q , value= (x^q, y^q) .
- 25: **for** node $a \in A$ **do**
- 26: **if** $y^a - q_2 > x^a - p_1$ **then**
- 27: Delete a from A , i.e., $A = A \setminus a$.
- 28: Insert a new node into B , key= y^a , value= (x^a, y^a) .
- 29: **if** $i \geq \mathcal{N}$ **then**
- 30: Find the key k at rank $\mathcal{N}$ in $(A - p_1) \cup (B - q_2)$.
- 31: $\ell'_2 = \frac{1}{2}k$.
- 32: **if** $\ell'_2 < \ell_2$ **then**
- 33: Set $\ell_2 \leftarrow \ell'_2$.
- 34: **return** ℓ_2 .

line through a point in V . At each p_1 , the algorithm sweeps a horizontal line q_2 from the top point to the bottom point of V . Two binary search trees A and B are maintained to store every point (x, y) above q_2 . If the horizontal distance $x - p_1$ is greater than the vertical distance $y - q_2$, the point is stored in A in increasing x -order. Otherwise it is stored in B in increasing y -order. For each q_2 , the element at rank k in the set $(A - p_1) \cup (B - q_2)$ is selected. This is the side length for a square that covers k points in the area from the top of V to q_2 . We compute ℓ'_2 as the half of the side length, and if $\ell'_2 < \ell_2$, we update ℓ_2 to be ℓ'_2 .

Theorem 2. For a gaze sequence $\mathcal{S}$, ℓ_2 is a valid lower bound for r_f , $f = 1, \dots, \mathcal{F}$, i.e. $\ell_2 \leq r_f$.

Proof. Consider the contrary, a fixation f has $\ell_2 > r_f$ by Algorithm 2. A different ℓ'_2 can be calculated by randomly choosing exactly $\mathcal{N}$ of the fixation points in f , as there are at least $\mathcal{N}$ fixation points in the box bounded by r_f . The enclosing square apothem can only decrease when reducing to $\mathcal{N}$ of the enclosed points. Hence, we can conclude that $\ell'_2 \leq r_f$. It suggests that these $\mathcal{N}$ points have a smaller bounding box apothem ℓ'_2 than ℓ_2 , which contradicts the fact that ℓ_2 is the apothem of the minimum bounding box covering $\mathcal{N}$ points in the given gaze data for fixation f . Hence, the original statement holds. $\square$

3.4.3. Comparison of two lower bounds

In this section, we discuss the relation between ℓ_1 and ℓ_2 and we find that neither bound dominates the other.

Proposition 1. Neither lower bound ℓ_1 or ℓ_2 dominates the other.

Example 1. Consider the examples of identifying one fixation in a gaze sequence with seven points, as depicted in Fig. 6. Supposing that $\mathcal{N}$ is four and the outlier budget $\mathcal{P}$ is one, ℓ_1 is determined by the x, y distances between $\lfloor \frac{\mathcal{N}}{\mathcal{P}+1} \rfloor = \lfloor \frac{4}{2} \rfloor = 2$ consecutive points, while ℓ_2 is the apothem of the smallest square bounding box covering $\mathcal{N} = 4$ points in the plane. The relationship of ℓ_1 and ℓ_2 varies based on the distribution of gaze points: (a) shows $\ell_1 < \ell_2$; (b) shows $\ell_1 = \ell_2$; and (c) shows $\ell_1 > \ell_2$.

4. Computational experiments

Formulation (6a)–(6j) with the decomposition principle described in Section 3.1 represents the FID⁺ filter, which extends the earlier FID filter of [36]. We now discuss our computational experiments using real eye-tracking data. We use a dataset obtained from the visual task of answering Graduate Record Examination (GRE) Math reading questions on a computer display [36], though we note that data from a variety of eye-tracking applications could be used to evaluate the FID⁺ filter, as outliers occur largely independent of the context. Algorithms 1 and 2 are introduced to derive lower bounds on r_f to improve the computational performance for solving the new formulation.

4.1. Experimental setup and data preprocessing

The GRE Math dataset contains ten recordings collected by a Tobii Pro-TX300 eye-tracking device at 300 Hz. Each recording is approximately five minutes in duration. Table 1 summarizes this dataset. We used the same data preprocessing strategy as discussed in [36, Section 4.2]. For each recording, we separate the data sequence $\mathcal{S}$ into chunks $\mathcal{C}^k$, $k = 1, \dots, \mathcal{K}_\ell$ using the Tobii Studio I-VT filter [49] with the default velocity threshold of $V = 30^\circ/\text{s}$. The minimum number of gaze points is set to $\mathcal{N} = 30$ (100ms), which is necessary for information processing to occur [50]. As shown in Table 1, this setting eliminates some data chunks and remain approximately 721 valid data chunks in each recording on average. We set $\mathcal{F}_{\min}^k = \mathcal{F}_{\max}^k = 1$ for formulation (6a)–(6j). The fixation outlier budget $\mathcal{P}$ is set as 1% of the total number of gaze points in each data chunk $\mathcal{C}^k$, that is, outlier budget $\mathcal{P}^k = \lceil 0.01 \cdot |\mathcal{C}^k| \rceil$. This value of $\mathcal{P}^k$ allows for at least one point per data chunk to be identified as a fixation outlier in formulation (6a)–(6j). As depicted in Fig. 7(a), the distribution of data chunks is long-tailed. Of the total 7208 data chunks with at least $\mathcal{N}$ points, there are 1860 data chunks having more than 100 points (25.8% of total), and 59 data chunks with length of greater than 500 points (0.8% of total). As the size of the data chunk increases, so does the expected computational effort in solving formulation (6a)–(6j). All computational experiments were conducted using an Intel core i7-6700MQ computer with 3.40 GHz and 16.0 GB RAM running 64-bit Windows 10. Gurobi Optimizer [47] with Python 2.7 was used for the optimization modeling, algorithm development and solution process. We used default parameter settings for seeking global optimality. We also set a time limit of one hour (wall-clock) for solving the optimization model for each data chunk. MATLAB 2016a [51] was used for additional data processing and analysis.

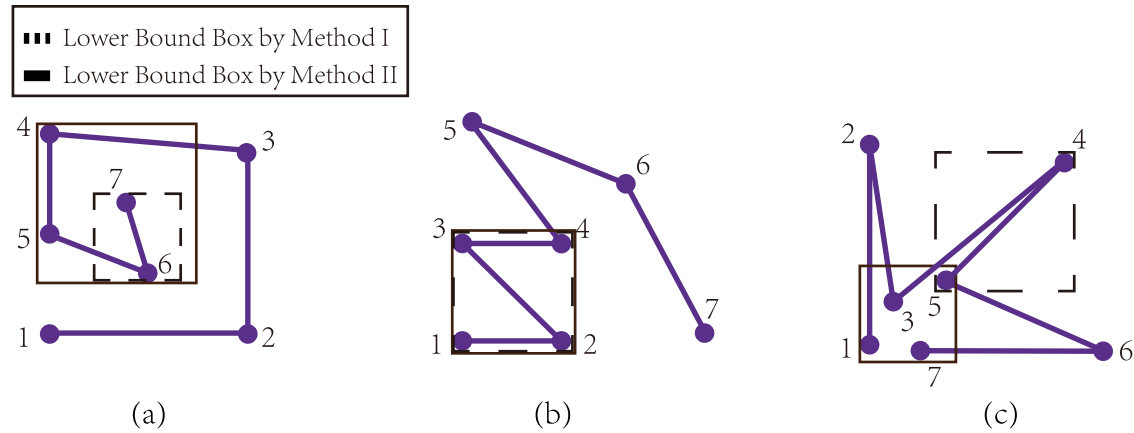
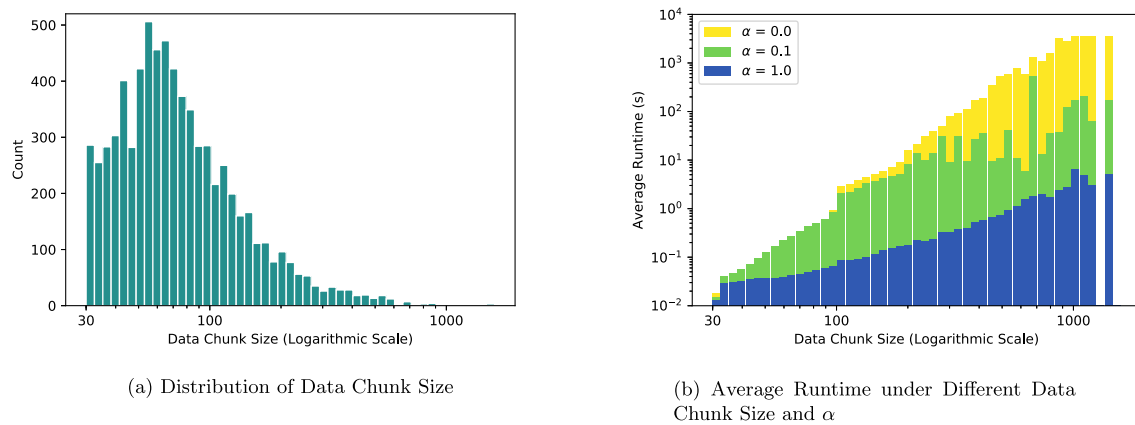
4.2. Computational results and discussion

Table 2 highlights the computational results of running the FID⁺ filter on the 300 Hz GRE Math reading dataset, as well as formulation (6a)–(6j) using lower bounds from Algorithms 1 and 2.

Table 1

Summary results on 300 Hz GRE Math Reading data with I-VT filter, averaged over ten recordings per dataset [36].

Stimuli	Avg # of All Points in Sequence	Avg # of Data Chunks	Avg # of Valid Data Chunks	Avg # of Points in All Data Chunks	Avg # of Points in Valid Data Chunks
GRE Math Reading Data	90,580	3612	721	80,956	66,677

**Fig. 6.** Comparison of lower bounding approaches. The gaze sequence length $\mathcal{T} = 7$, minimum number of covering points $\mathcal{N} = 4$, and outlier budget $\mathcal{P} = 1$. As shown in (a), (b) and (c), depending on how the points are distributed, the effectiveness of lower bounds ℓ_1 and ℓ_2 vary.**Fig. 7.** Depicting the distribution of data chunk size (left panel) and the average runtime using formulation (6a)–(6j) in each bin under $\alpha = 0, 0.1, 1$ (right panel). The right panel also shows that with the increase of α , the runtime decreases; with the increase of $|c^k|$, the runtime increases substantially, and becomes especially apparent when $|c^k|$ exceeds 100.**Table 2**Results of the FID⁺ filter, (6a)–(6j) with lower bound ℓ_1 , and (6a)–(6j) with lower bound ℓ_2 on 300 Hz GRE Math reading dataset. The entries in the evaluation metrics columns report the average metrics over all data chunks in each of the ten recordings; the entries in the runtime columns report the total runtime averaged over each each recording, containing approximately 721 data chunks.

α 300 Hz GRE Math Reading Data													
δ	Duration	Density Measures			Cover Rate	Center Shift	Budget Usage	Avg Runtime (s)		Avg Runtime (s) w/ ℓ_1		Avg Runtime (s) w/ ℓ_2	
		ρ_1^{avg}	ρ_2^{avg}	ρ_3^{avg}				Gurobi	Overall	Gurobi	Overall	Gurobi	Overall
0	0.1038	5.3981	81.2267	10.4529	0.2539	1.9494	0.91	12,548.8	12,647.9	10,624.0	10,724.8	10,729.2	10,969.9
0.1	0.2597	6.1667	231.2483	9.7190	0.6510	1.0814	0.87	1,898.2	2,006.3	1,637.5	1,742.5	1,560.6	1,802.7
0.2	0.2744	6.4515	259.4887	9.4059	0.6863	0.8331	0.82	315.5	430.6	242.3	345.5	249.3	493.9
0.3	0.2787	6.5700	268.5097	10.0599	0.6956	0.7397	0.75	186.3	305.6	145.9	253.6	150.9	398.1
0.4	0.2806	6.6383	273.4140	10.2574	0.6997	0.6916	0.74	147.1	267.0	116.3	226.0	115.6	363.2
0.5	0.2831	6.7417	279.7678	10.0763	0.7056	0.6213	0.45	128.8	247.6	96.3	205.4	99.7	347.4
0.6	0.2840	6.7941	282.5965	10.2516	0.7076	0.5861	0.41	108.9	225.9	82.3	191.0	83.9	331.7
0.7	0.2844	6.8136	283.7578	10.3622	0.7084	0.5750	0.41	97.7	214.7	71.8	181.6	73.0	320.3
0.8	0.2848	6.8364	284.9439	10.4752	0.7094	0.5603	0.39	88.4	205.7	63.1	172.5	64.5	311.6
0.9	0.2850	6.8465	285.3952	10.5318	0.7098	0.5541	0.38	80.3	197.3	56.9	164.8	58.4	306.7
1.0	0.2859	6.9006	288.2015	10.8704	0.7122	0.5151	0.24	73.4	190.7	51.2	158.4	52.9	303.1

Table 3

Results of the FID filter with formulation (1a)–(1h) on 300 Hz GRE Math reading dataset; reproduced from [36].

α	300 Hz GRE Math Reading Data							
	Duration	Density Measures			Cover Rate	Center Shift	Avg Runtime (s)	
		δ^{avg} (s)	ρ_1^{avg}	ρ_2^{avg}	ρ_3^{avg}			
0	0.1062	5.8589	90.1959	31.9361	0.2598	1.8150	574.3	659.5
0.1	0.2607	6.5335	241.3585	28.8872	0.6528	0.9478	364.5	454.1
0.2	0.2762	6.7828	268.4264	28.5850	0.6911	0.6739	264.7	354.7
0.3	0.2803	6.8764	277.5209	28.2034	0.7004	0.5727	207.2	299.7
0.4	0.2827	6.9654	283.6307	27.5299	0.7053	0.5046	154.7	246.6
0.5	0.2840	7.0202	287.1474	27.7181	0.7083	0.4589	119.0	212.0
0.6	0.2848	7.0571	289.3265	27.8777	0.7100	0.4300	87.0	178.1
0.7	0.2853	7.0816	290.6830	28.0161	0.7112	0.4095	67.1	159.0
0.8	0.2857	7.1100	292.1223	28.1589	0.7121	0.3880	53.9	145.1
0.9	0.2860	7.1251	292.7735	28.2548	0.7126	0.3777	43.4	136.5
1.0	0.2863	7.1483	294.0966	28.3347	0.7134	0.3612	37.7	128.8

The rows of Table 2 are indexed by parameter α , and the columns display the evaluation metrics, budget usage and runtime, and are to be compared with those of Table 3 which is reproduced from [36], depicting similar results without outlier detection. As in Table 3, the evaluation metrics are averaged over all data chunks in each of the ten data recordings. The evaluation metrics we consider are: fixation duration δ ; cover rate γ ; three fixation inner-density metrics: ρ_1 , ρ_2 , and ρ_3 ; and center shift λ .

The average fixation duration δ is the average number of fixation points in each fixation, divided by the sampling frequency. The cover rate γ measures the ratio of points recognized as fixations points, to the total number of points in a recording. We consider the three density metrics in [36], each of which is inversely proportional to density. That is, they represent greater density as the magnitudes become smaller. The first metric ρ_1 is the average pairwise distance between fixation points within one fixation:

$$\rho_1 = \frac{\sum_{p=1}^{p-1} \sum_{q=p+1}^p d_{pq}}{\binom{p}{2}}. \quad \rho_1$$

The second density metric ρ_2 has the same numerator with ρ_1 : the pairwise distances of all identified fixation points. The denominator is simply the number of fixation points. Hence, as the number of included points increases, ρ_2 experiences greater amplification as compared to ρ_1 . The reason that ρ_2 is considered in [36] is due to the relationship with the objective function of its first formulation, *Minimize Average Intra-Fixation Sum of Distances* [36, formulation (12a) – (12f)]. Though our demonstration for detecting fixation outliers focuses on the latter formulation in [36], we retain ρ_2 in our comparison for the sake of completeness:

$$\rho_2 = \frac{\sum_{p=1}^{p-1} \sum_{q=p+1}^p d_{pq}}{p}. \quad \rho_2$$

The third density metric ρ_3 is the minimal square area covering the fixation divided by the number of included fixation points:

$$\rho_3 = \frac{(2f)^2}{p}. \quad \rho_3$$

The center shift λ measures the Euclidean distance between the FID⁺ fixation centroid to the I-VT filter centroid. Additionally, we report the fixation outlier budget usage β , which is the ratio of the total number of identified fixation outliers to the cumulative outlier budget over all data chunks in the ten data recordings. The reported runtime is the average of the cumulative runtime of all data chunks in each of the ten data recordings.

Each entry in the evaluation metrics columns in Tables 2 and 3 is averaged over ten recordings and all data chunks per recording. Each entry in the runtime columns reports the averaged cumulative runtime for solving approximately 721 data chunks of the α -densest fixations. Even for the most time-consuming α level, $\alpha = 0$, the average runtime per chunk to find the densest fixation with outliers was still well under 20

seconds (17.8 seconds). For larger values of α , the average runtime exhibited even better performance: for $\alpha = 0.8$, the average runtime of each data chunk is less than 0.13 second. In Table 2, the optimization models for all but twelve chunks (eleven for $\alpha = 0$, and one for $\alpha = 0.1$) solved to global optimality within the one-hour time limit for formulation (6a)–(6j). The addition of the lower bound ℓ_1 and ℓ_2 enabled two additional models at $\alpha = 0$, and the sole model with $\alpha = 0.1$, to be solved to global optimality.

The general trend of evaluation metrics and runtime from $\alpha = 0$ to $\alpha = 1$ are similar in Tables 2 and 3. It indicates that α has a similar effect on fixation identification and fixation properties in both formulations.

When compared to Table 3, the entries in the initial columns of Table 2 demonstrate the effect of removing outliers. In particular, values of the average fixation duration δ^{avg} rate are smaller in Table 2, indicating that less gaze points are identified as fixation points by the FID⁺ filter. The difference of δ^{avg} is actually rather small, roughly akin to a single gaze point, between Tables 2 and 3. Similar to δ^{avg} , the average cover rate γ^{avg} value under every α level is slightly smaller in Table 2. Both Tables 2 and 3 have the same increasing trends on δ^{avg} and γ^{avg} when α increases.

The three density metrics appear with smaller values in Table 2, as compared to Table 3. Recalling that density is larger for smaller values of ρ_1 , ρ_2 and ρ_3 , it demonstrates that when allowing outliers within fixations, the mathematical formulation can further refine the gaze points within chunks to identify denser fixations. It is worth noting that ρ_3^{avg} is two to three times smaller in Table 2 than in Table 3. ρ_3^{avg} is the ratio of the minimal area bounding box of the identified fixation, to the number of points this fixation contains, is identical to the objective in formulation (6a)–(6j). ρ_3^{avg} becomes smaller either when the fixation bounding area is smaller, or when the fixation duration decreases.

This trend of ρ_3^{avg} is strong evidence for the impact of outlier points on fixation density. Using the outlier budget $p^k = \lceil 0.01 \cdot |C^k| \rceil$ as specified in the experimental setup, 74.2% of the fixations by formulation (6a)–(6j) identify only a single outlier point per fixation (chunk size less than or equal to 100 points). This is further underscored in Table 2, as the change in fixation duration is relatively minimal. However, ρ_3 reduced by nearly two thirds. This indicates that a small group of outlier points are substantially skewing the size of the minimum apothem r and so the minimum fixation bounding box, and should be eliminated in the fixation.

For all values of α , the center shift λ^{avg} reported in Table 2 is larger than λ^{avg} in Table 3; λ^{avg} measures the Euclidean distance (in pixels) between the FID⁺ fixation centroid (as specified by (x^f, y^f)), and the I-VT filter centroid. This increase in λ^{avg} reflects stray data points being eliminated via the outlier budget in the FID⁺

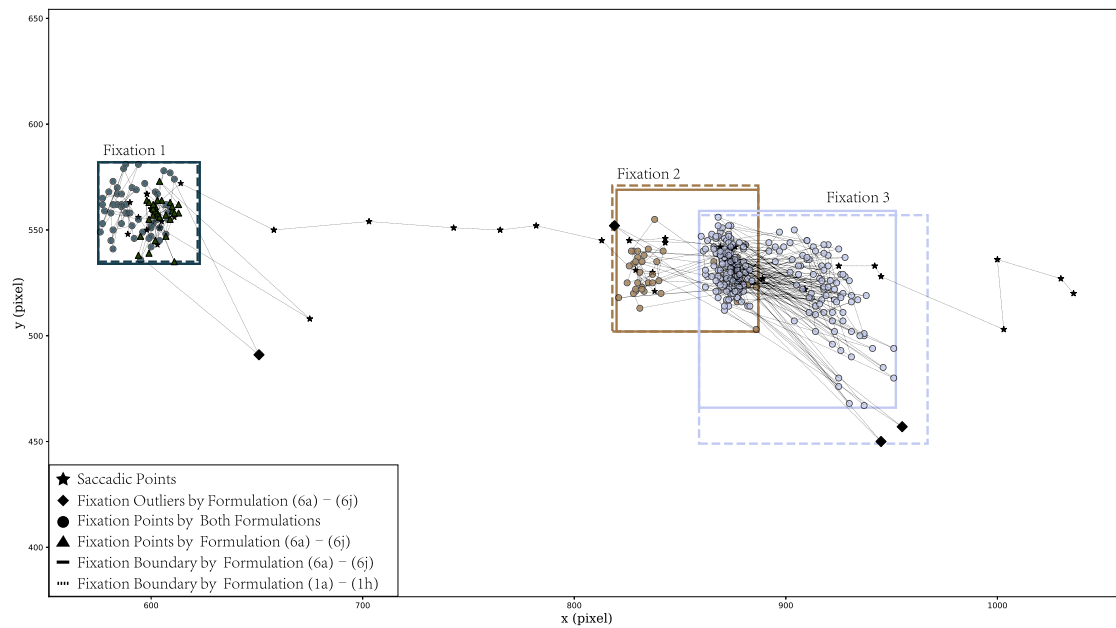


Fig. 8. Fixation identification result with the FID⁺ filter versus the FID filter, $\alpha = 0.5$, on the gaze sequence in Fig. 4(b).

filter, so as to better concentrate around the actual fixation. The outlier budget ratio β in Table 2 decreases as α increases, due to identified fixation outlier points being penalized in objective function (6a). Therefore, the penalty parameter α not only serves for balancing the trade-off between density and number of fixation points in for formulation (6a)–(6j), it also has significant influence on the number of fixation outliers identified by the formulation. One notable finding is that the budget $\mathcal{P}$ is not always used, even for small α levels.

The improved fixation metrics come with the trade-off of increased computational run time. The Gurobi runtime in Table 2 increases substantially compared with Table 3. The increase appears between $\alpha = 0$ and $\alpha = 0.1$, where much more effort is consumed in balancing the objective function trade-off of including a point, or incurring the penalty of α [36]. As shown in Fig. 7(b), the average runtime at each level of data chunk size increases significantly at $\alpha = 0$ and $\alpha = 0.1$. At the same time, we find that nearly 95% of the outlier-aware optimization models still solved to global optimality in under one second at $\alpha = 0$ and $\alpha = 0.1$, which we believe to be quite competitive.

The last four columns in Table 2 report the average Gurobi runtime and overall runtime when using lower bounds derived from Algorithms 1 and 2. Under all α levels, the reported Gurobi runtime from formulation (6a)–(6j) with Algorithms 1 and 2 is less than the Gurobi time from solely solving the formulation (6a)–(6j), which demonstrates that the bounds produced by both of the algorithms are effective in reducing the computational difficulty to the solver. However, because Algorithm 2 requires additional computational cost for processing the dataset, the average overall runtime for formulation (6a)–(6j) with Algorithm 2 only outperforms the experiment using solely formulation (6a)–(6j) for the $\alpha = 0$ and $\alpha = 0.1$ levels. Moreover, the additional time cost for running Algorithm 2 averages around 246 seconds. On the other hand, the time cost for running Algorithm 1 per chunk is negligible, and thus does not contribute to much additional time in Table 2. The average overall runtime of formulation (6a)–(6j) with Algorithm 1 is still smaller than the runtime for running the formulation (6a)–(6j) solely. The runtime comparison indicates that both of the algorithms contribute to reducing the runtime of solving optimization

models. That said, because Algorithm 2 incurs additional computational cost for data processing, only formulation (6a)–(6j) with Algorithm 1 outperforms in both Gurobi optimization time and overall runtime at every α level than only using formulation (6a)–(6j). Future work may focus on improving the computational efficiency of the implementation of Algorithm 2.

5. Conclusions

This paper introduces outlier aware fixation identification for gaze data by extending the recent FID (fixation-inner-density) filter that identifies the densest fixations in gaze data. Our new FID⁺ filter enables stray gaze points within fixations to be flagged and eliminated from fixation consideration, thereby increasing the accuracy and precision of key metrics related to the actual fixation. Gaze data collected by eye-tracking devices is collected as a sequence of points representing the locations where eyes focus. Spatially and temporally adjacent points are clustered as fixations. Fixation features – such as location, duration and inner-density – carry information about user attention and awareness in behavioral research. Such features are inherently influenced by how fixations and saccades (gaze points between fixations) are labeled by the fixation identification algorithms. Downstream behavioral properties, such as dwell time, fixation heatmap and pupil dilation during fixations, are impacted by the accuracy and precision of the fixation identification approach that is used.

Two popular fixation identification methods in practice are the I-VT and I-DT filters. They use relatively simple properties of gaze data and can be implemented efficiently in commercial eye-tracking devices. However, they can lead to inaccurate fixation results, which will result in misrepresenting behavioral patterns. The recently developed FID filter [36] overcomes the limitations of these baseline methods via integer optimization to optimize for fixation inner-density, with an iterative algorithm that exploits the ability to decompose an entire gaze stream into components, or chunks. In this paper we augment the FID optimization formulation with a new set of variables that indicate whether gaze point t is an outlier for fixation f . Moreover, we carefully design enhanced constraints that enable the strict fixation time consistency condi-

tion to be relaxed, by allowing for a small budget of fixation outlier points to be admitted. The enhanced integer optimization formulation (6a)–(6j) can recognize stray gaze points as fixation outliers, a concept that is underexplored in fixation identification algorithms. Raw gaze data contains inevitable noise (as depicted in Fig. 4(b)), and we demonstrate that the FID⁺ filter outlined in this paper can robustly identify within-fixation outlier points, which is a significant enhancement to the existing FID filter [36].

We conduct computational experiments to compare the new FID⁺ filter with the FID filter with formulation (1a)–(1h) on the 300 Hz GRE Math reading dataset used in [36]. The result shows that the FID⁺ filter can identify fixations with substantially greater density. In particular, when comparing the density metric ρ_3 , the ratio of minimal area bounding box and fixation point number, the FID⁺ filter featured a 2–3 times reduction in ρ_3^{avg} while considering a small number of points as outliers within each fixation. Thus, these developments hold much promise for outlier-aware fixation identification.

Fig. 8 highlights the comparison of fixation identification results from the FID⁺ filter and the FID filter on the noisy raw gaze sequence showed in Fig. 4(b). The illustrated gaze stream segment contains three fixations. For Fixation 1, while the identified fixation boundary looks identical for both methods, it turns out that, due to the ability to eliminate outlier points, the enhanced formulation contains 50% more points than the original formulation. This has the unexpected effect that formulation (6a)–(6j) has a slightly larger area, because such increased area greatly increases the number of included fixation points after outlier removal. Formulation (1a)–(1h) identifies all gaze points appearing before the outlier point flagged by formulation (6a)–(6j) as non-fixation points, while balancing the inherent trade-off present in objective function (6a). The gaze points at Fixation 2 are well clustered, so the two formulations have fairly similar results. For Fixation 3, formulation (6a)–(6j) identifies two fixation outliers and the fixation area decreases significantly as compared with the area identified by formulation (1a)–(1h). The outlier-aware identification results of formulation (6a)–(6j) likely have substantial impacts on the number of identified fixation points, as well as fixation bounding regions. This behavior is similar across chunks in the gaze data stream.

The approach outlined in this paper does have some limitations. Due to the additional variables and constraints, the runtime for solving formulation (6a)–(6j) is slower than formulation (1a)–(1h) at each level of α , and substantially so for the instances with large chunk size at $\alpha = 0$ and $\alpha = 0.1$. We introduce two geometric arguments, and algorithms, for deriving lower bounds on r_f to accelerate the speed of reaching global optimality. Both algorithms find stronger lower bounds (ℓ_1 and ℓ_2) that are able to reduce Gurobi runtime, although more work is needed to improve the competitiveness for a small number of instances at $\alpha = 0$ and $\alpha = 0.1$.

Moreover, more work remains for refining Algorithm 2 to reduce its overall run time for computing lower bound ℓ_2 . Another possible direction of future work is to more carefully investigate suitable budget values for each data chunk. While we set the outlier budget value to approximately 1% of the length of the data chunk, other features such as data chunk dispersion, and the average velocity of points, could suggest improved estimates for the number of fixation outliers. Each data chunk could thereby have a data-driven budget value based on its features.

More broadly, we believe that the efforts of FID⁺ will empower future studies on fixation *micro-patterns* – that is, the distribution of gaze points within an individual fixation which represent a further refinement of eye movement data [37]. Prior work in [46] shows that these patterns can reveal significant information about focused attention and effort, which subsequent findings further support [36]. Inner-density, as a representation of fixation micro-patterns, incorporates both the temporal and spatial aspects

of the fixation. When combined, these aspects reveal significant and previously undiscovered information about attention.

CRediT authorship contribution statement

Wen Liu: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing - original draft, Visualization. **Andrew C. Trapp:** Conceptualization, Methodology, Formal analysis, Writing - original draft, Writing - review & editing, Supervision, Project administration. **Soussan Djamshbi:** Conceptualization, Resources, Writing - review & editing, Supervision.

Acknowledgments

The authors would like to thank the WPI User Experience and Decision Making (UXDM) Laboratory for providing the eye-tracking environment. We are especially grateful to Dr. Mina Shojaeizadeh for conducting the experiments and providing the data recordings used for computational experiments in this paper.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.omega.2020.102298.

References

- [1] Djamshbi S. Eye tracking and web experience. *AIS Transactions on Human-Computer Interaction* 2014;6(2):37–54.
- [2] Goldberg JH, Stimson MJ, Lewenstein M, Scott N, Wichansky AM. Eye tracking in web search tasks: Design implications. In: *Proceedings of the 2002 Symposium on Eye Tracking Research & Applications*. ACM; 2002. p. 51–8.
- [3] Djamshbi S, Siegel M, Tullis T. Generation y, web design, and eye tracking. *Int J Hum Comput Stud* 2010;68(5):307–23.
- [4] Graham DJ, Orquin JL, Visschers VH. Eye tracking and nutrition label use: a review of the literature and recommendations for label enhancement. *Food Policy* 2012;37(4):378–82.
- [5] Van Gog T, Scheiter K. Eye tracking as a tool to study and enhance multimedia learning. *Learn Instr* 2010;20:95–9.
- [6] Shi SW, Wedel M, Pieters F. Information acquisition during online decision making: a model-based exploration using eye-tracking data. *Manage Sci* 2013;59(5):1009–26.
- [7] Meißner M, Musalem A, Huber J. Eye tracking reveals processes that enable conjoint choices to become increasingly efficient with practice. *Journal of Marketing Research* 2016;53(1):1–17.
- [8] Hübner AH, Kuhn H. Retail category management: state-of-the-art review of quantitative research and software applications in assortment and shelf space management. *Omega (Westport)* 2012;40(2):199–209.
- [9] Chandon P, Hutchinson JW, Bradlow ET, Young SH. Does in-store marketing work? effects of the number and position of shelf facings on brand attention and evaluation at the point of purchase. *J Mark* 2009;73(6):1–17.
- [10] Al-Moteri MO, Symmons M, Plummer V, Cooper S. Eye tracking to investigate cue processing in medical decision-making: a scoping review. *Comput Human Behav* 2017;66:52–66.
- [11] Traxler MJ, Morris RK, Seely RE. Processing subject and object relative clauses: evidence from eye movements. *J Mem Lang* 2002;47(1):69–90.
- [12] Duchowski AT. A breadth-first survey of eye-tracking applications. *Behavior Research Methods, Instruments, & Computers* 2002;34(4):455–70.
- [13] Chen Z, Fu H, Lo W-L, Chi Z. Strabismus recognition using eye-tracking data and convolutional neural networks. *J Healthc Eng* 2018;2018.
- [14] Vidal M, Turner J, Bulling A, Gellersen H. Wearable eye tracking for mental health monitoring. *Comput Commun* 2012;35(11):1306–11.
- [15] Gillespie-Smith K, Fletcher-Watson S. Designing AAC systems for children with autism: evidence from eye tracking research. *Augmentative and Alternative Communication* 2014;30(2):160–71.
- [16] Zhang X, Kulkarni H, Morris MR. Smartphone-based gaze gesture communication for people with motor disabilities. In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*; 2017. p. 2878–89.
- [17] Logan K, Iacono T, Trembath D. A systematic review of research into aided AAC to increase social-communication functions in children with autism spectrum disorder. *Augmentative and Alternative Communication* 2017;33(1):51–64.
- [18] Pai S, Ayare S, Kapadia R. Eye controlled wheelchair. *International Journal of Scientific & Engineering Research* 2012;3(10).
- [19] Maimon-Mor RO, Fernandez-Quesada J, Zito GA, Konaris C, Dziemian S, Faisal AA. Towards free 3d end-point control for robotic-assisted human reaching using binocular eye tracking. In: *2017 International Conference on Rehabilitation Robotics (ICORR)*. IEEE; 2017. p. 1049–54.

- [20] Clarke M, Price K. Augmentative and alternative communication for children with cerebral palsy. *Paediatr Child Health (Oxford)* 2012;22(9):367–71.
- [21] Komogortsev OV, Khan JI. Eye movement prediction by oculomotor plant kalman filter with brainstem control. *Journal of Control Theory and Applications* 2009;7(1):14–22.
- [22] Holmqvist K, Nyström M, Mulvey F. Eye tracker data quality: What it is and how to measure it. In: *Proceedings of the Symposium on eye tracking research and applications*. ACM; 2012. p. 45–52.
- [23] Blignaut P, Wium D. Eye-tracking data quality as affected by ethnicity and experimental design. *Behav Res Methods* 2014;46(1):67–80.
- [24] Rousseeuw PJ, Driessen KV. A fast algorithm for the minimum covariance determinant estimator. *Technometrics* 1999;41(3):212–23.
- [25] Wright C, Hu M. A note on detecting outliers in short autocorrelated data using joint estimation and exponentially weighted moving average. *Omega (Westport)* 2003;31(4):319–26.
- [26] Gupta M, Gao J, Aggarwal CC, Han J. Outlier detection for temporal data: a survey. *IEEE Trans Knowl Data Eng* 2013;26(9):2250–67.
- [27] Campos GO, Zimek A, Sander J, Campello RJ, Micenkova B, Schubert E, et al. On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study. *Data Min Knowl Discov* 2016;30(4):891–927.
- [28] Ahmad S, Lavin A, Purdy S, Agha Z. Unsupervised real-time anomaly detection for streaming data. *Neurocomputing* 2017;262:134–47.
- [29] Feit AM, Williams S, Toledo A, Paradiso A, Kulkarni H, Kane S, et al. Toward everyday gaze input: Accuracy and precision of eye tracking and implications for design. In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM; 2017. p. 1118–30.
- [30] Domingues R, Filippone M, Michiardi P, Zouaoui J. A comparative evaluation of outlier detection algorithms: experiments and analyses. *Pattern Recognit* 2018;74:406–21.
- [31] Salvucci DD, Goldberg JH. Identifying fixations and saccades in eye-tracking protocols. In: *Proceedings of the 2000 Symposium on Eye Tracking Research & Applications*. ACM; 2000. p. 71–8.
- [32] Blignaut P. Fixation identification: the optimum threshold for a dispersion algorithm. *Attention, Perception, & Psychophysics* 2009;71(4):881–95.
- [33] Veneri G, Piu P, Rosini F, Federighi P, Federico A, Ruffa A. Automatic eye fixations identification based on analysis of variance and covariance. *Pattern Recognit Lett* 2011;32(13):1588–93.
- [34] Nyström M, Holmqvist K. An adaptive algorithm for fixation, saccade, and glissade detection in eyetracking data. *Behav Res Methods* 2010;42(1):188–204.
- [35] Andersson R, Larsson L, Holmqvist K, Stridh M, Nyström M. One algorithm to rule them all? an evaluation and discussion of ten eye movement event-detection algorithms. *Behav Res Methods* 2016;1–22.
- [36] Trapp AC, Liu W, Djamasbi S. Identifying fixations in gaze data via inner density and optimization. *INFORMS J Comput* 2019;31(3):459–76.
- [37] Liu W, Djamasbi S, Trapp AC, Shojaeizadeh M. Measuring focused attention using fixation inner-density. In: *International Conference on Augmented Cognition*. Springer; 2018. p. 105–16.
- [38] Robinson DA. The mechanics of human smooth pursuit eye movement. *J Physiol (Lond)* 1965;180(3):569–91.
- [39] Nyström M, Andersson R, Holmqvist K, Van De Weijer J. The influence of calibration method and eye physiology on eyetracking data quality. *Behav Res Methods* 2013;45(1):272–88.
- [40] Tobii. Tobii technology. <http://www.tobii.com>; 2020.
- [41] Holmqvist K, Nyström M, Andersson R, Dewhurst R, Jarodzka H, Van de Weijer J. Eye tracking: a comprehensive guide to methods and measures. OUP Oxford; 2011.
- [42] Engbert R, Kliegl R. Microsaccades uncover the orientation of covert attention. *Vision Res* 2003;43(9):1035–45.
- [43] Engbert R, Mergenthaler K. Microsaccades are triggered by low retinal image slip. *Proceedings of the National Academy of Sciences* 2006;103(18):7192–7.
- [44] Van der Lans R, Wedel M, Pieters R. Defining eye-fixation sequences across individuals and tasks: the binocular-individual threshold (bit) algorithm. *Behav Res Methods* 2011;43(1):239–57.
- [45] Li B, Wang Q, Barney E, Hart L, Wall C, Chawarska K, et al. Modified DBSCAN algorithm on oculomotor fixation identification. In: *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*. ACM; 2016. p. 337–8.
- [46] Shojaeizadeh M, Djamasbi S, Trapp AC. Density of gaze points within a fixation and information processing behavior. In: *International Conference on Universal Access in Human-Computer Interaction*. Springer; 2016. p. 465–71.
- [47] Gurobi Optimization, LLC. Gurobi optimizer reference manual. <http://www.gurobi.com>; 2020.
- [48] Smid M. Finding K points with a smallest enclosing square. Report MPI-92-152, Max-Planck-Institut für Informatik; 1993.
- [49] Olsen A. The Tobii I-VT fixation filter. 2012.
- [50] Komogortsev OV, Gobert DV, Jayarathna S, Koh DH, Gowda SM. Standardization of automated analyses of oculomotor fixation and saccadic behaviors. *IEEE Trans Biomed Eng* 2010;57(11):2635–45.
- [51] MathWorks, Inc. MATLAB User Guide. 2016.