Behavior Research Methods (2020) 52:1122–1139
https://doi.org/10.3758/s13428-019-01304-3

An adaptive algorithm for fast and reliable online saccade detection

Richard Schweitzer[1,2,3] · Martin Rolfs[1,2]

Published online: 8 November 2019

Abstract

To investigate visual perception around the time of eye movements, vision scientists manipulate stimuli contingent upon the onset of a saccade. For these experimental paradigms, timing is especially crucial, because saccade offset imposes a deadline on the display change. Although efficient online saccade detection can greatly improve timing, most algorithms rely on spatial-boundary techniques or absolute-velocity thresholds, which both suffer from weaknesses: late detections and false alarms, respectively. We propose an adaptive, velocity-based algorithm for online saccade detection that surpasses both standard techniques in speed and accuracy and allows the user to freely define the detection criteria. Inspired by the Engbert–Kliegl algorithm for microsaccade detection, our algorithm computes two-dimensional velocity thresholds from variance in the preceding fixation samples, while compensating for noisy or missing data samples. An optional direction criterion limits detection to the instructed saccade direction, further increasing robustness. We validated the algorithm by simulating its performance on a large saccade dataset and found that high detection accuracy (false-alarm rates of < 1%) could be achieved with detection latencies of only 3 ms. High accuracy was maintained even under simulated high-noise conditions. To demonstrate that purely intrasaccadic presentations are technically feasible, we devised an experimental test in which a Gabor patch drifted at saccadic peak velocities. Whereas this stimulus was invisible when presented during fixation, observers reliably detected it during saccades. Photodiode measurements verified that—including all system delays—the stimuli were physically displayed on average 20 ms after saccade onset. Thus, the proposed algorithm provides a valuable tool for gaze-contingent paradigms.

Keywords Saccade detection · Eye movements · Intrasaccadic perception · Gaze-contingent presentation

In the field of active vision, most eyetracking experiments study visual perception around goal-directed rapid eye movements, so-called saccades. Specifically, when investigating trans-saccadic or intrasaccadic perception, an experimental paradigm has to be implemented in a way that a stimulus or the configuration of stimuli is manipulated online (i.e., in real time) and gaze-contingently, starting with the onset of a saccade (Higgins & Rayner, 2015; Hollingworth, Richard, & Luck, 2008; Melcher & Colby, 2008; Prime, Vesia, & Crawford, 2011; Wolf & Schütz, 2015). Because saccades are rapid and brief events, often with a skewed velocity profile (Figs. 1a and 1b), this is not always as trivial as it may sound. Every computational step between an eye movement and a change in the display adds undesired delays, and every shortcut (e.g., through rough approximations) may lead to false alarms—that is, the detection of a saccade when none has happened. Here we will discuss an algorithm that realizes early online saccade detection without sacrificing reliability, and is thus able to greatly reduce the overall delay between saccade onset and display change.

To elucidate the challenge that gaze-contingent paradigms pose with regard to timing, let us consider a typical trans-saccadic experimental scenario: Participants are instructed to make a saccade toward a colored patch at a 10-deg of visual angle (dva) eccentricity, resulting in saccades with average peak velocities of 300 dva per second (dva/s) and durations of 40 ms (Collewijn, Erkelens, & Steinman, 1988). When trying to manipulate the color of the patch during the saccade, so that upon landing an updated stimulus with a new color is displayed to the observer, we as experimenters have to consider at least four additive sources of latency in our experimental setups (Fig. 2) in order to make the presentation deadline of each saccade offset.

✉ Richard Schweitzer
richard.schweitzer@hu-berlin.de

[1] Department of Psychology, Humboldt-Universität zu Berlin, Berlin, Germany

[2] Bernstein Center for Computational Neuroscience Berlin, Berlin, Germany

[3] Berlin School of Mind and Brain, Humboldt-Universität zu Berlin, Berlin, Germany

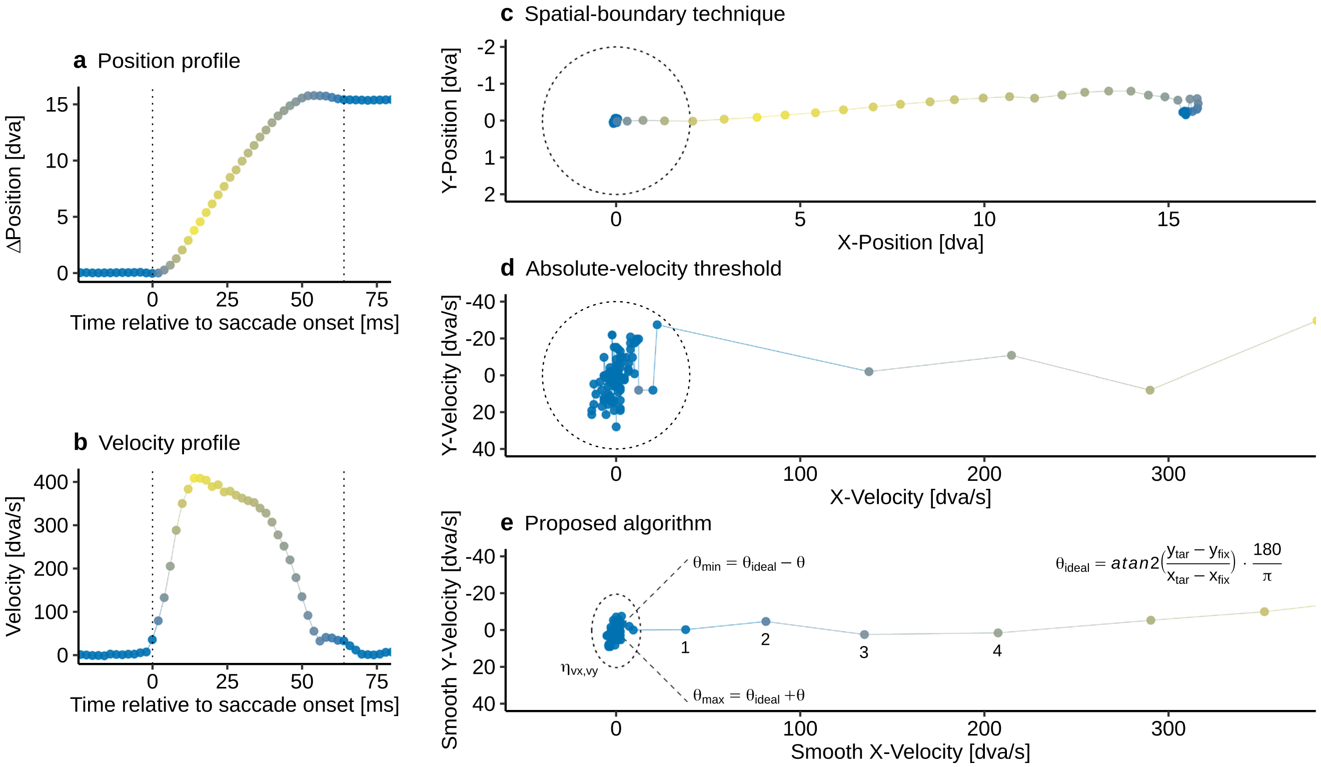

Fig. 1 Illustration of different saccade detection techniques, based on an exemplary saccade. (a–b) Plots of position and velocity profiles of a horizontal, rightward 15-dva saccade, sampled uniformly at 500 Hz. Color represents the sample-to-sample velocity (yellow = peak velocity). (c) Illustration of saccade detection using a spatial-boundary technique. Saccades are detected once gaze position reaches past the spatial boundary, defined by a 2-dva radius (dotted circle) around the instructed fixation position. (d) Illustration of an absolute-velocity threshold. Gaze position data are transformed into two-dimensional velocity space, and a saccade is detected once velocities exceed a predefined value—for example, 40 dva/s in this example. (e) Illustration of the proposed algorithm. Gaze position data are resampled to a uniform sampling rate, transformed into two-dimensional velocity space, which is smoothed by a five-point running-average filter. Median-based standard deviations are computed separately for the horizontal and vertical dimensions, forming an elliptic velocity threshold $\eta_{vx,vy}$. An optional direction criterion θ (here, 45°) can restrict detection to a range around the instructed saccade direction θ_{ideal} (e.g., computed via the fixation and saccade landing positions xy_{fix} and xy_{tar}), with θ_{max} and θ_{min} as the upper and lower boundaries. Moreover, the user may specify, in order to detect a saccade, how many samples are needed that satisfy both the velocity and direction criteria. In this example, we have numbered the first four samples for which this is the case.

First, the online access to gaze position data is delayed. This *end-to-end sample delay* includes not only the time taken for a physical event to be registered, processed, and made available online by the eyetracking system (e.g., capturing

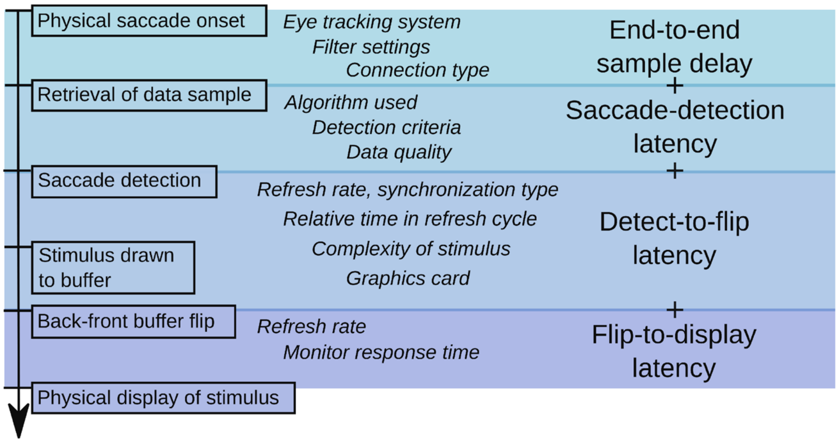

Fig. 2 Schematic illustrating four categories of latencies in a temporal sequence (top to bottom) occurring when display changes are locked to saccade onset. Factors influencing the magnitude of the delays are shown in italics.

an image of the eye, fitting the pupil and corneal reflection, and extrapolating gaze position), but also the time needed to retrieve the data via Ethernet, USB, or analog ports. Although the retrieval time is usually negligible (i.e., on the order of microseconds), the total end-to-end sample delay can be considerable. According to manufacturer manuals, it may range from 1.8 to 3 ms in the EyeLink 1000 (SR Research, 2010), from 1.7 to 1.95 ms in the Trackpixx3 (VPixx Technologies, 2017), from 3 to 14 ms in the EyeLink II (SR Research, 2005), and up to 33 ms in the Tobii TX Series (Tobii Technology AB, 2010).

Second, as we need a reliable, and thus often a more conservative, criterion to decide whether a saccade has actually been initiated, the onset of the saccade detected online usually lags behind the onset of the saccade detected offline. Henceforth, this delay will be referred to as the *saccade detection latency.* Techniques to detect saccades during experiments often involve an invisible spatial boundary (Fig. 1c) at some distance from the initial fixation point that the gaze position has to cross (Rayner, 1975). This widely used technique (e.g., Collins, Rolfs, Deubel, & Cavanagh, 2009; Kalogeropoulou & Rolfs, 2017; Szinte & Cavanagh, 2011) usually provides reliable but late saccade detection (~ 15 ms after the actual saccade onset at a sampling rate of 500 Hz for a boundary 2 dva from fixation; see Fig. 5a in the Results). An alternative to the boundary technique is based on velocity thresholds (Fig. 1d): The measured speed of the eye must exceed a certain value, such as 30 dva/s (Deubel, Schneider, & Bridgeman, 1996; Han, Saunders, Woods, & Luo, 2013; Panouillères et al., 2016), 40 dva/s (Castet, Jeanjean, & Masson, 2002), or even 100 dva/s (Arabadzhiyska, Tursun, Myszkowski, Seidel, & Didyk, 2017), so saccades can be detected much earlier, but they often suffer from increased false alarm rates.

Third, once we have detected a saccade in the data retrieved online, the stimulus has to be drawn to the graphics card's back-buffer and the flip with the front-buffer has to be synchronized with the display's vertical retrace (Kleiner, Brainard, & Pelli, 2007). This *detect-to-flip latency* is determined by the refresh rate of the monitor and depends on the time of detection within the refresh cycle. Novel technologies such as G-Sync are able to reduce this latency to the submillisecond range by allowing flips as soon as rendering is complete, without having to wait for the screen refresh (Poth et al., 2018).

Fourth, there is the *flip-to-display latency*—that is, the time from the execution of the flip until the physical stimulus presentation on the screen. Whereas the transfer of the entire video signal takes up to one frame duration, the display's reaction time can additionally increase the flip-to-display latency, as well as introduce temporal jitter.

Taking into account all sources of delay (e.g., a 5-ms end-to-end sample delay using an EyeLink II at 500 Hz with normal link filtering + 15-ms detection latency using a boundary technique + 5-ms mean detect-to-flip latency with a 120-Hz monitor + 8.3-ms flip-to-display latency), the physical change will occur in the last quarter of the 40-ms saccade. Because both gaze-contingent displays and saccade profiles can be subject to considerable variance, we thus increase the risk of achieving a postsaccadic instead of the intended intrasaccadic display change. Failure to acknowledge or control these latencies may thus lead to erroneous results and unwarranted conclusions.

Although most of the latencies mentioned above largely depend on the specific hardware used, we can optimize the saccade detection latency to achieve low-latency gaze-contingent presentations. Crucially, the choice of the saccade detection criterion determines both the timing and the reliability of the experimental paradigm: Whereas a conservative detection criterion (e.g., a spatial boundary) may provide reliable but late detection, a liberal detection criterion (e.g., a low absolute-velocity threshold) may lead to early detection at the cost of increased false alarm rates. This may become especially relevant when detecting saccades based on velocity using high sampling frequencies, as any error in gaze position divided by a shorter sampling interval will lead to amplified velocity estimates (Han, Saunders, Woods, & Luo, 2013). To achieve reliable online detection, velocity thresholds would therefore have to be manually adjusted to the precision and sampling frequency of the eyetracker, as well as to the situation- and participant-dependent noise levels (see also Engbert & Mergenthaler, 2006). To date, no algorithm provides both fast and early online saccade detection while at the same time remaining reliable in noisier conditions.

Here we present a velocity-based online saccade detection algorithm that adaptively estimates noise levels on the basis of preceding fixation data to provide robust results in the presence of random sample-to-sample noise, dropped samples, blinks, and fixational eye movements, while allowing its user to flexibly adjust the detection criterion to the specific experimental situation. We tested the performance of the algorithm and the impact of various parameter combinations and noise levels in a large-scale simulation with more than 34,000 saccades, and compared the algorithm to boundary techniques and absolute-velocity thresholds. We then present an objective and perceptual test for reliable, gaze-contingent, and strictly intrasaccadic presentations that underlines the algorithm's usefulness in real-time experimental scenarios.

The proposed algorithm

Online saccade detection relies on the continuous sampling of gaze position data (x, y) and the corresponding timestamps (t) throughout each trial of the experiment. Gaze position data collected during fixation is used to establish a threshold to demarcate the transition from fixation to saccade. Following Engbert and Kliegl's widely used algorithm for microsaccade detection (Engbert & Kliegl, 2003; Engbert & Mergenthaler, 2006), the algorithm thus detects the onset of a saccade based on an elliptical, two-dimensional velocity threshold $\eta_{vx,vy}$ (dotted line, Fig. 1d), as defined by the product of the median-based standard deviation of horizontal and vertical gaze position dimensions (σ_{vx}, σ_{vy}) and a free scaling parameter λ to adjust the velocity criterion.

$$\eta_{vx,vy} = \lambda \cdot \sigma_{vx,vy}$$

In addition, the user may provide a parameter k specifying how many of the most recent of all velocity samples must exceed the defined threshold. That way, robustness against false alarms due to noise-related velocity peaks is increased. In case the user intends to limit detection of saccades to an instructed saccade direction (θ_{ideal}), which is often the case in controlled experimental paradigms, the algorithm allows for specification of an additional direction criterion θ that determines the direction range around the ideal saccade direction that individual velocity samples are allowed in (dashed lines, Fig. 1d). This direction criterion can be used to avoid false detections of the instructed saccade as a consequence of other eye movements events that may satisfy the velocity criterion, such as blinks or microsaccades.

To make the algorithm suitable for online applications, two important features were implemented. First, owing to the fact that during online experiments it is rarely possible to retrieve every single data sample, missing position samples are linearly interpolated, either to a sampling rate specified by the user or to a sampling rate computed on the basis of the number of samples retrieved in a given time. Second, two-point velocity samples are computed (to avoid edge velocities of zero) and then smoothed by a five-point running average to reduce the impact of high frequency noise (Engbert & Mergenthaler, 2006). To not overestimate the first and most recent velocity samples, the vector edges are padded with repetitions of the first and the most recent samples, respectively. Subsequently, based on smoothed velocity samples, the median velocities (in most cases equaling zero) and the median-based standard deviations (σ_{vx}, σ_{vy}) are computed as described by Engbert and colleagues (Engbert & Kliegl, 2003; Engbert & Mergenthaler, 2006; Engbert, Rothkegel, Backhaus, & Trukenbrod, 2016):

$$\sigma_{vx,vy} = \sqrt{\left\langle \left(v_{x,y} - \left\langle v_{x,y} \right\rangle \right)^2 \right\rangle}$$

The brackets ⟨.⟩ stand for the median estimator. To optimize processing speed, we use the quick select algorithm for median selection (Press, Teukolsky, Vetterling, & Flannery, 2007).

To determine whether a saccade is ongoing, only the most recently retrieved k samples (k has to be defined by the user beforehand) are tested whether eye velocity exceeds the specified threshold, which is computed on the basis of all preceding n–k samples. An ongoing saccade is detected only if all k samples pass this velocity test criterion *vel*:

$$vel(i) = \left(\left(\frac{v_{x,n-k+i}}{\eta_{vx}} \right)^2 + \left(\frac{v_{y,n-k+i}}{\eta_{vy}} \right)^2 \right) > 1$$

As we mentioned above, in case of the application of an additional direction criterion *dir*, the direction of the same samples must also fall within a direction range specified by the user.

$$dir(i) = \theta_{max} > \left(atan2\left(v_{y,n-k+i}, v_{x,n-k+i} \right) \cdot \frac{180}{\pi} \right) > \theta_{min}$$

On the basis of this equation, the ideal saccade direction can be conveniently computed using the instructed fixation and saccade target regions (θ_{ideal}, Fig. 1e).

The algorithm automatically returns the used velocity thresholds, as well as (optionally) interpolated position data, and—if a saccade has been detected—the timestamp and computed eye velocity at detection. Since online saccade detection by definition occurs after saccade onset, and lower detection threshold are more susceptible to noise, the algorithm also provides an estimate for the actual saccade onset by tracing back in time one sample that falls below another velocity threshold—that is, the product of a user-defined threshold parameter λ_{onset} (not necessarily the same as the λ used for saccade detection) and the computed median-based standard deviation $\sigma_{vx,vy}$. This two-step procedure (Arabadzhiyska et al., 2017; Dorr, Martinetz, Gegenfurtner, & Barth, 2010) allows the user to get real-time access to a reliable timestamp of saccade onset, for instance to provide feedback on saccade latency in a certain trial, to trigger a display change at a predefined time relative to saccade onset, or to fit ongoing saccade trajectories (Han et al., 2013).

To code is openly available online at https://github.com/richardschweitzer/OnlineSaccadeDetection. It uses standard C libraries and can thus be used across platforms. We provide a module in Python, as well as an implementation to be compiled as a mex-function in Matlab (The Mathworks, Natick, MA, USA).

Materials and method

Simulation

Data

For validation of the algorithm, we compiled a dataset consisting of a total of 34,183 saccades, measured from participants' dominant eye. The data was collected from two past experiments (i.e., Schweitzer & Rolfs, 2017; Watson, Schweitzer, Castet, Ohl, & Rolfs, 2017), as well as from one pilot study. Using an EyeLink II at a sampling rate of 500 Hz, a number of 17,090 horizontal (left- and rightward) saccades with an instructed amplitude of 14.6 dva, as well as a number of 10,809 saccades in eight different directions (cardinal and intercardinal directions) and of 10 dva amplitude, entered analysis. Furthermore, collected with an EyeLink 1000+ at a sampling rate of 1000 Hz, we included 6,284 additional saccades in the same eight directions, but of 8-dva amplitude.

Pre-processing of the data used for the validation (offline data analysis) involved three steps. First, trial data was reduced to those samples between the onset of successful fixation (preceding the saccade go signal) and 100 ms after the participant's gaze first reached the target area (boundary with 2-dva radius around saccade target). Second, the onset and offset of the saccade—defined as the ground truth in all analyses—was detected using the Engbert–Kliegl algorithm (Engbert & Kliegl, 2003; Engbert & Mergenthaler, 2006) with a threshold parameter of λ = 5 and a minimum duration of 16 ms (eight samples at 500 Hz and 16 samples at 1000 Hz). Trials, in which saccades could not be detected or in which more than one saccade occurred within the chosen time interval were excluded. Third, eye movement data was transformed from the setup-specific pixel values to degrees of visual angle. Subsequently, position data and timestamps were normalized relative to the detected onset of the saccade to allow for comparisons between saccades of different amplitudes and durations. Saccade data and code used for simulations are available on the Open Science Framework: https://osf.io/3pck5/.

Procedure

To simulate the performance of the online detection algorithm, we divided the data of each trial in saccade absent (i.e., prior to saccade onset as detected offline) and saccade present (i.e., after saccade onset as detected offline) segments. The algorithm was then run sequentially on each data sample (ordered by time stamps) in the respective segment, taking into account all previous samples for threshold estimation. That way, we simulated its usage during an experimental trial in which new data samples are retrieved cumulatively. If saccades were detected while iterating through absent segments, we registered a false alarm (FA), if not, the trial counted as a correct rejection (CR). Similarly, if saccades were detected after offline-detected saccade onset, we registered a correct detection (hit), if not, the trial counted as a miss. To evaluate the performance of the boundary technique (2 dva) and absolute-velocity threshold (40 dva/s), we used the same procedure.

To explore the behavior of the algorithm in a larger parameter space, online saccade detection was tested in both absent and present segments for every available parameter combination—that is, threshold parameter λ (levels: 5, 10, 15, 20), samples above threshold needed k (levels: 1, 2, 3, 4), and direction criterion θ (levels: none, 45°, 30°, 15°). In addition, we convoluted all data samples with Gaussian noise (*SD*s: 0, 0.025, 0.05, 0.1 dva) on both horizontal and vertical dimension and randomly removed a proportion of all samples (levels: 0, 10, 20, 30%), to simulate eyetracker noise and sample loss, respectively. This test setup resulted in a total of 1,024 within-saccade conditions. To achieve a fair comparison between the proposed algorithm and the two traditional techniques, we also tested the performance of the boundary and absolute-velocity techniques for varying numbers of samples (levels: 1, 2, 3, 4).

For each within-saccade condition and additionally for each available sampling rate and saccade direction, we computed detection sensitivity index d' and summary statistics for detection latency (saccade present segments only) of the three detection methods—that is, boundary techniques, absolute-velocity thresholds, and the described online detection algorithm. In addition, we computed an efficiency score (ES)—that is, the proportion of correct rejections divided by the mean detection latency relative to the actual saccade onset (Townsend & Ashby, 1983).

Analysis

As a first step, we computed summary statistics (mean, standard deviation, standard error) for detection latency (separately for each online detection technique and within-saccade condition), and median-based standard deviation of velocity samples. For detection accuracy, we computed d', proportion of hits and false alarms for each online detection technique and condition, and estimated their standard error using nonparametric bootstrapping with 2,000 repetitions.

To understand the individual effects of the algorithm's parameters on the dependent variables d' and detection latency (in milliseconds), we applied multiple regression to the aggregated data. Sampling rate was included as an effect-coded factor (– 0.5 = 500 Hz; + 0.5 = 1000 Hz), while the threshold factor λ and samples above threshold k were included as continuous predictors centered around their mean. Direction restriction was also included as a continuous predictor (in degrees: 180, 45, 30, 15).

To analyze the online detection algorithms' robustness against noise, we ran a second multiple regression on detection accuracy (d') and detection latency (in milliseconds), including five factors and their interactions: sampling rate (effect-coded: – 0.5 = 500 Hz; + 0.5 = 1000 Hz), Gaussian noise standard deviation (continuous; 0, 1.5, 3, 6 arcmin), percentage of samples dropped (continuous; 0%, 10%, 20%, 30%), detection technique used (dummy-coded: boundary, absolute velocity, algorithm [λ = 5], algorithm [λ = 10], algorithm [λ = 15], algorithm [λ = 20]), and the respective number of samples needed above criteria (centered, effect-coded: – 1.5 = one sample; – 0.5 = two samples; 0.5 = three samples; 1.5 = four samples).

Experimental test

Participants

Ten observers (including the first author) participated in the experiment. All observers (four female; age range: 22–35 years old) had normal or corrected-to-normal vision. The study was conducted in agreement with the Declaration of Helsinki (2008), approved by the Ethics Committee of the German Society for Psychology, and all observers provided written informed consent before participation. We tracked participants' dominant eye (eight of ten observers with right ocular dominance) for one session with an average duration of 30 min for 480 trials in total.

Apparatus

The experiment took place in a dimly lit, sound-attenuated cabin. A Propixx DLP Projector (Vpixx Technologies, Saint-Bruno, QC, Canada) running at a temporal resolution of 1,440 frames per second and a spatial resolution of 960 × 540 pixels projected into the cabin onto a 200 × 113 cm screen (Celexon HomeCinema, Tharston, Norwich, UK). The projector was connected to the experimental host-PC via a Datapixx3 (Vpixx Technologies, Saint-Bruno, QC, Canada). Observers were seated at a distance of 180 cm away from the projection screen with their head supported by a chin rest. Stimulus display was controlled using the PsychProPixx function from PsychToolbox (Kleiner et al., 2007; Pelli, 1997), running in Matlab 2016b (Mathworks, Natick, MA, USA) on a custom-built desktop computer with an Intel i7-2700K eight-core processor, 8 GB working memory, and an Nvidia GTX 1070 Ti graphics card, running Ubuntu 18.04.1 (64-bit) as the operating system. The setup is illustrated in Fig. 3. Eyetracking was performed using an EyeLink 1000+ desktop base system, tracking participants' dominant eye at a sampling rate of 2000 Hz. Tracking was controlled during the experiment using the EyeLink Toolbox (Cornelissen, Peters, & Palmer, 2002). Moreover, we collected data from a photodiode connected to an actiChamp electroencephalographic (EEG) amplifier (Brain Products, Gilching, Germany), which was attached to the lower right corner of the projection screen (i.e., at approximately 36-dva eccentricity relative to central fixation), again at a sampling rate of 2000 Hz. To synchronize the eyetracking and photo sensor data, we applied a DB-25 Y-splitter cable to simultaneously send triggers of 1-ms duration to the EyeLink host computer and EEG host computer. During pre-processing of the data, we used the EYE-EEG Toolbox (Dimigen, Sommer, Hohlfeld, Jacobs, & Kliegl, 2011) in EEGLAB (Delorme & Makeig, 2004) to temporally align both recordings. For all triggers across recordings, we found a mean absolute misalignment error of 0.38 ms—that is, below one sample.

Stimuli

The stimuli were Gabor patches of vertical orientation enveloped in a Gaussian window with a standard deviation of 0.5 dva, presented on a uniform gray background (luminance of 30 cd/m^2). All Gabor patches had a spatial frequency of 0.5 cycles per degree of visual angle (cpd) and a contrast of 100% (0% in stimulus-absent conditions).

In both saccade and fixation trials (see below and Fig. 4), the stimulus presentation duration amounted to 20 frames at a frame rate of 1440 Hz—that is, 13.9 ms. To reduce the transient elicited by a sudden stimulus onset, the first four and last four presentation frames, respectively, were used to linearly ramp up and down stimulus contrast. Presentation locations were randomly chosen in each trial: Relative to the screen center, stimuli could appear at an eccentricity of up to 8 dva within a range of 360 deg. Because we aimed to present stimuli at largely the same retinal eccentricities both during saccade and fixation trials, we estimated that intrasaccadic presentations would be realized in the first half of the saccade and would therefore be effective when the saccade crossed the screen's center vertical midline. In fact, across all participants the gaze position at stimulus onset was 1.24 dva (*SD* = 0.92 dva) left of the vertical midline for rightward saccades, and 1.31 dva (*SD* = 1.0 dva) right of it for leftward saccades.

Throughout their presentation, the Gabor patches were drifting at a constant speed equivalent to the saccadic peak velocity, which was automatically computed during the experiment. That is, after each saccade trial, gaze position data collected during the trial were resampled to 500 Hz and cropped to the relevant time interval between cue onset and 30 ms after reaching the target region. Second, we used the Engbert–Kliegl saccade detection algorithm (Engbert & Kliegl, 2003; Engbert & Mergenthaler, 2006), with a minimum duration of eight samples and λ = 10, to extract the saccade latency and saccadic peak velocity. Third, we computed the median saccadic peak velocity based on the 30 most recent saccade trials. Fourth, to investigate the effect of stimulus drift velocity relative to saccade velocity, we defined the stimulus drift velocity as the resulting median or added or subtracted 50 dva/s, on the basis of which we then computed the Gabor's phase change per frame. This procedure resulted in three conditions and distributions of stimulus drift velocities (M_{-50} = 366 dva/s, M_0 = 416 dva/s, M_{+50} = 466 dva/s), matching the mean saccadic peak velocity of 419 dva/s. Although in both conditions drift velocity was computed on the basis of the most recent saccadic peak velocities, given the fact that fixation and saccade trials were presented in an

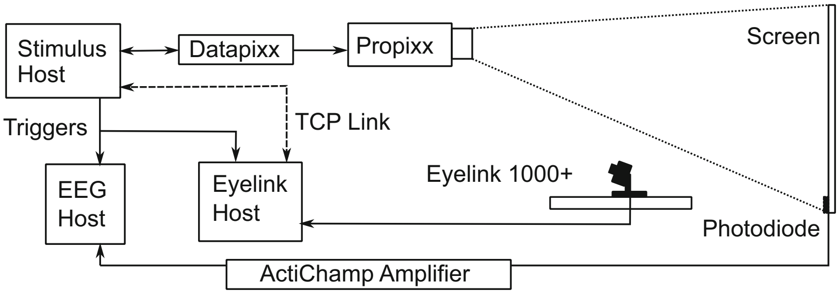

Fig. 3 Setup used to co-register the gaze position and photodiode data. The stimulus host computer performed online monitoring of gaze position via the TCP link, stimulus presentation using a ProPixx DLP projector with a frame rate of 1440 Hz, as well as synchronized triggering of the electroencephalographic (EEG) and EyeLink host computers, recording photodiode and gaze position, respectively.

interleaved manner, the presented drift velocities might have differed between conditions. This, however, was not the case [$t_{-50}(9) = -0.28$, $p_{-50} = .80$; $t_0(9) = -1.8$, $p_0 = .10$; $t_{+50}(9) = -0.43$, $p_{+50} = .68$]. To achieve visibility during saccades, the Gabor patches always drifted in the direction of the saccade (Castet & Masson, 2000; Deubel, Elsner, & Hauske, 1987).

The instructed fixation location was marked using a full-contrast black rectangular dot with a white outline and size of 0.4 dva. For the saccade target location, a similar stimulus was applied, only of twice the size—that is, 0.8 dva.

Procedure

Each participant performed a total of 480 trials, consisting of 240 saccade trials and 240 fixation trials. For each trial type, there were 120 stimulus-absent and 120 stimulus-present

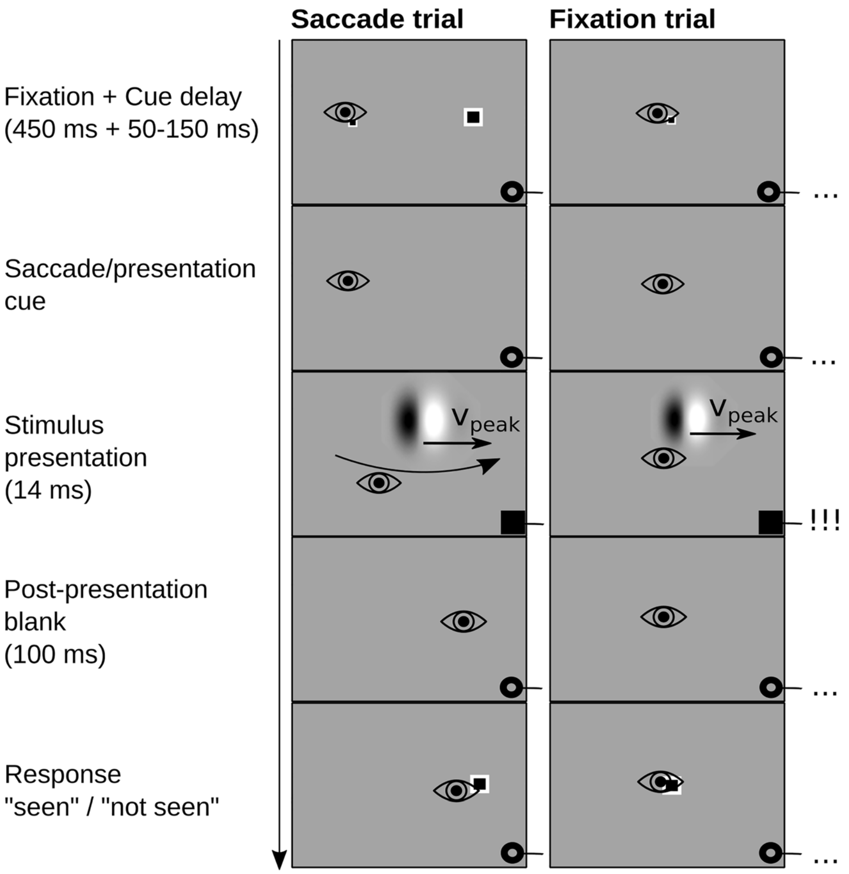

Fig. 4 Experimental procedures used in saccade and fixation trials. In saccade trials, observers made a 16-dva saccade, whereas in fixation trials, observers maintained central fixation. In saccade trials, as soon as a saccade was detected online, and in fixation trials, after the observer's median saccade latency, a Gabor patch (vertical orientation, 0.5 cpd) enveloped in a Gaussian window with a standard deviation of 0.5 dva and drifting at the saccadic peak velocity (the median peak velocity of the 30 most recent saccade trials) was presented for 13.9 ms. During stimulus presentation, a black square was projected onto a photodiode located in the lower right corner of the screen, generating a signal change in the photodiode.

trials, which then contained three stimulus velocity conditions (sum of the median peak velocity and either – 50, 0, or +50 dva/s) and two stimulus drift directions (leftward vs. rightward, in saccade trials according to the saccade direction). All trials were presented in interleaved and randomized order.

Saccade trials Each saccade trial (Fig. 4, left sequence) began with the display of two dots (see the Stimuli section), of which the smaller one represented the fixation location and the larger one the saccade target. Both dots had a horizontal eccentricity of 8 dva relative to the screen center. After successful fixation within a 2-dva radius around the fixation dot for 450 ms, followed by a random delay of 50 to 150 ms, both dots disappeared from the screen—that is, the saccade cue. Participants were instructed to make a saccade (16 dva) toward the remembered target location right after the disappearance of the dots. Saccades were detected online within a window of 10 s (mean saccade latency was 275 ms, *SD* = 135 ms) after the onset of the saccade cue using the algorithm described in this article (parameters: $\lambda = 10$, $k = 3$, $\theta = 30°$). As soon as a saccade in the instructed direction was detected, we triggered the presentation of a Gabor patch drifting at saccadic peak velocities. In stimulus-present trials, the patch occurred intrasaccadically for 13.9 ms within a radius of 8 dva around the screen center with 100% contrast, whereas in stimulus-absent trials, the patch had zero contrast. Stimulus-absent and -present trials were present in an interleaved manner and were equally probable. Regardless of whether a stimulus was present or absent in a given trial, the presentation was always accompanied by a black dot with a size of 4 dva that was displayed (for the same time as the stimulus) at the location of the photodiode attached to the lower right corner of the screen. Then, 100 ms after stimulus offset (i.e., on average, 82 ms after saccade offset), the saccade target dot would reappear, to give participants feedback on the accuracy of their saccade and prompt their response. Participants were instructed to respond with the right arrow if they had detected a stimulus, and the left arrow if they had not. They did not receive feedback on their detection performance but were shown their own saccade trajectory on the screen whenever they did not reach the saccade-target area (2-dva radius) or made more than one saccade before reaching the latter. Trials with these insufficient saccades were not repeated.

Fixation trials Fixation trials (Fig. 4, right sequence) were initiated with the display of a small dot (0.4 dva) representing the center of a fixation area with 2-dva radius. Just like during saccade trials, gaze had to stay within this area for 450 ms to initiate the presentation sequence (plus random delay of 50 to 150 ms), until the dot disappeared. Prior to stimulus presentation, a delay with the duration of the participant's median saccade latency (based on the 30 most recent saccade trials) was added to imitate the saccade trials and to increase temporal predictability. For the presentation of the rapidly drifting Gabor patch and the photodiode dot under fixation, the same parameters were applied as during saccade trials (see the Stimuli and Saccade Trials sections above). Again, 100 ms after stimulus offset, a larger dot (0.8 dva) appeared, prompting the participant's response (right arrow = "seen," left arrow = "not seen"). Participants were instructed to maintain fixation until the appearance of the response cue and received feedback whenever their gaze left the fixation area.

Analysis

We collected a total of 480 trials (240 saccade trials and 240 fixation trials in interleaved and randomized order) per participant plus one additional set of 305 trials from one participant owing to an aborted session. Due to insufficient fixation or early responses, 9.2% of all fixation trials had to be excluded. In saccade trials, 17.1% were excluded due to unsuccessful initial fixations, not reaching the target area with only one saccade or responding before having reached the target area. Although the Gabor stimuli should be invisible during fixation due to their high drift velocity (Castet & Masson, 2000; Deubel et al., 1987; García-Pérez & Peli, 2011; Watson et al., 2017), 1% of the remaining saccade trials were still excluded because the saccade offset preceded the stimulus offset (as measured by the photodiode). Finally, 0.4% of all trials were removed because of dropped frames. On average, 222 (*SD* = 13) of the initial 240 fixation trials and 201 (*SD* = 23) of the initial 240 saccade trials were included for analysis.

Photodiode data and eye movement data were merged using the EYE-EEG Toolbox (Dimigen et al., 2011) and downsampled to 1000 Hz. Saccades were detected using Engbert–Kliegl algorithm (Engbert & Kliegl, 2003; Engbert & Mergenthaler, 2006) with a threshold of $\lambda = 5$ and a minimum saccade duration of 16 samples, constituting the ground truth for the analyses on latency. In addition, messages on saccades and fixations generated by the EyeLink system were imported to validate the saccade detection results from the Engbert–Kliegl algorithm. Unfiltered photodiode voltage time series data was transformed to standard *z*-scores separately for each experimental session, so that the standard luminance of the screen produced values around 0 and the reduction in photodiode response due to the brief presentation of the black dot during stimulus presentation resulted in values well below – 4. To determine whether the stimulus was on the screen we thus selected those values below the cutoff of – 3. In both saccade and fixation trials, we computed retinal velocity of the stimulus during its presentation by estimating eye velocity (using a five-point running mean) from those gaze samples collected during stimulus presence as determined by the photodiode, and subtracting it from the drift velocity of the stimulus.

To analyze the influence of retinal velocity on task performance on a trial-by-trial basis, we used a logistic mixed-effect regression with random intercepts and slopes for observers (Bates, Mächler, Bolker, & Walker, 2015) to predict correct responses from retinal velocity (continuous predictor, *z*-scores computed separately for fixation and saccade conditions) and condition (effect-coded; – 0.5 = fixation, + 0.5 = saccade). Confidence intervals for log odds weights were determined via parametric bootstrapping with 10,000 repetitions. Hierarchical model comparisons were performed using the likelihood ratio test.

We furthermore used *t* tests to determine whether task performance (*d'*) was different from chance levels and a univariate repeated measures analysis of variance (ANOVA) to compare task performance in the fixation and saccade conditions.

Results

Simulation results

Here we simulated a situation similar to most experimental paradigms: Once an observer receives a cue to make a saccade, we continuously retrieve data samples from the eyetracker to determine whether at any point in time the observer has initiated a saccade or not. In this setup, detection performance has two main aspects, namely accuracy (i.e., detection after saccade onset and not before that) and latency (i.e., when is a saccade detected relative to the actual saccade onset, as determined offline). In this simulation, we asked two main questions: (1) How is the performance of the proposed algorithm impacted by the choice of parameters, and (2) how does its performance compare to classic techniques, such as spatial boundaries and absolute-velocity thresholds, especially under conditions of additional noise and data loss?

As is shown in Fig. 5a, online saccade detection is a trade-off between speed and accuracy. At a sampling rate of 500 Hz, given that only one retrieved data sample is sufficient for detection, boundary techniques (red squares) have a very high accuracy [*p*(FA) = 0.4%, *SD* = 0.3%; mean *d'* = 6.3, *SD* = 0.74] but long saccade detection latencies (*M* = 15 ms, *SD* = 1.1 ms), whereas absolute-velocity thresholds (green squares) have shorter detection latencies (*M* = 4.4 ms, *SD* = 0.23 ms), but with lower accuracy [*p*(FA) = 11.6%, *SD* = 3%; mean *d'* = 4.6, *SD* = 0.34]. Importantly, the type of eyetracking system and the sampling frequency of the eyetracker are major moderators of the performance of both techniques. At low sampling rates, samples become less frequently available, whereas at high sampling rates, sample-to-sample noise impacts the velocity estimates to a larger extent (Han et al., 2013). For comparison, at a sampling rate of 1000 Hz, the saccade detection latencies of both techniques decrease as compared to 500 Hz (boundary: *M* = 11.2 ms, *SD* = 0.96 ms; absolute velocity: *M* = 1.4 ms, *SD* = 0.08 ms), whereas false alarm rates increase drastically for absolute-velocity thresholds [*p*(FA) = 63%, *SD* = 6%; mean *d'* = 2.8, *SD* = 0.16].

Even though these traditional online saccade detection methods are often used on only one data sample, they are naturally not restricted to this definition. To enable a fair comparison to the proposed algorithm that evaluates a number of samples defined by the user (here, one to four samples), we tested whether the performance of the traditional techniques can be improved when more than one sample is taken into account (Fig. 5). Indeed, for absolute-velocity thresholds applied to two or more samples, detection accuracy increased remarkably [500 Hz: *p*(FA) = 2%, *SD* = 1%; mean *d'* = 5.5, *SD* = 0.5; 1000 Hz: *p*(FA) = 0.5%, *SD* = 0.3; mean *d'* = 5.9, *SD* = 0.25], but at the cost of an increase in saccade detection latencies (500 Hz: *M* = 8.6 ms, *SD* = 1.7 ms; 1000 Hz: *M* = 4.3 ms; *SD* = 0.53 ms). Boundary techniques, on the other hand, benefited very little from additional test samples [500 Hz: *p*(FA) = 0.3%, *SD* = 0.2%, mean *d'* = 6.4, *SD* = 0.7; 1000 Hz: *p*(FA) = 0.1%, *SD* = 0.7%, mean *d'* = 6.2, *SD* = 0.2], as their accuracy was already at ceiling for one sample, and their detection latencies only increased further (500 Hz: *M* = 18.9 ms, *SD* = 1.18 ms; 1000 Hz: 13.3 ms, *SD* = 0.95 ms).

Detection accuracy of the proposed algorithm (shapes in shades of violet in Figs. 5 and 6) remained largely unaltered across sampling frequencies [averaged across the entire tested parameter space, for 500 Hz: mean *p*(FA) = 11.7%, *SD* = 2.5%; *d'* = 5.1, *SD* = 1.0; 1000 Hz: mean *p*(FA) = 14.6%, *SD* = 3.2%; *d'* = 5.1, *SD* = 1.5; β = – 0.01, *t*(128) = – 0.06, *p* = .95], due to the adaptive adjustment of the noise level, while detection latencies decreased [500 Hz: *M* = 5.5 ms, *SD* = 2.1 ms; 1000 Hz: *M* = 3.7 ms, *SD* = 1.9 ms; β = – 1.5, *t*(128) = – 6.1, *p* < .0001].

Saccade-detection accuracy (Fig. 5b) and latency (Fig. 5c), however, strongly depended on the choice of the necessary parameters *k*, λ, and θ. First, increasing the number of samples needed above threshold *k* improved accuracy (*d'*) by 0.48 per sample [β = 0.48, *t*(128) = 8.2, *p* < .0001], but also increased latency by 1.27 ms per sample [β = 1.27, *t*(128) = 11.6, *p* < .0001]. Note that a similar effect was found above for absolute-velocity thresholds (see also Fig. 5a). Second, a higher threshold parameter λ similarly increased both accuracy [β = 0.125, *t*(128) = 10.6, *p* < .0001] and latency [β = 0.14, *t*(128) = 6.3, *p* < .0001]. Third, accepting a wider range of saccade directions (in degrees) led to a decrease of accuracy [β = – 0.004, *t*(128) = – 5.9, *p* < .0001] and latency [β = – 0.01, *t*(128) = – 6.9, *p* < .0001]. Although for saccade detection latencies all three parameters had additive effects (Fig. 5c), interactions were present for detection accuracy: The benefit of increasing the number of samples above thresholds *k* [β = – 0.07, *t*(128) = – 6.6, *p* < .0001] or restricting directions θ was smaller at higher thresholds [β = 0.0004, *t*(128) = 3.2, *p* = .002], because detection accuracy would reach ceiling (Fig.

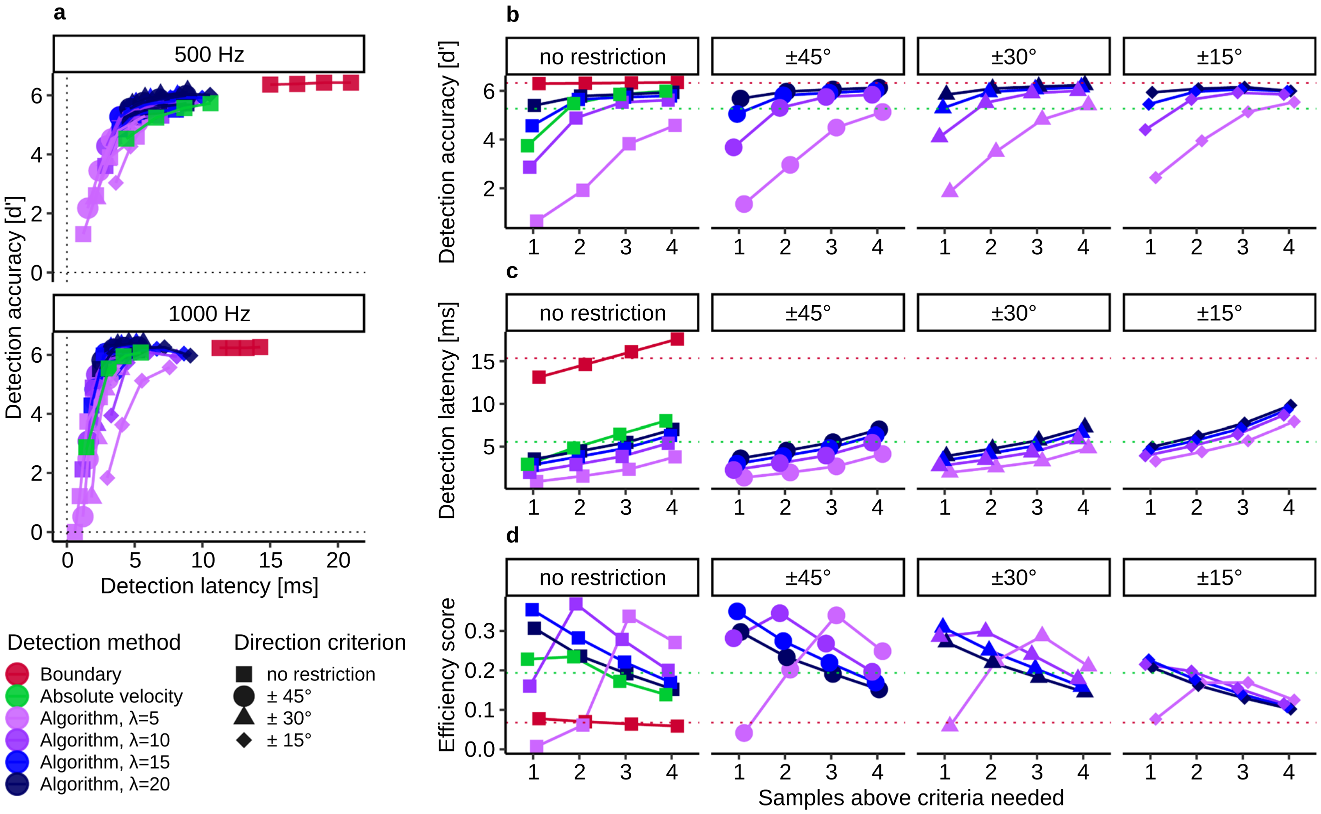

Fig. 5 Grand averages of detection performance and latency, as determined by simulation. (a) The trade-off of detection accuracy and detection latency for each sampling rate. Every dot represents the mean across all trials, including all eight tested saccade directions, with color indicating the type of detection method (and threshold factor λ) used and shape indicating the direction criterion (*θ*) used. The four connected values indicate the number of samples above threshold (*k*) needed for detection in each condition (always increasing from left to right). (b–d) Mean detection accuracy, latency, and efficiency, respectively, averaged across sampling rates for different parameter combinations (λ, *θ*, *k*). The green and red dotted reference lines indicate the average results (across samples above criteria, *k*) for the absolute-velocity thresholds and boundary techniques, respectively.

5b). Accordingly, direction restriction was more effective at low λ and low *k* [$\beta = 0.001$, $t(128) = 2.1$, $p = .043$].

To improve the understanding of this speed–accuracy trade-off, we introduced an efficiency score (Townsend & Ashby, 1978), based on the ratio of correct rejection rate and detection latency (Fig. 5d). Importantly, it becomes evident that for optimal parameter choice, the efficiency of the proposed algorithm is well above the efficiency of both the boundary and absolute-velocity techniques, even when these techniques evaluated more than one sample. With extremely conservative settings (see Figs. 5b–5d, rightmost panels), however, detection latency will be increased to a large degree, so that some saccades might not be detected in time. With regard to the optimal choice of parameters, it is important to consider the noise levels and sampling rate of the eyetracker. For our simulation, we chose two eyetrackers with similar spatial precision (RMS = 0.01 dva; SR Research, 2005, 2010), but varying sampling rates. We found that the positive effects of detection threshold λ and number of samples above threshold *k* on detection accuracy were slightly stronger when tracking at higher than at lower sampling rates [λ: $\beta = 0.047$, $t(128) = 1.99$, $p = .049$; *k*: $\beta = 0.24$, $t(128) = 2.01$, $p = .038$]. This suggests that a more conservative parameter choice is more beneficial at higher sampling rates, where increased velocity due to tracker noise is more likely to occur (see also Fig. 6, bottom row).

How do online saccade detection techniques cope with conditions in which noise is drastically increased or in which several samples are dropped? We simulated these situations by adding uncorrelated, Gaussian noise (standard deviations of up to 6 arcmin) to the data and by randomly removing data samples (up to 30%). As is shown in Fig. 6 (top row), absolute-velocity thresholds (green lines) are strongly impacted by noise [$\beta = -0.75$, $t(672) = -6.1$, $p < .0001$], as the false alarm rate reached almost 100% after adding Gaussian noise of 1.5 arcmin *SD* at 1000 Hz, reducing this technique's efficiency to virtually zero—that is, 0.0005 (*SD* = 0.0004). As predicted, at 500 Hz the impact of noise was smaller, but still an efficiency of virtually zero was reached at a Gaussian noise *SD* of 3.0 arcmin (mean efficiency = 0.001, *SD* = 0.0008). The detection accuracy of the proposed algorithm (500 Hz: mean efficiency = 0.11, *SD* = 0.012; 1000 Hz: mean efficiency =

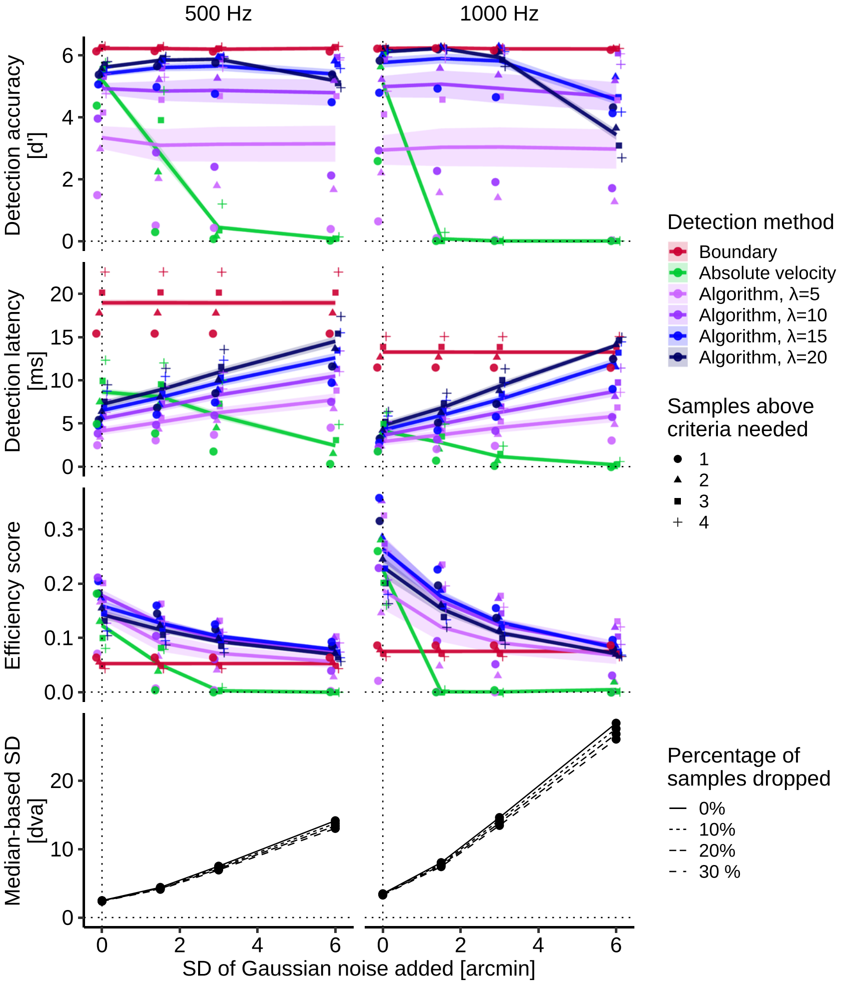

Fig. 6 *Top three rows:* Mean detection accuracy, latency, and efficiency of the three online saccade detection techniques for different noise levels (*SD* of Gaussian noise added to both sample dimensions, *x* and *y*) and sampling rates (left column = 500 Hz, right column = 1000 Hz), averaged across all levels of percentage of samples dropped. The lines represent averages across the entire tested parameter space, and symbols represent the number of samples above threshold needed to detect a saccade (k). Shaded areas indicate 95% confidence intervals. *Bottom row:* Median-based standard deviations of absolute velocity estimates used to compute velocity thresholds. Different line types represent the percentage of samples dropped.

0.15, *SD* = 0.021), on the other hand, was largely unimpaired by noise [$\lambda = 5$: $\beta = -0.03$, $t(672) = -0.4$, $p = .68$; $\lambda = 10$: $\beta = -0.04$, $t(672) = -0.6$, $p = .53$; $\lambda = 15$: $\beta = -0.12$, $t(672) = -1.7$, $p = .08$]. At a threshold factor of $\lambda = 20$, however, detection accuracy decreased starting at Gaussian noise *SD*s of 6 arcmin [$\beta = -0.3$, $t(672) = -4.1$, $p = .0004$], since the velocity thresholds were simply too high: If median-based velocity *SD*s such as 26 dva/s (Fig. 6, bottom row, right panel) were multiplied by a factor of 20, we would achieve unreasonable velocity thresholds as high as 520 dva/s, and thus miss most ongoing saccades. Furthermore, in the presence of noise and when working with lower velocity thresholds, it was beneficial to increase the number of samples that should be evaluated by the algorithm, to improve accuracy [absolute velocity: $\beta = 0.69$, $t(672) = 3.1$, $p = .002$; algorithm [$\lambda = 5$]: $\beta = 1.5$, $t(672) = 6.6$, $p < .0001$; algorithm [$\lambda = 10$]: $\beta = 0.78$, $t(672) = 3.5$, $p = .0004$].

Because the velocity threshold is estimated on the basis of the current noise level, to preserve robustness across trials and participants, higher velocity thresholds due to increased noise levels should be accompanied by increased detection latencies. Indeed, for every threshold factor λ, the saccade detection latency of the algorithm increased with noise level [$\lambda = 5$: $\beta = 0.55$, $t(672) = 19.7$, $p < .0001$; $\lambda = 10$: $\beta = 0.85$, $t(672) = 30.7$, $p < .0001$; $\lambda = 15$: $\beta = 1.19$, $t(672) = 43.1$, $p < .0001$; $\lambda = 20$: $\beta = 1.42$, $t(672) = 51.2$, $p < .0001$]. Naturally, increasing the number of samples above the criteria needed to detect the saccade also increased latency across all tested algorithms [$\beta = 1.47$, $t(672) = 24.5$, $p < .0001$].

With respect to dropped samples, boundary techniques and absolute-velocity thresholds suffered from longer detection latencies due to dropped samples [boundary: $\beta = 0.05$, $t(672) = 14.8$, $p < .0001$; absolute velocity relative to boundary: $\beta = 0.015$, $t(672) = 3.0$, $p = .003$], especially when more samples were evaluated [boundary: $\beta = 0.02$, $t(672) = 6.7$, $p < .0001$; absolute velocity relative to boundary: $\beta = 0.007$, $t(672) = 0.7$, $p = .46$]. The latency of the proposed algorithm was less affected by dropped samples than were the two traditional techniques [$\lambda = 5$: $\beta = -0.03$, $t(672) = -6.9$, $p < .0001$; $\lambda = 10$: $\beta = -0.03$, $t(672) = -6.4$, $p < .0001$; $\lambda = 15$: $\beta = -0.03$, $t(672) = -6.4$, $p < .0001$; $\lambda = 20$: $\beta = -0.03$, $t(672) = -6.7$, $p < .0001$], a result that is likely related to the interpolation of missing data samples that the algorithm performs prior to smoothing and threshold estimation. In fact, median-based standard deviations depended strongly on the noise level, but hardly on the percentage of dropped samples in the data (Fig. 6, bottom row). Although latency increased, all tested algorithms maintained their accuracy when samples were dropped [boundary: $\beta = -0.007$, $t(672) = -0.8$, $p = .44$; absolute velocity: $\beta = 0.005$, $t(672) = 0.35$, $p = .73$; algorithm [$\lambda = 5$]: $\beta = -0.01$, $t(672) = -1.0$, $p = .31$; algorithm [$\lambda = 10$]: $\beta = -0.007$, $t(672) = -0.49$, $p = .62$; algorithm [$\lambda = 15$]: $\beta = -0.003$, $t(672) = -0.24$, $p = .81$; algorithm [$\lambda = 20$]: $\beta = -0.001$, $t(672) = -0.01$, $p = .92$].

As is shown in Fig. 6 (third row), efficiency scores remained constant for the boundary technique [$\beta = -0.0003$, $t(672) = 0.001$, $p = .99$] and decreased with increasing noise levels for all velocity-based algorithms [absolute velocity: $\beta = -0.027$, $t(672) = -7.6$, $p < .0001$; algorithm [$\lambda = 5$]: $\beta = -0.017$, $t(672) = -4.9$, $p < .0001$; algorithm [$\lambda = 10$]: $\beta = -0.024$, $t(672) = -6.9$, $p < .0001$; algorithm [$\lambda = 15$]: $\beta = -0.022$, $t(672) = -6.5$, $p < .0001$; algorithm [$\lambda = 20$]: $\beta = -0.02$, $t(672) = -5.7$, $p < .0001$], but the least square means of the computed linear model revealed a considerable difference between the absolute efficiency of the tested algorithms: Across sampling rates and for the means of noise level (i.e., 2.62 arcmin), percentage of dropped samples (i.e., 15% dropped samples), and number of samples above threshold (i.e., 2.5 samples), the proposed algorithm outperformed both boundary techniques ($M = 0.065$, 95% CI [0.058, 0.071]) and absolute-velocity thresholds ($M = 0.052$, 95% CI [0.045, 0.058]) on all threshold factor levels ($\lambda = 5$: $M = 0.105$, 95% CI [0.098, 0.11]; $\lambda = 10$: $M = 0.14$, 95% CI [0.134, 0.147]; $\lambda = 15$: $M = 0.14$, 95% CI [0.134, 0.147]; $\lambda = 20$: $M = 0.12$, 95% CI [0.117, 0.13]).

Finally, in a separate simulation, we also found that the algorithm's running time (i.e., the time elapsed from invocation to return of the `mex` function in Matlab 2016b on a Dell Optiplex 3020 with an Intel i5-4590 processor running Kubuntu 18.04) on data collected for 2 s was on average 0.051 ms at a sampling rate of 500 Hz ($SD = 0.005$ ms, $N = 30{,}000$), 0.097 ms at 1000 Hz ($SD = 0.009$ ms, $N = 30{,}000$), and 0.187 ms at 2000 Hz ($SD = 0.011$ ms, $N = 30{,}000$). The algorithm's time average complexity is thus linear (and quadratic, in the worst case).

Experimental results

To present an example of application of the algorithm and to show that its application makes strictly intrasaccadic presentations well possible, we developed an objective and a perceptual test. Since our setup allowed for the co-registration of stimulus events, EEG recordings, and eyetracking at a high temporal resolution, the objective test used a photodiode to measure physical stimulus onset and offset during the saccade, in which the saccade was detected online with the proposed algorithm (see Fig. 3 for the Apparatus). In addition, as a perceptual test, we presented a Gabor patch (vertical orientation, 0.5 cpd spatial frequency, 0.5 dva *SD* Gaussian aperture) that—due to its high drift velocity (on average, 420 dva/s)—would be invisible during fixation but become detectable only if the eye was moving at a similar velocity in the same direction (Castet & Masson, 2000; Watson et al., 2017). Observers were instructed to indicate whether they perceived a Gabor patch that could be presented anywhere within a range of 8 dva around screen center and was present in 50% of all saccade and fixation trials (Fig. 4).

Online saccade detection in saccade trials worked well. Only 1% of valid trials had to be excluded due to too early or too late detection. The mean detection latency amounted to 5 ms ($SD = 2.06$ ms). Unlike the results from the simulation, however, this latency estimate still included system delays, most crucially the end-to-end sample delay of the eyetracker: According to the manufacturer, the EyeLink 1000+ with normal filter settings at a sampling rate of 2000 Hz is expected to have an average delay of 2.7 ms ($SD = 0.2$ ms; SR Research, 2013). Correcting for these delays, the mean detection latency would be around 2.3 ms. As a comparison, the mean detection latency (determined by our simulation) with the same parameters and eyetracking system, but tracking at 1000 Hz, amounted to 3.4 ms ($SD = 0.66$ ms), while remaining at a high accuracy level [$p(\text{FA}) = 0.3\%$, $SD = 0.06\%$; mean $d' = 6.1$, $SD = 0.4$]. The second crucial latency for gaze-contingent displays is the flip latency (i.e., the time when the flip occurs relative to saccade onset; see Fig. 7b), which depends on the hardware, the display frame rate used, as well as the time within the refresh cycle. In this experiment, the flip latency was on average 11 ms ($SD = 3.1$ ms), as a mean detect-to-flip delay of 6 ms ($SD = 2.46$ ms) incurred. Finally, the flip-to-display latency should be deterministic, as the ProPixx DLP projector updates its entire display with microsecond precision as soon as all video data are transferred. Indeed, the flip-to-display latency amounted to an average of 8.15 ms ($SD = 0.35$ ms), which is in line with the graphic card's refresh rate of 120 Hz. The addition of all system delays resulted in a physical stimulus onset around 19.6 ms ($SD = 3.1$ ms) after saccade onset (photodiode onset in Fig. 7b), leading to the intended intrasaccadic display right when the eye was in midflight (Fig. 7a).

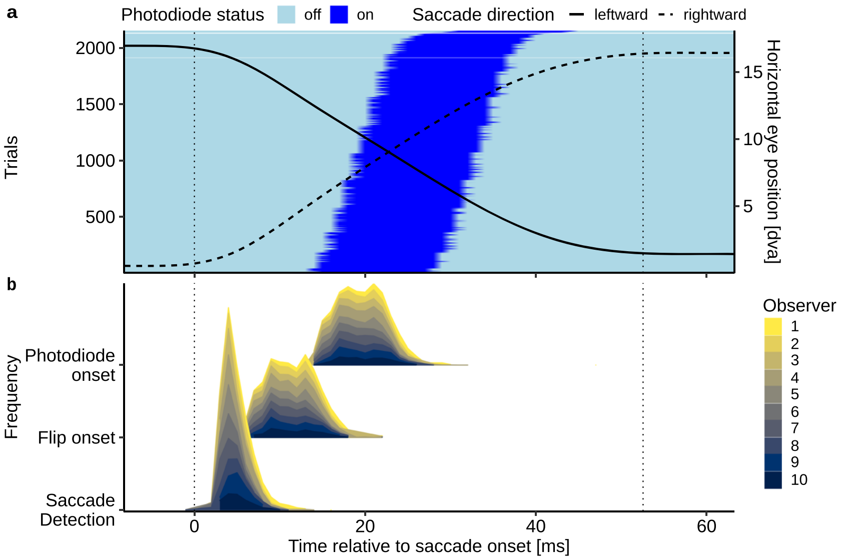

Fig. 7 Timing events in the experimental test. (a) "On" times of the photodiode (dark blue) of all saccade trials, displayed and sorted according to their time relative to the onset and offset of the saccade (dotted vertical lines). The solid and dashed lines represent the mean horizontal saccade trajectories of leftward and rightward saccades over time (smoothed by a univariate general additive model). (b) Distributions of detection, flip, and photodiode onset times relative to the onset of the saccade. Shadings indicate data from different observers.

If the intrasaccadic stimulus presentation was indeed successful, then observers should have been able to detect the rapidly drifting Gabor during saccades and not during fixation, as only an ongoing eye movement could decrease the (relative) velocity of the stimulus on the retina to the point where it would become perceivable. When observers were fixating, their detection performance was not significantly different from chance level [mean d' = 0.06, SD = 0.14; $t(9)$ = 1.28, p = .23], suggesting that the Gabor stimulus drifting at an average velocity of 419 dva/s (SD = 61.4 dva/s) was indeed invisible when the eye was not moving (Fig. 8a). In contrast, we found that the stimuli were readily detected during saccades, as performance drastically improved relative to the fixation condition [mean d' = 2.94, SD = 1.1; $F(1, 9)$ = 59.6, η^2 = .79, p < .0001].

To further explore the potential effect of retinal velocity on detection performance, we computed each trial's mean retinal velocity during stimulus presentation (see the Analysis section). We found that the retinal velocity was on average 416 dva/s (SD = 45 dva/s) during fixation, whereas it was reduced to 68 dva/s (SD = 68.8 dva/s) during saccades. Note that the mean retinal velocity during saccades was in most cases positive, because presentation of the stimulus extended into the deceleration phase of the saccadic velocity profile (Fig. 7a).

A logistic mixed-effect regression (Bates et al., 2015) revealed not only a large increase of correct responses in the saccade condition (β = 3.0, t = 5.7, 95% CI [2.13, 4.13]), but also a significant negative effect of retinal velocity (β = – 0.12, t = – 2.41, 95% CI [– 0.23, – 0.016]), suggesting that, across both conditions, higher retinal velocity was associated with lower task performance. An interaction between condition and retinal velocity was also significant (β = – 0.20, t = – 2.06, 95% CI [– 0.4, – 0.01]). Because overall performance was much lower in the fixation condition, this interaction suggests that the effect of retinal velocity was exclusive to the saccade condition (Fig. 8b). To check whether the difference between the fixation and saccade conditions was mediated by a difference in retinal eccentricity at the time of stimulus presentation, we also computed the mean retinal position of the stimulus. In the saccade condition, stimuli had an average 1-dva offset of horizontal eccentricity against the saccade direction ($M_{x,\mathrm{sac,left}}$ = 1.31 dva, $SD_{x,\mathrm{sac,left}}$ = 1.1 dva; $M_{x,\mathrm{sac,right}}$ = – 1.15 dva, $SD_{x,\mathrm{sac,right}}$ = 1.2 dva) relative to the fixation condition ($M_{x,\mathrm{fix,left}}$ = 0.30 dva, $SD_{x,\mathrm{fix,left}}$ = 0.54 dva; $M_{x,\mathrm{fix,right}}$ = – 0.37 dva, $SD_{x,\mathrm{fix,right}}$ = 0.62 dva), which can be explained by the fact that stimulus presentation extended into the second half of the saccade in most cases—that is, when the eye had already passed the screen center (the mean horizontal presentation location). To determine whether this slight difference in eccentricity had any effect on task performance, we added absolute horizontal and vertical eccentricity to the logistic mixed-effect regression. We found an increase in log-likelihood that was not significant [ΔLL = + 8.8, $\chi^2(21)$ = 17.51, p = .68], suggesting that retinal eccentricity played, if any, a subordinate role in our task.

Discussion

Timing is crucial when studying visual perception around the time of saccades, especially when manipulating stimulus configurations gaze-contingently with the onset of a saccade. In

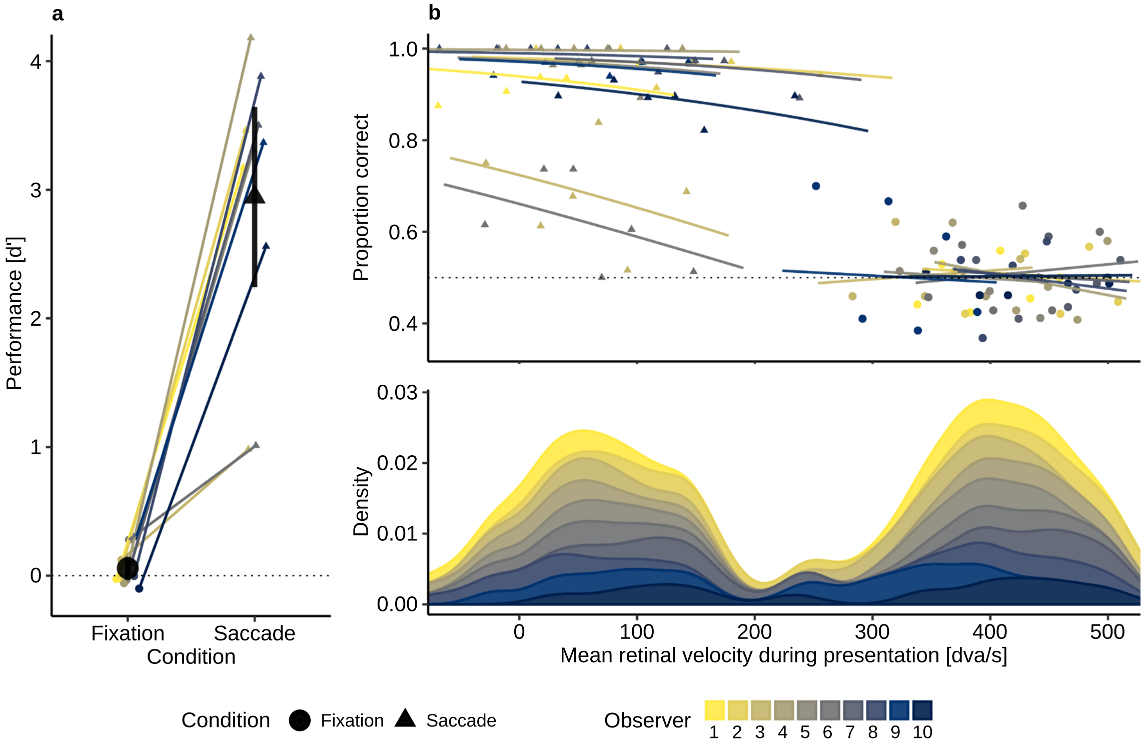

Fig. 8 Behavioral results from the perceptual test. (a) Stimulus detection performance (d') in the fixation and saccade conditions for individual observers. The large black circles and triangles represent group means for the fixation and saccade conditions. Error bars indicate 95% confidence intervals, based on ± 2 *SEM*s. (b) *Upper panel:* Model fits from the logistic mixed-effect regression with random intercepts and slopes for observers. The points indicate mean proportions correct in six equal-sized bins of retinal velocity (i.e., mean eye velocity during stimulus presentation subtracted from stimulus velocity) per condition per observer. *Lower panel:* Distribution of retinal velocities for each observer for the fixation (right peak) and saccade (left peak) conditions.

gaze-contingent experimental paradigms, various sources of latency have to be considered, ranging from eyetracker delays, saccade detection latencies, graphic card refreshes, video delays, and monitor reaction time. Whereas most of these latencies can be reduced by enhancing hardware capabilities, a more efficient online saccade detection algorithm improves timing performance independently of the experimental setup. Unfortunately, the most commonly used online detection techniques—that is, the spatial boundary technique and the absolute-velocity threshold—have significant shortcomings: The former provides reliable but late saccade detection, whereas the latter is fast but struggles with reliability, especially at higher sampling rates or slightly increased noise levels. Inspired by a widely used algorithm for (micro)saccade detection (Engbert & Kliegl, 2003; Engbert & Mergenthaler, 2006), we developed a velocity-based online saccade detection algorithm that incorporates both algorithms' strong points: It allows for rapid saccade detection due to low velocity thresholds, it is robust against noise by applying smoothing, adaptive adjustment of velocity thresholds, and an optional direction criterion, and it allows the user to flexibly specify more liberal or more conservative detection criteria. Across various gaze-contingent experimental paradigms, as well as in non-scientific applications, this open-source algorithm could help create comparable and reproducible results by (1) avoiding timing problems (due to its early saccade detection) and (2) increasing stability (due to its increased robustness against noise).

We validated the algorithm, as well as the boundary and the absolute-velocity technique, in a large-scale simulation (> 30,000 saccades). We found that the algorithm provided considerably earlier saccade detection than boundary techniques (up to 10 ms or more, depending on sampling rate), which was more similar to (although in most cases slightly slower than) the latency of the absolute-velocity technique. Crucially, the algorithm's accuracy in online saccade detection was on par with the boundary technique and significantly larger than that of the absolute-velocity technique, especially at high sampling rates and even when the number of evaluated samples was accounted for. Moreover, when corrupting the collected data with noise, absolute-velocity techniques suffered from a drastic increase in false alarms, whereas the proposed algorithm maintained its detection accuracy by updating its velocity threshold and exhibited significantly larger efficiency scores than both traditional techniques. This is an important result and prerequisite for the use during eyetracking experiments, because many factors that vary throughout the experiment or on a trial-by-trial basis, such as pupil diameter, time since last calibration, head movements, gaze eccentricity, or marker visibility, may influence accuracy and precision of recordings (Nyström, Andersson, Holmqvist, & Van De Weijer, 2013) and thus alter the noise level. Since the algorithm updates velocity thresholds with every new incoming data sample, various negative influences on detection accuracy can be accounted for. For instance, even small drifts in measured gaze positions, such as those occurring during fixation when head-mounted eyetrackers shift their position slightly relative to the participant's eyes (which can lead to erroneous saccade detections when using boundary techniques), would simply lead to elevated velocity thresholds that preserve detection accuracy. At the same time, the incentive to achieve high data quality remains, as increased noise levels may still have a negative impact on detection latency. The idea of employing adaptive thresholds for online saccade detection to reduce susceptibility to noise is not entirely new. For example, in their PyGaze toolbox, Dalmaijer, Mathôt, and Van der Stigchel (2014) have used an additional root-mean-square (RMS) criterion along with user-defined thresholds for velocity and acceleration. Unlike that approach, our proposed algorithm depends neither on absolute threshold definitions nor on a calibration of RMS, because thresholds are computed on the basis of all preceding samples collected during a trial and thus efficiently capture trial-to-trial variance. Moreover, its nonblocking implementation in the programming language C allows for flexible, multi-platform usage (e.g., Python, Matlab, Octave), and yields high speed: Running the algorithm in real time is feasible because a large number of collected samples, such as those collected from eyetrackers sampling at 2000 Hz, can be processed within microseconds. However, due to its linear time complexity, it cannot be guaranteed that the algorithm will finish prior to any given deadline. For example, if the algorithm were run on an unnecessarily large number of samples, such as 4,000,000, then it would take ~187 ms (based on the results of our simulation), exceeding by far the duration of a saccade.

We found that the detection criterion and subsequent performance of the algorithm strongly depends on the parameters supplied: More conservative settings (i.e., higher threshold factor, more samples above threshold, a tighter range of accepted directions) will improve detection accuracy to a maximum, but will come at the cost of increased detection latency, and in the worst case—as is shown by the high-noise conditions—may lead to abnormally high velocity thresholds that will make saccade detection impossible. We thus suggest a careful weighting of parameters depending on the experimental setup and paradigm used. For example, when using a threshold factor of $\lambda = 5$, it makes sense to have at least three samples above threshold to detect a saccade. Indeed, if the algorithm were used to detect microsaccades during an experiment, low thresholds should be used, as eye velocities during microsaccades are not as high as those during saccades, whereas three or more samples should be evaluated (for a successful application with a different implementation, see Yuval-Greenberg, Merriam, & Heeger, 2014). If, however, a threshold factor of $\lambda = 20$ were used, one sample above threshold might often be enough to reliably detect a saccade without adding significant additional delay. In pilot work with the TrackPixx3 (VPixx Technologies, 2017), we also found that binocular online saccade detection (running the detection algorithm on each eye separately) allows for lower threshold factors, as the probability that velocity thresholds are exceeded simply due to noise in both eyes simultaneously is smaller than for one eye only. The choice of threshold also depends on the noise level and sampling rate of the eyetracker in use, as a higher sampling rate can inflate velocity estimates (Han et al., 2013). In some systems, such as the EyeLink 1000+ (SR Research, 2013), these two variables are not independent: With deactivated heuristic filters, RMS noise amounts to 0.02 dva at 1000 Hz and to 0.03 dva (monocular tracking) and 0.04 dva (binocular tracking) at 2000 Hz. Additional filter levels supplied by the manufacturer can reduce these noise levels, but introduce additional end-to-end sample delay (also depending on sampling rate). It is thus important to understand that the parameter choice for optimal online saccade detection performance is intrinsically dependent on both the recording settings and the nature of the task. For instance, a direction restriction can only be applied in paradigms, in which it is certain or at least very likely that the participant will indeed make a saccade in a given direction. In case of a two-alternative forced choice paradigm, it would be possible to call the algorithm twice, each time with different direction criteria, but it would be impossible to apply a direction criterion in a free viewing context. Ultimately, it remains an advantage that the experimenter is able to fine-tune the detection criterion according to the relative costs for longer detection latencies or for an increased false alarm rate incurring in a specific task.

With the introduction of the objective and perceptual test experiment, we provided a real-world example of a gaze-contingent paradigm in which timing was crucial, in this case for the intrasaccadic presentation to be successful. This test builds on the finding that rapidly drifting or flickering gratings that are invisible during fixation can be rendered visible due to the reduction of retinal velocity occurring when the eye moves across them (Castet & Masson, 2000; Mathôt, Melmi, & Castet, 2015; García-Pérez & Peli, 2011). We used a projection system operating at submillisecond temporal resolution to briefly display a rapidly drifting Gabor patch entirely during the saccade, and we asked observers to detect it. We established that the Gabor patch was indeed largely invisible during fixation. It was absolutely crucial in this task, therefore, that the stimulus was presented while the eye was in midflight to achieve an approximate match of the velocities of stimulus and eye. Both observers' high task performance in the saccade condition (perceptual test) and recordings from a photodiode (objective test) confirmed that despite all possible system delays strictly intrasaccadic presentations with a physical onset as early as 20 ms after saccade onset were well possible. At the same time, only 1% of all trials had to be removed from analysis because presentations did not happen strictly during the saccade—for example, due to late detections or erroneous detections while fixating (see the Method section). Importantly, the described perceptual test can be a valuable tool to determine whether a presentation is actually intrasaccadic when photodiode measurements are unavailable: If a drifting stimulus that is otherwise undetectable during fixation can be reliably discriminated during a saccade, then its presentation must be (at least partly) intrasaccadic. By examining discrimination performance as a function of time relative to the measured saccade detection, the timing of a gaze-contingent paradigm can be systematically investigated. Note that although we displayed the drifting Gabor patch using a projector with a frame rate of 1440 Hz, other studies have achieved similar presentations using much lower frame rates, such as 122 Hz (García-Pérez & Peli, 2011), 150 Hz (Mathôt et al., 2015), or 160 Hz (Castet & Masson, 2000), suggesting that the perceptual test should work with any gamma-calibrated laboratory monitor running at frame rates of 120 Hz or more.

But the finding is interesting for two other reasons. First, it shows that if a stimulus has high contrast, is optimized for the high velocity of saccades (Castet & Masson, 2000; Deubel et al., 1987; García-Pérez & Peli, 2011; Mathôt et al., 2015; Schweitzer, Watson, Watson, & Rolfs, 2019), and is not affected by pre- and postsaccadic masking (Campbell & Wurtz, 1978; Castet, 2010), then it is readily detectable, if not highly salient. Second, the finding indicates that timing during and around saccades matters. It is widely assumed that visual processing is suppressed during and around the time of saccades (Burr, Holt, Johnstone, & Ross, 1982; Burr, Morrone, & Ross, 1994; Ross, Morrone, Goldberg, & Burr, 2001). This is the reason why many trans-saccadic paradigms relying on gaze-contingent manipulations assume that as long as a display change falls within the window of saccadic suppression—which precedes saccade onset by up to 100 ms and exceeds the saccade duration by up to 50 ms (Diamond, Ross, & Morrone, 2000; Volkmann, 1986; Volkmann, Riggs, White, & Moore, 1978)—it is neither noticed nor processed. There is, however, converging evidence that stimuli undergoing saccadic suppression can shape postsaccadic perception (Watson & Krekelberg, 2009), and that the relative timing of a stimulus relative to saccade offset drastically changes both the appearance of that stimulus and its likelihood of being consciously perceived (Balsdon, Schweitzer, Watson, & Rolfs, 2018; Bedell & Yang, 2001; Campbell & Wurtz, 1978; Duyck, Collins, & Wexler, 2016; Matin, Clymer, & Matin, 1972). There is also evidence that brief intrasaccadic flashes were able to drive saccadic adaptation when presented during the deceleration phase of an ongoing saccade (Panouillères et al., 2016). Although for many experiments and the conclusions drawn from them intrasaccadic display changes may not be absolutely crucial, it is important to be aware of the possibility that an intended intrasaccadic change might in fact be a nonintended postsaccadic change due to insufficient control of timing or hidden latencies in the hardware. Earlier saccade detection can alleviate this risk.

We conclude that implementing efficient gaze-contingent display changes across saccades can be tricky, owing to a range of system latencies that have an impact on a paradigm's timing behavior. We as experimenters need to examine these latencies closely in order to draw the right conclusions from our results. Early online saccade detection can assist greatly in this task, as it saves valuable time for the setup to perform the intended (trans-saccadic) changes, but it comes at the cost of reduced online saccade detection accuracy—especially at higher noise levels—making it ultimately harder to smoothly collect data. The algorithm proposed here outperforms traditional detection methods in speed and accuracy, while adjusting detection thresholds in response to increased noise levels. These properties make it a reliable tool even when collecting data under suboptimal recording circumstances, as well as computationally feasible to use for the online scenario, due to its near real-time processing and linear complexity. Finally, the open-source availability of the code leaves it open for researchers to use and adapt the algorithm to their specific needs, making it a versatile tool for the field of active vision.

Acknowledgements We acknowledge the significant contributions of Ralf Engbert, Konstantin Mergenthaler, Petra Sinn, and Hans Trukenbrod for making the code of their microsaccade detection toolbox publicly available, as well as Nicolas Devillard for the excellent ANSI C implementations and comparisons of different median search algorithms (http://ndevilla.free.fr/median/median/index.html). R.S. was supported by the Studienstiftung des deutschen Volkes and the Berlin School of Mind and Brain. M.R. was supported by the Deutsche Forschungsgemeinschaft (DFG, grants RO3579/2-1, RO3579/8-1, and RO3579/10-1).

Availability Implementations of the proposed algorithm in C, Python, and Matlab are available on Github: https://github.com/richardschweitzer/OnlineSaccadeDetection.

Open practices statement The saccade data and code used for simulations, data collected throughout the experimental test, experimental code, and data analysis scripts are available on the Open Science Framework: https://osf.io/3pck5/. The experimental test was not preregistered.

Author contributions R.S. implemented the algorithm and ran simulations. Validation procedure was conceptualized by R.S. and M.R. The experimental test was designed, run, and analyzed by R.S. under M.R.'s supervision. R.S. drafted the manuscript, and M.R. provided critical revisions.

References

Arabadzhiyska, E., Tursun, O. T., Myszkowski, K., Seidel, H.-P., & Didyk, P. (2017). Saccade landing position prediction for gaze-contingent rendering. *ACM Transactions on Graphics*, *36*, 50.

Balsdon, T., Schweitzer, R., Watson, T. L., & Rolfs, M. (2018). All is not lost: Post-saccadic contributions to the perceptual omission of intra-saccadic streaks. *Consciousness and Cognition*, *64*, 19–31.

Bates, D., Mächler, M., Bolker, B., & Walker, S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, *67*, 1–48. doi:https://doi.org/10.18637/jss.v067.i01

Bedell, H. E., & Yang, J. (2001). The attenuation of perceived image smear during saccades. *Vision Research*, *41*, 521–528.

Burr, D. C., Holt, J., Johnstone, J. R., & Ross, J. (1982). Selective depression of motion sensitivity during saccades. *Journal of Physiology*, *333*, 1–15.

Burr, D. C., Morrone, M. C., & Ross, J. (1994). Selective suppression of the magnocellular visual pathway during saccadic eye movements. *Nature*, *371*, 511–513. doi:https://doi.org/10.1038/371511a0

Campbell, F. W., & Wurtz, R. H. (1978). Saccadic omission: Why we do not see a grey-out during a saccadic eye movement. *Vision Research*, *18*, 1297–1303.

Castet, E. (2010). Perception of intra-saccadic motion. In U. J. Ilg & G. S. Masson (Eds.), Dynamics of visual motion processing (pp. 213–238). Berlin, Germany: Springer.

Castet, E., Jeanjean, S., & Masson, G. S. (2002). Motion perception of saccade-induced retinal translation. *Proceedings of the National Academy of Sciences*, *99*, 15159–15163.

Castet, E., & Masson, G. S. (2000). Motion perception during saccadic eye movements. *Nature Neuroscience*, *2*, 177–183.

Collewijn, H., Erkelens, C. J., & Steinman, R. M. (1988). Binocular coordination of human horizontal saccadic eye movements. *Journal of Physiology*, *404*, 157–182.

Collins, T., Rolfs, M., Deubel, H., & Cavanagh, P. (2009). Post-saccadic location judgments reveal remapping of saccade targets to non-foveal locations. *Journal of Vision*, *9*(5), 29. doi:https://doi.org/10.1167/9.5.29

Cornelissen, F. W., Peters, E. M., & Palmer, J. (2002). The Eyelink Toolbox: Eye tracking with MATLAB and the Psychophysics Toolbox. *Behavior Research Methods, Instruments, & Computers*, *34*, 613–617. doi:https://doi.org/10.3758/BF03195489

Dalmaijer, E. S., Mathôt, S., & Van der Stigchel, S. (2014). PyGaze: An open-source, cross-platform toolbox for minimal-effort programming of eyetracking experiments. *Behavior Research Methods*, *46*, 913–921. doi:https://doi.org/10.3758/s13428-013-0422-2

Delorme, A., & Makeig, S. (2004). EEGLAB: An open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *Journal of Neuroscience Methods*, *134*, 9–21. doi:https://doi.org/10.1016/j.jneumeth.2003.10.009

Deubel, H., Elsner, T., & Hauske, G. (1987). Saccadic eye movements and the detection of fast-moving gratings. *Biological Cybernetics*, *57*, 37–45.

Deubel, H., Schneider, W. X., & Bridgeman, B. (1996). Postsaccadic target blanking prevents saccadic suppression of image displacement. *Vision Research*, *36*, 985–996. doi:https://doi.org/10.1016/0042-6989(95)00203-0

Diamond, M. R., Ross, J., & Morrone, M. C. (2000). Extraretinal control of saccadic suppression. *Journal of Neuroscience*, *20*, 3449–3455.

Dimigen, O., Sommer, W., Hohlfeld, A., Jacobs, A. M., & Kliegl, R. (2011). Coregistration of eye movements and EEG in natural reading: Analyses and review. *Journal of Experimental Psychology: General*, *140*, 552–572. doi:https://doi.org/10.1037/a0023885

Dorr, M., Martinetz, T., Gegenfurtner, K. R., & Barth, E. (2010). Variability of eye movements when viewing dynamic natural scenes. *Journal of Vision*, *10*(10), 28. doi:https://doi.org/10.1167/10.10.28

Duyck, M., Collins, T., & Wexler, M. (2016). Masking the saccadic smear. *Journal of Vision*, *16*(10), 1. doi:https://doi.org/10.1167/16.10.1

Engbert, R., & Kliegl, R. (2003). Microsaccades uncover the orientation of covert attention. *Vision Research*, *43*, 1035–1045. doi:https://doi.org/10.1016/S0042-6989(03)00084-1

Engbert, R., & Mergenthaler, K. (2006). Microsaccades are triggered by low retinal image slip. *Proceedings of the National Academy of Sciences*, *103*, 7192–7197.

Engbert, R., Rothkegel, L., Backhaus, D., & Trukenbrod, H. A. (2016). *Evaluation of velocity-based saccade detection in the smi-etg 2W system* [Technical Report]. Retrieved from http://read.psych.uni-potsdam.de/attachments/article/156/TechRep-16-1-Engbert.pdf

García-Pérez, M. A., & Peli, E. (2011). Visual contrast processing is largely unaltered during saccades. *Frontiers in Psychology*, *2*, 247. doi:https://doi.org/10.3389/fpsyg.2011.00247

Han, P., Saunders, D. R., Woods, R. L., & Luo, G. (2013). Trajectory prediction of saccadic eye movements using a compressed exponential model. *Journal of Vision*, *13*(8), 27. doi:https://doi.org/10.1167/17.8.4

Higgins, E., & Rayner, K. (2015). Transsaccadic processing: Stability, integration, and the potential role of remapping. *Attention, Perception, & Psychophysics*, *77*, 3–27. doi:https://doi.org/10.3758/s13414-014-0751-y

Hollingworth, A., Richard, A. M., & Luck, S. J. (2008). Understanding the function of visual short-term memory: Transsaccadic memory, object correspondence, and gaze correction. *Journal of Experimental Psychology. General*, *137*, 163–181. doi:https://doi.org/10.1037/0096-3445.137.1.163

Kalogeropoulou, Z., & Rolfs, M. (2017). Saccadic eye movements do not disrupt the deployment of feature-based attention. *Journal of Vision*, *17*(8), 4. doi:https://doi.org/10.1167/17.8.4

Kleiner, M., Brainard, D., & Pelli, D. (2007). What's new in Psychtoolbox-3? *Perception*, *36*(ECVP Abstract Suppl), 14.

Mathôt, S., Melmi, J., & Castet, E. (2015). Intrasaccadic perception triggers pupillary constriction. *PeerJ*, *3*, e1150.

Matin, E., Clymer, A. B., & Matin, L. (1972). Metacontrast and saccadic suppression. *Science*, *178*, 179–182.

Melcher, D., & Colby, C. L. (2008). Trans-saccadic perception. *Trends in Cognitive Sciences*, *12*, 466–473. doi:https://doi.org/10.1016/j.tics.2008.09.003

Nyström, M., Andersson, R., Holmqvist, K., & Van De Weijer, J. (2013). The influence of calibration method and eye physiology on eyetracking data quality. *Behavior Research Methods*, *45*, 272–288. doi:https://doi.org/10.3758/s13428-012-0247-4

Panouillères, M. T., Gaveau, V., Debatisse, J., Jacquin, P., LeBlond, M., & Pélisson, D. (2016). Oculomotor adaptation elicited by intra-saccadic visual stimulation: Time-course of efficient visual target perturbation. *Frontiers in Human Neuroscience*, *10*, 91. doi:https://doi.org/10.3389/fnhum.2016.00091

Pelli, D. G. (1997). The VideoToolbox software for visual psychophysics: Transforming numbers into movies. *Spatial Vision*, *10*, 437–442. doi:https://doi.org/10.1163/156856897X00366

Poth, C. H., Foerster, R. M., Behler, C., Schwanecke, U., Schneider, W. X., & Botsch, M. (2018). Ultrahigh temporal resolution of visual presentation using gaming monitors and g-sync. *Behavior Research Methods*, *50*, 26–38.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (2007). Numerical recipes: The art of scientific computing (3rd ed.). Cambridge, UK: Cambridge University Press.

Prime, S. L., Vesia, M., & Crawford, J. D. (2011). Cortical mechanisms for trans-saccadic memory and integration of multiple object features. *Philosophical Transactions of the Royal Society B*, *366*, 540–553.

Rayner, K. (1975). The perceptual span and peripheral cues in reading. *Cognitive Psychology*, *7*, 65–81.

Ross, J., Morrone, M. C., Goldberg, M. E., & Burr, D. C. (2001). Changes in visual perception at the time of saccades. *Trends in Neurosciences*, *24*, 113–121.

Schweitzer, R., & Rolfs, M. (2017). Intra-saccadic motion streaks as a cue to the localization of objects across eye movements [Abstract]. *Journal of Vision*, *17*(10), 918. doi:https://doi.org/10.1167/17.10.918

Schweitzer, R., Watson, T., Watson, J., & Rolfs, M. (2019). The joy of retinal painting: A build-it-yourself device for intrasaccadic presentations. *Perception*, *48*, 1020–1025. doi:https://doi.org/10.1177/0301006619867868

SR Research. (2005). EyeLink II user manual, version 2.14. Mississauga, ON: SR Research Ltd.

SR Research. (2010). EyeLink 1000 user manual, version 1.5.2. Mississauga, ON: SR Research Ltd.

SR Research. (2013). EyeLink 1000 plus user manual, version 1.0.12. Mississauga, ON: SR Research Ltd.

Szinte, M., & Cavanagh, P. (2011). Spatiotopic apparent motion reveals local variations in space constancy. *Journal of Vision*, *11*(2), 4. doi:https://doi.org/10.1167/11.2.4

Tobii Technology AB. (2010). *Timing guide for Tobii eye trackers and eye tracking software* [Technical Report]. Retrieved from https://www.tobiipro.com/siteassets/tobii-pro/learn-and-support/design/eye-tracker-timing-performance/tobii-eye-tracking-timing.pdf

Townsend, J. T., & Ashby, F. G. (1978). Methods of modeling capacity in simple processing systems. In J. N. J. Castellan & F. Restle (Eds.), Cognitive theory (Vol. 3, pp. 199–239). New York, NY: Erlbaum.

Townsend, J. T., & Ashby, F. G. (1983). *Stochastic modeling of elementary psychological processes*. Cambridge University Press Archive.

Volkmann, F. C. (1986). Human visual suppression. *Vision Research*, *26*, 1401–1416.

Volkmann, F. C., Riggs, L. A., White, K. D., & Moore, R. K. (1978). Contrast sensitivity during saccadic eye movements. *Vision Research*, *18*, 1193–1199.

VPixx Technologies. (2017). TRACKPIXX3 hardware manual version 1.0. Saint-Bruno, QC: VPixx Technologies Inc.

Watson, T. L., & Krekelberg, B. (2009). The relationship between saccadic suppression and perceptual stability. *Current Biology*, *19*, 1040–1043.

Watson, T., Schweitzer, R., Castet, E., Ohl, S., & Rolfs, M. (2017). Intra-saccadic localisation is consistently carried out in world-centered coordinates [Abstract]. *Journal of Vision*, *17*(10), 1276. doi:https://doi.org/10.1167/17.10.1276

Wolf, C., & Schütz, A. C. (2015). Trans-saccadic integration of peripheral and foveal feature information is close to optimal. *Journal of Vision*, *15*(16), 1. doi:https://doi.org/10.1167/15.16.1

Yuval-Greenberg, S., Merriam, E. P., & Heeger, D. J. (2014). Spontaneous microsaccades reflect shifts in covert attention. *Journal of Neuroscience*, *34*, 13693–13700. doi:https://doi.org/10.1523/JNEUROSCI.0582-14.2014

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.