Biomedical Signal Processing and Control 18 (2015) 145–152

Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

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

Detection of fixations and smooth pursuit movements in high-speed eye-tracking data

Linnéa Larsson [a,*], Marcus Nyström [b], Richard Andersson [b], Martin Stridh [a]

[a] Department of Biomedical Engineering, Lund University, Box 118, SE-221 00 Lund, Sweden
[b] Lund University Humanities Laboratory, Lund University, Box 201, SE-221 00 Lund, Sweden

ARTICLE INFO

Article history:
Received 27 June 2014
Received in revised form
25 November 2014
Accepted 15 December 2014
Available online 24 January 2015

Keywords:
Signal processing
Eye-tracking
Smooth pursuit

ABSTRACT

A novel algorithm for the detection of fixations and smooth pursuit movements in high-speed eye-tracking data is proposed, which uses a three-stage procedure to divide the intersaccadic intervals into a sequence of fixation and smooth pursuit events. The first stage performs a preliminary segmentation while the latter two stages evaluate the characteristics of each such segment and reorganize the preliminary segments into fixations and smooth pursuit events. Five different performance measures are calculated to investigate different aspects of the algorithm's behavior. The algorithm is compared to the current state-of-the-art (I-VDT and the algorithm in [11]), as well as to annotations by two experts. The proposed algorithm performs considerably better (average Cohen's kappa 0.42) than the I-VDT algorithm (average Cohen's kappa 0.20) and the algorithm in [11] (average Cohen's kappa 0.16), when compared to the experts' annotations.

1. Introduction

Measurement of eye movements is an important tool in basic research in, e.g., visual attention, perception, cognition, and medicine. In studies of visual attention and perception, eye movements are used to investigate, e.g., how the focus of our attention is chosen depending on the content of an image [1], how objects are identified [2], and how decisions are made [3]. In medicine, eye tracking is employed in studies investigating the functionality of the brain, e.g., in patients with schizophrenia [4].

Until recently, the majority of eye-tracking studies have used static stimuli, e.g., images and texts. The two most common types of eye movements when viewing static stimuli are *fixations* and *saccades*. Fixations are periods when the eye is more or less still, while saccades are fast movements between the fixations that take the eyes from one object of interest to the next. Currently, the interest in dynamic stimuli is growing and it is becoming increasingly common to conduct studies where video clips are used as stimuli [5]. The type of eye movement called *smoothpursuit* occurs when the eyes are following a moving object [6]. Traditionally, algorithms have been developed for signals recorded during static stimuli, i.e., developed to detect fixations and saccades. When smooth pursuit movements are not considered by an algorithm, they will be spread into the other types of detected eye movements and make the interpretation of these difficult. Smooth pursuit movements may for instance be erroneously classified as very long fixations interspersed with very short saccades [7].

Many of the measures that earlier have been used to investigate eye movements during image viewing are based on the detection of fixations and their properties, e.g., fixation duration and number of fixations [8]. When dynamic stimuli are used, these fixation measures are still of interest. However, in order to be able to investigate and draw well-founded conclusions from fixations in data where smooth pursuit movements are present, a robust algorithm for separation of fixations and smooth pursuit movements is needed.

Since the signal characteristics of fixations and smooth pursuit movements are overlapping [9], classification of fixations in the presence of smooth pursuit movements is a difficult task [5,10]. The task is also different depending on whether the algorithm is intended for analysis of data recorded with a high or low sampling frequency, and for real-time or offline processing. Classification of data with different sampling frequencies require different event detection methods, mainly due to differences in the level of high frequency noise.

In [10], three algorithms for detection of fixations, saccades, and smooth pursuit movements were evaluated: a velocity based algorithm with two velocity thresholds (I-VVT), a velocity and

* Corresponding author. Tel.: +46 46 222 75 24.
E-mail addresses: linnea.larsson@bme.lth.se (L. Larsson), marcus.nystrom@humlab.lu.se (M. Nyström), richard.andersson@humlab.lu.se (R. Andersson), martin.stridh@bme.lth.se (M. Stridh).

http://dx.doi.org/10.1016/j.bspc.2014.12.008
1746-8094/

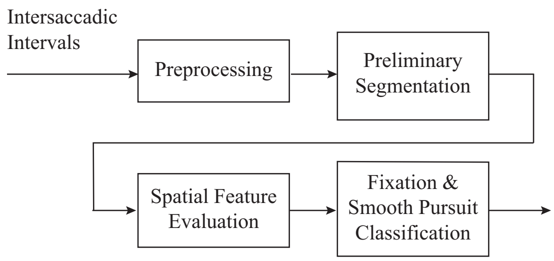

Fig. 1. Overview of the proposed algorithm.

movement pattern based algorithm (I-VMP), and a velocity and dispersion based algorithm (I-VDT). All algorithms were evaluated with data recorded using the EyeLink 1000 from SR Research. The stimuli consisted of dots moving with different speeds and different directions. The results showed that the most successful method was the I-VDT, which used a combination of velocity and dispersion thresholds.

Another algorithm, proposed in [11], employed principal component analysis in combination with a velocity threshold to distinguish between saccades, fixations, and smooth pursuit movements. The algorithm was used to analyze saccades in humans and monkeys watching short video clips, but the performance of the algorithm was not evaluated in detail. In the following, the algorithm proposed in [11] is referred to as I-PCA.

A completely different method, intended for real-time detection of smooth pursuit movements using a low-speed mobile eye-tracker was proposed in [12]. The method used a set of features and a k-nearest neighbor classifier in order to distinguish between smooth pursuit movements and the remaining parts of the data. The performance of the algorithm was evaluated using data recorded with stimuli where a dot was moving over the screen in different speeds and different directions. The results showed that a combination of features that capture temporal aspects of smooth pursuit movements was a successful detection method.

In this work, the focus is on offline processing of fixations and smooth pursuit movements in data recorded using a high-speed eye-tracker. The paper consists of two main parts: Firstly, an algorithm for classification of fixations and smooth pursuit movements is developed for eye-tracking data when dynamic stimuli are used, and secondly, a detailed evaluation is performed, where the performance of the algorithm is evaluated from different aspects.

2. Methods

A schematic overview of the proposed algorithm for detection of fixations and smooth pursuit movements is shown in Fig. 1. The algorithm is applied to the *intersaccadic intervals*, i.e., the intervals between the detected saccades, PSO, and blinks, and comprises three stages where the first stage performs a preliminary segmentation while the latter two evaluate the characteristics of each such segment and reorganize the preliminary segments into fixations and smooth pursuit events. In this paper, the intersaccadic intervals are identified using the algorithm in [13].

2.1. Preprocessing

In order to avoid any influence of adjacent saccades or PSO, the intersaccadic intervals are preprocessed. Since neither fixations nor smooth pursuit movements physiologically can have a velocity higher than 100°/s [14], the sample-to-sample velocities of the intervals are computed and all samples in the beginning and/or end of each interval exceeding this threshold are removed.

2.2. Preliminary segmentation

Each intersaccadic interval is divided into windows, w_i, of size t_w (ms), with an overlap of t_o (ms). For all pairs of x- and y-coordinates contained in the window, the sample-to-sample direction, $\alpha(n)$, is computed as the angle of the line between two consecutive pairs of x- and y-coordinates to the x-axis. In order to investigate whether the sample-to-sample directions in each window are consistent a Rayleigh test is performed [15]. The sample-to-sample direction, $\alpha(n)$, is transformed into Cartesian coordinates $r_i(n)$, for $n = 1, 2, \ldots, N-1$, where N is the number of samples in w_i.

$$r_i(n) = \begin{pmatrix} \sin(\alpha(n)) \\ \cos(\alpha(n)) \end{pmatrix} \tag{1}$$

The mean vector, $\bar{r}_i$, is calculated as

$$\bar{r}_i = \frac{1}{N}\sum_{n=1}^{N} r_i(n) \tag{2}$$

The Reyleigh test uses the resultant vector $R_i = \| \bar{r}_i \|$ to determine whether the sample-to-sample directions in the window are uniformly distributed or not. An approximation of the p-value under H_0 is computed using

$$P_i = \exp[\sqrt{1 + 4N + 4(N^2 - (R_i \cdot N)^2)} - (1 + 2N)] \tag{3}$$

The null and alternative hypotheses of the test, H_0 and H_A, respectively, are:

- H_0: The samples in the window are distributed uniformly around the unit circle.
- H_A: The samples in the window are not distributed uniformly around the unit circle.

The p-value of the test, P_i, is computed for each window i. Since there is an overlap between the windows, each sample may belong to more than one window. The mean value of P_j, for all windows j which sample k belongs to is computed as,

$$P_{\text{mean}}(k) = \frac{1}{K}\sum_{j=1}^{K} P_j \tag{4}$$

where K is the number of windows each sample belongs to, $k = 1, 2, \ldots, M$, and M is the number of samples in the intersaccadic interval. All consecutive samples in the interval satisfying either $P_{\text{mean}}(k) \geq \eta_P$ or $P_{\text{mean}}(k) < \eta_P$ are grouped together into preliminary segments sharing similar properties in terms of directionality. These preliminary segments are further analyzed in the next step.

2.3. Evaluation of spatial features in the position signal

For all preliminary segments that have a duration longer than t_{min}, four parameters, p_D, p_{CD}, p_{PD}, and p_R, are calculated. These four parameters describe the dispersion (D), the consistency in the direction (CD), the positional displacement (PD), and the range (R) of the segment, which all are parameters that are typical for a smooth pursuit movement. In order to measure the dispersion, Principle Component Analysis (PCA) is employed. The first principle component determines the direction in which the data have their largest variance and the second principle component is chosen orthogonal to the first one. The principle components, pc_1 and pc_2, are computed by removing the respective mean from the preliminary x- and y-segments and estimating the covariance matrix, $\hat{C}$, between these. The zero mean data are projected onto the principle components, d_{pc_1} and d_{pc_2} respectively, and the lengths of the

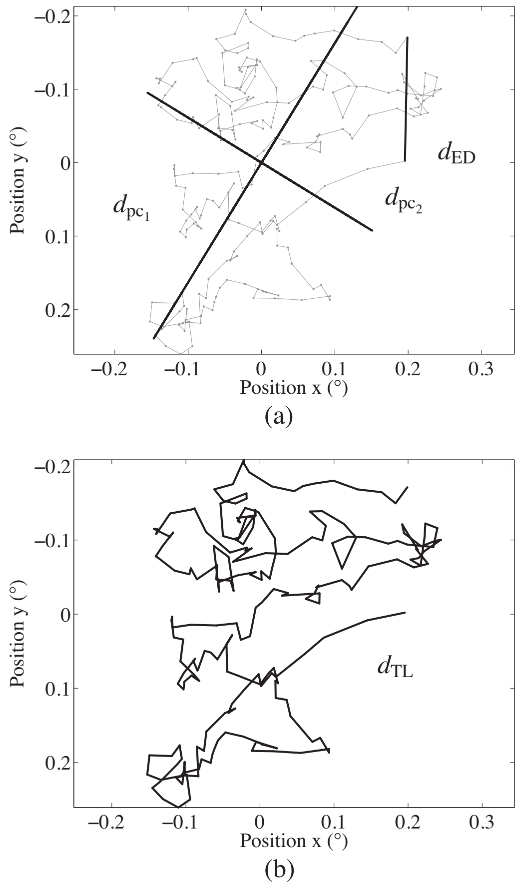

Fig. 2. (a) Illustration of d_{pc_1}, d_{pc_2}, and d_{ED}, in parameters p_D and p_{CD}. (b) Illustration of d_{TL} which is measured in parameter p_{PD}.

corresponding vectors are calculated, [11]. An illustration of d_{pc_1} and d_{pc_2}, is shown in Fig. 2a.

The first parameter, p_D, determines the relationship between the lengths of the first and the second principle components, d_{pc_1} and d_{pc_2}.

$$p_D = \frac{d_{pc_2}}{d_{pc_1}} \tag{5}$$

The parameter, p_D, measures if a preliminary segment is more dispersed in one direction than in the other, i.e., a value of p_D close to one means that the segment is equally spread in both directions.

The second parameter, p_{CD}, measures if the segment has a consistent direction or not. It is determined by computing the Euclidean distance (ED) between the starting and ending positions of the interval, d_{ED}, and comparing it to d_{pc_1}. An example of d_{ED} is shown in Fig. 2a.

$$p_{CD} = \frac{d_{ED}}{d_{pc_1}} \tag{6}$$

Hence, a value of p_{CD} close to one corresponds to that the data in the preliminary segment are starting and ending in the largest direction of the data. The third parameter, p_{PD}, measures the relationship between d_{ED} and the trajectory length (TL) of the segment, d_{TL}.

$$p_{PD} = \frac{d_{ED}}{d_{TL}} \tag{7}$$

A straight line will have p_{PD} equal to one, see Fig. 2b for an illustration of d_{TL}.

The fourth parameter, p_R, measures the absolute spatial range of the segment, and is computed as

$$p_R = \sqrt{(\max x - \min x)^2 + (\max y - \min y)^2} \tag{8}$$

where x and y are the x- and y-coordinates in the segment. The four parameters are calculated for each preliminary segment, and are compared to individual thresholds resulting in one criterion for each parameter.

1. Dispersion: $p_D < \eta_D$
2. Consistent direction: $p_{CD} > \eta_{CD}$
3. Positional displacement: $p_{PD} > \eta_{PD}$
4. Spatial range: $p_R > \eta_{maxFix}$

2.4. Classification of fixations and smooth pursuit movements

The segments are divided into three categories, depending on how many criteria that are satisfied. All segments where none of the criteria are satisfied are classified as fixations. Likewise, all segments with all criteria satisfied are classified as smooth pursuit movements. Finally, all segments where 1–3 criteria are satisfied are placed in a third category containing uncertain segments. The segments in this category have properties that may characterize both fixations and smooth pursuit movements. Consecutive segments belonging to the same category are grouped together.

The categorization of other segments in the same intersaccadic interval may provide information of whether the uncertain segment is part of a larger fixational interval or a larger smooth pursuit interval. The following strategy is used: First, each uncertain segment is evaluated through criterion 3, which is the criterion that evaluates the most typical feature of a smooth pursuit movement compared to a fixation. If criterion 3 is satisfied, the uncertain segment is most similar to a smooth pursuit movement and the spatial range, p_R, is recalculated by adding the spatial ranges of other smooth pursuit segments in the intersaccadic interval that has a mean direction that does not differ more than ϕ to the mean direction of the uncertain segment. If the merged segment has a $p_R > \eta_{minSmp}$, the segment is classified as a smooth pursuit and otherwise as a fixation. If, on the other hand, the segment is most similar to a fixation, i.e., criterion 3 is not satisfied, criterion 4 decides whether the segment is classified as a fixation; if criterion 4 is not satisfied, the segment is classified as a fixation and vice versa.

2.5. Performance evaluation

The performance of the proposed algorithm is evaluated using the following five methods:

1. *Event properties.* The total number of fixations and smooth pursuit movements are calculated as well as the mean duration for each of the two types of events.
2. *Proportion of events for different types of stimuli.* The percentage of each type of event is calculated for image-, and moving dot stimuli. The expected result for image stimuli is to have close to 100% detected fixations and close to 0% detected smooth pursuit movements. For moving dot stimuli the expected result is to have an as large amount of detected smooth pursuit movements as possible.
3. *Sensitivity and specificity analysis.* The sensitivity describes the ability of the algorithm to detect a certain type of event. The specificity is a complementary measure that describes the ability of the algorithm to correctly detect each type of event (c.f. [16]). When calculating the sensitivity and specificity, manual annotations are used as the “gold standard”. The annotations of each intersaccadic interval are compared to the detections of

the algorithm. Since the on- and offsets of the saccades may differ between the algorithm and the annotations, the data are classified into four groups: fixations (Fix), smooth pursuit movements (Smp), disturbances (Dist), and others, where disturbances includes all samples that are detected as blinks or removed outliers and others contains samples from adjacent saccades and PSO.

4. *Cohen's kappa analysis.* In order to evaluate the overall agreement between the manual annotations and the detections of the algorithm, Cohen's kappa is used. A detailed description of the calculations of sensitivity, specificity, and Cohen's kappa can be found in [13].
5. *Scores evaluation.* In [17] and later in [10], scores were proposed as an evaluation method for saccades, fixations, and smooth pursuit movements. Since the work in this paper focuses on the separation between fixations and smooth pursuit movements, only a set of the proposed scores in [10] are computed. The following scores are used:
   - *PQnS.* The ratio between the sum of the durations of all the detected smooth pursuit movements and the sum of the durations of moving dots in the stimuli. PQnS is compared to its corresponding ideal value, $\mathrm{PQnS}_{\mathrm{ideal}}$, which is calculated as the total duration of moving dots in the stimuli where the duration of the first fixation and the duration of the first corrective saccade are removed.
   - *$PQlS_P$.* Determines the mean distance between the moving dot stimuli and the samples detected as smooth pursuit movements. $\mathrm{PQlS}_{\mathrm{P}}$ is compared to its ideal value which is 0°.
   - *$PQlS_V$.* Determines the mean difference between the velocities of the detected smooth pursuit and of the corresponding stimuli. $\mathrm{PQlS}_{\mathrm{V}}$ is compared to its ideal value which is 0°/s.

   A detailed description and background to all scores can be found in [17,10].

3. Experiment and database

The eye-tracking signals used in this paper were collected during an experiment described in [13], where a Hi-Speed 1250 eye-tracker from SensoMotoric Instruments (Teltow, Germany) was used. The eye-tracking signals were recorded binocularly, with a sampling frequency of 500 Hz. In this paper, the signals from the right eye were used. The experiment was designed specifically for the evaluation of event detection algorithms when smooth pursuit movements are present. The experiment includes static images and short video clips as well as dots moving in different directions and speeds. The database was split into two parts: one development database and one test database. A subset of each database was manually annotated by two experts. In total for all stimuli, 33 trials were annotated by Expert 1 and 58 trials by Expert 2.

4. Results

All results presented in this section were generated using the settings shown in Table 1, which were chosen to maximize both the sensitivity and specificity of the algorithm with respect to the manually annotated development database. The detected fixations and smooth pursuit movements are compared to those detected by the I-VDT algorithm proposed in [10] and the I-PCA algorithm proposed in [11]. The I-VDT algorithm is used with the parameter settings proposed in [10], i.e., using a velocity threshold T_V =75°/s, a temporal window T_W = 150 ms, and a dispersion threshold T_D = 1.9°. The I-PCA algorithm, which is part of the iLab C++ Neuromorphic Vision Toolkit, was downloaded from http://iLab.usc.edu/toolkit and used with default settings. The preprocessing, where disturbances and blinks are removed, is the same for the three algorithms, see [13] for a description.

Table 1
Parameter settings for the proposed algorithm.

Parameter	Value	Description
t_w	22 ms	Window size
t_o	6 ms	Overlap of the windows
η_P	0.01	Significance level for the Rayleigh test
η_D	0.45	Threshold for p_D
η_{CD}	0.5	Threshold for p_{CD}
η_{PD}	0.2	Threshold for p_{PD}
η_{maxFix}	1.9°	Threshold for max spatial range for a fixation
η_{minSmp}	1.7°	Threshold for min spatial range for a smooth pursuit movement
ϕ	$\pi/4$	Max difference in mean direction for a smooth pursuit movement
t_{min}	40 ms	Minimum fixation duration

4.1. Event properties

The average properties of the detected fixations and smooth pursuit movements are shown in Table 2, for images, video, and moving dot stimuli, respectively. The results are summarized below:

- *Images*. For the development database the mean fixation durations are similar between the two experts and the three algorithms, with values ranging from 217 to 241 ms. The mean durations for the detected smooth pursuit movements are, however, less similar across the algorithms and the experts. In general, I-VDT detects the most and the shortest smooth pursuit movements with a mean duration of 48.7 ms. Expert 1 detects the fewest number of smooth pursuit movements with a mean duration of 361 ms. Except for the I-PCA, the algorithms detect a larger number of smooth pursuit movements than the experts. For the test database, the result has a similar pattern as for the development database. The largest difference is that Expert 1 does not detect any smooth pursuit movements.
- *Videos*. In the development database, the differences between experts and algorithms are larger than for images. The mean fixation duration is slightly larger for the proposed algorithm (218 ms) and considerably larger for I-VDT (360 ms) and I-PCA (298 ms), compared to the two experts with 206 ms and 179 ms, respectively. The I-VDT algorithm detects the largest number of smooth pursuit movements (66) and the I-PCA algorithm the fewest (22). For the test database, the agreement between the experts on the number of detected smooth pursuit movements is lower than for the development database.
- *Moving dots*. For the development database, the largest difference in the results is for the number of detected fixations, which ranges from 5 for Expert 1 to 37 for the I-VDT algorithm. The agreement between the proposed algorithm and the two experts is high for the number of detected smooth pursuit movements, 21 compared to 27 and 24, respectively. However, between the two experts, there is a large disagreement on the number of detected fixations and their durations, where Expert 1 detects fewer but longer fixations compared to Expert 2, with 5 and 17 fixations, respectively. For the test database, I-VDT and I-PCA have the shortest mean durations of smooth pursuit movements, 93.3 ms and 127 ms, respectively, compared to the proposed algorithm with a mean duration of 345 ms.

Table 2
Event properties for detected fixations and smooth pursuit movements, for image, video, and moving dot stimuli. A = proposed algorithm, B = I-VDT, C = I-PCA, D = Expert 1, and E = Expert 2.

Measure	Image					Video					Moving dot				
	A	B	C	D	E	A	B	C	D	E	A	B	C	D	E
Development database															
Mean fixation duration (ms)	217	241	224	217	214	218	360	298	206	179	191	266	297	256	157
Mean smooth pursuit duration (ms)	191	48.7	114	361	283	542	90.8	138	509	477	417	54.5	104	388	384
Number detected fixations	278	250	260	304	298	67	72	85	56	55	15	37	32	5	17
Number detected smooth pursuits	26	177	10	3	8	30	66	22	39	46	21	40	14	27	24
Test database															
Mean fixation duration (ms)	317	372	345	350	346	406	616	463	509	338	187	240	259	142	203
Mean smooth pursuit duration (ms)	350	38.4	80	0	310	759	147	173	583	484	345	93.3	127	328	344
Number detected fixations	96	87	92	99	93	25	29	37	26	26	8	23	20	4	2
Number detected smooth pursuits	9	82	3	0	9	13	19	12	12	24	16	18	8	20	20

4.2. Proportion of events for different types of stimuli

The percentages of fixations and smooth pursuit movements in the intersaccadic intervals are calculated for image and moving dot stimuli, see Table 3. The values for the proposed algorithm are based on the intersaccadic intervals resulting from the algorithm in [13], the values for I-VDT and I-PCA are resulting from the intersaccadic intervals from the saccade detection of each algorithm. The percentages are calculated for the complete development database and the complete test database. The results are summarized below:

- *Images*. The expected result for images is to have close to 100% detected fixations and 0% detected smooth pursuit movements. The proposed algorithm detects 91.2% fixations, I-VDT 86.9%, and I-PCA 98.1%. The corresponding numbers for detected smooth pursuit movements are 8.79%, 13.1%, and 1.94% for the development database. For the test database, the results are very similar to the results of the development database.
- *Moving dots*. The expected result is to have as large amount of detected smooth pursuit movements as possible and as few detected fixations as possible. The proposed algorithm detects 78.6% smooth pursuit movements and 21.4% fixations. The I-VDT algorithm detects 47.2% smooth pursuit movements and 52.8% fixations for the development database. The I-PCA algorithm detects the largest amount of fixations 86.6% and only 13.4% of smooth pursuit movements. For the test database, the results are very similar to the development database, with a slight increase in the percentages of smooth pursuit movements for all three algorithms.

4.3. Sensitivity and specificity analysis

Using both experts as references, the sensitivities and specificities of the detected fixations and smooth pursuit movements for the proposed algorithm, the I-VDT algorithm, and the I-PCA algorithm, respectively, are shown in Fig. 3. In the ideal case, both the sensitivity and specificity should be as close to one as possible. Below is a summary for the different types of events for the development database:

- *Fixations*. In general, the sensitivity for fixations is high, with values around 0.8–0.9, for the algorithms and with both experts as reference. For the specificity, there is a larger difference between the algorithms, where the proposed algorithm has values around 0.8–0.9 for all stimuli, while I-VDT and I-PCA have values ranging from 0.3 for video and moving dot stimuli, to 0.9 for image stimuli.
- *Smooth pursuit movements*. The sensitivity for the proposed algorithm is generally in the same range as for fixations, i.e., between 0.6 and 0.8 for all types of stimuli and compared to both experts. For the I-VDT algorithm, the sensitivity is between 0.2 and 0.6 for all types of stimuli. For the I-PCA algorithm, the sensitivity is between 0 and 0.2 for all types of stimuli, which is much lower than for fixations. The specificity for smooth pursuit movements is high for all algorithms, for all types of stimuli, and compared to both experts.
- *Disturbances and others*. The sensitivity and specificity for disturbances are high for all algorithms, independent of stimuli and expert, with values around 0.9. Since the proposed algorithm and I-VDT have the same preprocessing procedure the results for the two algorithms are almost identical. For I-PCA, which besides the preprocessing part from [13], has its own very strict disturbances detection, the result is slightly different from the other two algorithms. The specificities for the event type others are high for both algorithms, which means that the expert and the algorithms are in agreement about the transitions between saccades/PSO and other events.

4.4. Cohen's kappa analysis

In order to be able to measure the overall agreement between the experts and the algorithms, Cohen's kappa, κ, is calculated between each of the two experts and each of the three algorithms, see Tables 4 and 5. For the development database, Cohen's kappa for the proposed algorithm is larger than Cohen's kappa for I-VDT and I-PCA, for all types of stimuli. However, the agreement between the experts is even larger. For the test database, Cohen's kappa for the proposed algorithm is larger for video and moving dot stimuli, but lower than the other two algorithms for image stimuli. Also,

Table 3
Percentage of fixations and smooth pursuit movements in the intersaccadic intervals, for image and moving dot stimuli. A = proposed algorithm, B = I-VDT, C = I-PCA.

Measure	Development database						Test database					
	Image			Moving dot			Image			Moving dot		
	A	B	C	A	B	C	A	B	C	A	B	C
% Fixations	91.2	86.9	98.1	21.4	52.8	86.6	93.2	88.1	99.2	16.9	47.9	83.5
% Smooth pursuits	8.79	13.1	1.94	78.6	47.2	13.4	6.81	11.9	0.76	83.1	52.1	16.5

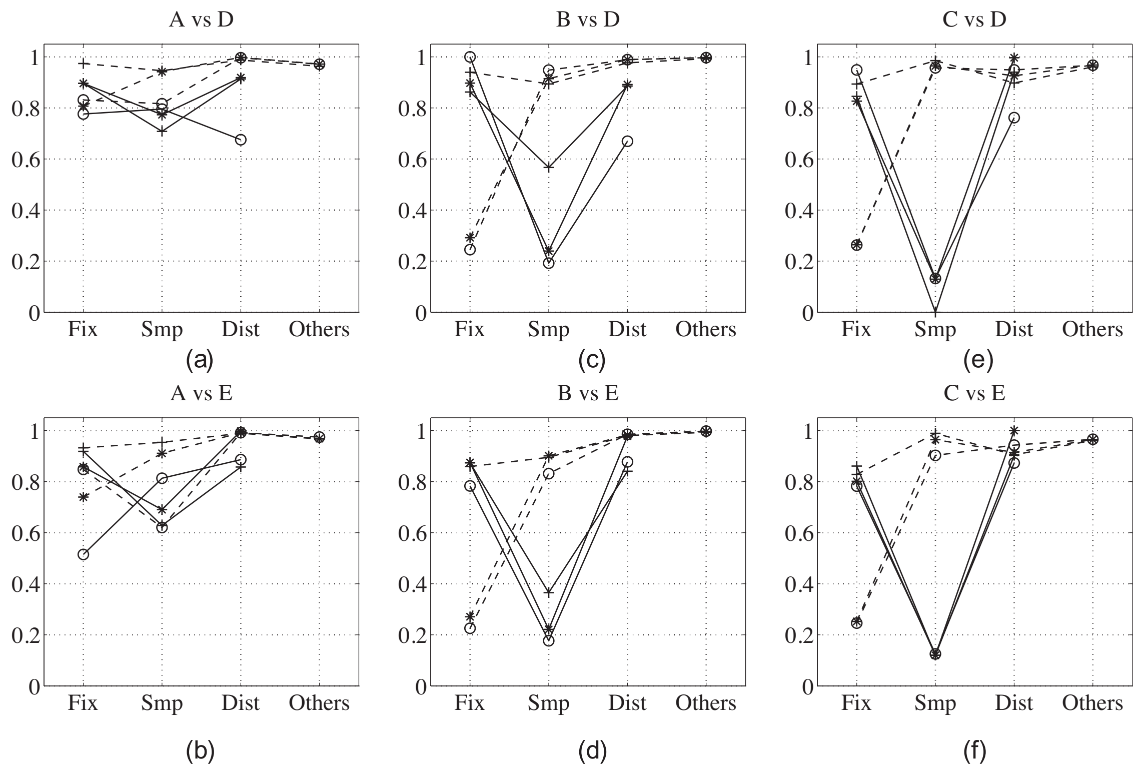

Fig. 3. Sensitivity (solid) and specificity (dashed), for images (+), video (*), and moving dot (o). (a) Proposed algorithm (A) with Expert 1 (D) as reference. (b) Proposed algorithm (A) with Expert 2 (E) as reference. (c) I-VDT algorithm (B) with Expert 1 (D) as reference. (d) I-VDT algorithm (B) with Expert 2 (E) as reference. (e) I-PCA algorithm (C) with Expert 1 (D) as reference. (f) I-PCA algorithm (C) with Expert 2 (E) as reference.

Table 4
Cohen's kappa between Expert 1 and the proposed algorithm, I-VDT, I-PCA, and Expert 2.

	Development database			Test database		
	Image	Video	Moving dot	Image	Video	Moving dot
Proposed algorithm	0.620	0.671	0.446	0.0685	0.383	0.423
I-VDT	0.524	0.180	0.098	0.091	0.378	0.0522
I-PCA	0.475	0.113	0.0827	0.12	0.24	0.0242
Expert 2	0.806	0.784	0.573	0.113	0.402	0.816

Table 5
Cohen's kappa between Expert 2 and the proposed algorithm, I-VDT, I-PCA, and Expert 1.

	Development database			Test database		
	Image	Video	Moving dot	Image	Video	Moving dot
Proposed algorithm	0.667	0.530	0.412	0.0595	0.401	0.309
I-VDT	0.537	0.127	0.050	0.105	0.172	0.0362
I-PCA	0.501	0.0744	0.0524	0.066	0.152	0.0257
Expert 1	0.834	0.779	0.550	0.116	0.395	0.687

Cohen's kappa between the experts is much lower for image stimuli than for other stimuli types.

4.5. Scores evaluation

The performance of the proposed algorithm is also evaluated by calculating scores for smooth pursuit movements, as proposed in [10]. Since scores can only be used when the coordinates of the stimuli are known, they are calculated for 17 trials containing moving dot stimuli. The scores were computed for the proposed algorithm, the I-VDT algorithm, the I-PCA algorithm, and Expert 2, and their values over the 17 trials are shown in Table 6, (these trials were not annotated by Expert 1).

- *PQnS*. All the algorithms and Expert 2 have a value of $PQnS_{ideal}$ close to 90%. The PQnS value of Expert 2 is closest to its corresponding ideal value, 76.8% compared to the ideal value 90.9%. Between the algorithms, the proposed algorithm is closer to its

Table 6
Values of the scores and Cohen's kappa between Expert 2 and the proposed algorithm, I-VDT, and I-PCA. A = proposed algorithm, B = I-VDT, C = I-PCA, and E = Expert 2.

Measure	A	B	C	E
$PQnS_{ideal}$ (%)	88.9	86.0	84.9	90.9
PQnS (%)	62.7	24.7	13.1	76.8
$PQlS_P$ (°)	2.83	2.98	2.1	2.9
$PQlS_V$ (°/s)	12.1	17.9	36.6	13.4
Cohen's kappa	0.31	0.07	0.07	1

corresponding ideal value with 62.7% and ideal value 88.9%, compared to I-VDT with 24.7% and ideal value 86%, and I-PCA with 13.1% and ideal value 84.9%.

- *PQIS*$_P$. The values for PQIS$_P$, which describes the mean distance between the smooth pursuit samples and the stimuli, are around $2.1-2.9^\circ$ for the proposed algorithm, I-VDT, I-PCA, and Expert 2. The corresponding ideal value is 0°.
- *PQIS*$_V$. The mean differences between the velocity of the stimuli and that of the eye are ranging from 12.1°/s for the proposed algorithm to 36.6°/s for I-PCA. The corresponding ideal value is 0°/s.

Cohen's kappa is also calculated separately between Expert 2 and the proposed algorithm, I-VDT, and I-PCA, respectively, for the 17 trials used in the calculation of the scores. The results are shown in Table 6; the proposed algorithm has a Cohen's kappa of 0.31 and I-VDT and I-PCA have 0.07. These results are in the same range as the values for Cohen's kappa for moving dot stimuli in Table 5.

5. Discussion

An algorithm for discriminating between fixations and smooth pursuit movements was developed. In order to perform the discrimination, the algorithm uses four features of the position signal. The algorithm was evaluated using signals recorded during both static and dynamic stimuli presentation, and was compared to the I-VDT algorithm [10] and the I-PCA algorithm [11], as well as to annotations performed by experts. In general, regardless of stimuli, the proposed algorithm detected longer but fewer smooth pursuit movements than the I-VDT algorithm. One reason for this behavior may be that the I-VDT algorithm used only one feature of the signal, the dispersion, in order to detect the smooth pursuit movement, and when the dispersion exceeded the threshold several times, the signal became more segmented. In comparison to the I-PCA, which uses several features to detect the smooth pursuit movements, the proposed algorithm detects longer and a larger amount of smooth pursuit movements.

The percentages of fixations and smooth pursuit movements were calculated for two types of stimuli – images and moving dots. In theory, it is expected to have close to 100% detected fixations and close to 0% detected smooth pursuit movements for images and the opposite for moving dots. For images, the results were 91.2% fixations and 8.8% smooth pursuit movements for the proposed algorithm. A part of the samples that was detected as smooth pursuit movements during image stimuli may be due to vergence. Since the proposed algorithm uses data from one eye only, it cannot distinguish such movements from smooth pursuit movements. It should be noted that also the experts detected 1–2% smooth pursuit movements in image stimuli, calculated for the manually annotated part of the development database.

The settings for the I-VDT algorithm were chosen as suggested in [10]. By using these settings on our database, the I-VDT algorithm is tuned to detect fixations well, which can be seen from the values of Cohen's kappa in Table 4, (0.52, 0.18, 0.10), for image, video, and moving dot stimuli, respectively. In order to make the algorithm less sensitive to fixations, Cohen's kappa was calculated also for $T_D = 1.1^\circ$, (0.33, 0.22, 0.28). By lowering the dispersion threshold, the I-VDT algorithm detects a larger number of smooth pursuit movements and shorter durations of fixations in the video and the moving dot stimuli, and gives a larger Cohen's kappa for these types of stimuli. Even though Cohen's kappa becomes more evenly distributed over the three types of stimuli, it is not in the ranges of the proposed algorithm that has a larger Cohen's kappa for all types of stimuli.

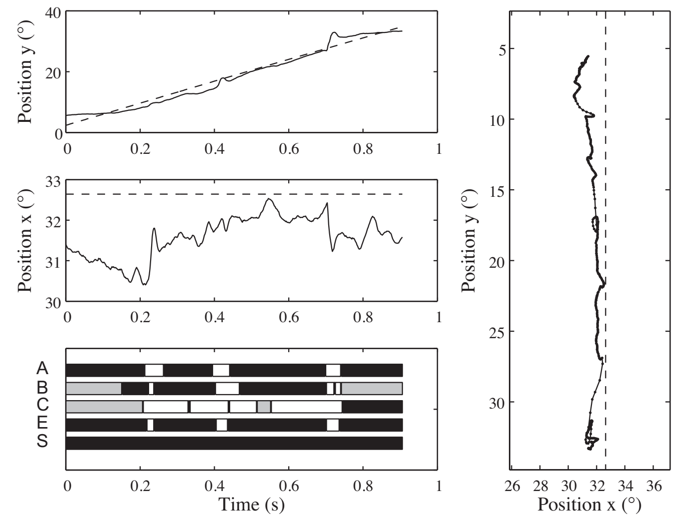

Fig. 4. Example of a trial with a moving dot, where the detections for the proposed algorithm are in agreement with the expert most of the time, but Cohen's kappa is 0. The lower panel shows the detection results for the proposed algorithm (A), I-VDT (B), I-PCA (C), Expert 2 (E), and what the stimuli was (S). Black color represents smooth pursuit movements, gray samples are fixations and white are all other types of eye movements.

For the I-PCA algorithm the default settings of the algorithm were chosen. The algorithm is clearly tuned and developed to be able to detect saccades and fixations of a predefined size, shape, and duration, which is shown by the low percentage-values in Table 3 for image stimuli. The results in Table 3 also show the difficult trade-off between accurately detecting few smooth pursuit movements in image stimuli and at the same time a large amount of smooth pursuit movements in moving dot stimuli.

When comparing Cohen's kappa of the algorithms to Cohen's kappa between the experts, the agreement between the experts is higher. However, Cohen's kappa between the experts still has a large variation between the different types of stimuli, with values from 0.55 for moving dot stimuli to 0.83 for image stimuli, which indicates that the separation of fixations and smooth pursuit movements is a difficult task even for experts. One explanation to why Cohen's kappa is a bit lower than expected between experts and between experts and algorithms, may be due to that the data are unbalanced; for image stimuli, the majority of samples are fixations with very few smooth pursuit movements while the opposite is true for moving dot stimuli. When Cohen's kappa is calculated for databases that have an unbalanced distribution between the types of events, small differences in the two compared detections lead to a substantial decrease in Cohen's kappa, even though the detections are correct most of the time. An example is given in Fig. 4 where Cohen's kappa is 0 between the expert and the proposed algorithm since the expert does not classify any sample as a fixation. An unbalanced database in combination with a low number of trials are the reasons for the much lower values of Cohen's kappa in Tables 4 and 5 for images stimuli in the test database, both for algorithms and experts.

An important question is whether the annotations represent a "gold standard". The information that the algorithm and the expert are using in order to make the decision may differ a lot and may potentially render the comparison unfair. The experts can often guess which types of stimuli that have been used. This may partly explain why the two experts have a larger agreement between themselves than between experts and algorithms. The fact that the two experts sometimes differ makes it even harder to decide which one to trust or use as the "gold standard". In [18], three manual coders were used and the correlation between the coders ranged between 0.58 and 0.85. This is comparable to the Cohen's kappa between experts reported in this paper.

The performance of the proposed algorithm was evaluated using five different methods, each with advantages and drawbacks. In order to provide an overview of the detected events, their properties, and the proportion of events for different types of stimuli, method 1 (event properties), and 2 (proportion of events for different types of stimuli), are satisfactory methods. However, these methods do not reveal whether the events were correctly detected or not. By using method 3 (sensitivity and specificity analysis), and method 4 (Cohen's kappa analysis), the accuracy of the classification is taken into account. The drawback with these methods is that there is a need for a "gold standard", to which the results of the algorithm can be compared. In this paper, manual annotations from two experts were used. When using method 5, (Scores), there is no need for time consuming annotations since the stimuli are used as references. However, this strategy cannot be used for all types of stimuli, e.g., not for images, text stimuli or video stimuli. In addition, not all types of events can be evaluated, e.g., PSO, since they are not driven by the stimulus. To summarize, when comparing and evaluating algorithms, the prerequisite in terms of stimuli and types of events to be detected will control which type of evaluation method that should be used. All methods are complementary and no single method will show the complete performance of the evaluated algorithm.

So far, discrimination between fixations and smooth pursuit movements has mainly been used in human–computer interaction using low speed eye-trackers, e.g., to stabilize the cursor during gaze control of a computer screen [19], and in interaction with information screens [20]. Having the possibility to separate between the two event types also for high-speed eye-trackers is paving the way for studies where the properties of the two types of events can be investigated and compared. Two examples of such applications are to measure the difference in smooth pursuit characteristics between experts and novices when watching dynamic stimuli [21], and the amount of smooth pursuit when viewing natural stimuli as a diagnostic tool for neural disorders [22].

6. Conclusions

Discrimination between fixations and smooth pursuit movements is a difficult task since many of the signal characteristics of the two event types are similar. In this work, an algorithm for the discrimination between fixations and smooth pursuit movements in high-speed eye-tracking data is developed and compared with two existing algorithms and to annotations from two experts. A rigorous performance evaluation strategy was employed to capture different aspects of the algorithm's behavior. The proposed algorithm outperforms two current state-of-the-art algorithms for detection of fixations and smooth pursuit movements, regardless of the stimuli and evaluation method. However, the agreement to annotations is not as high as the inter-rater agreement between the experts.

Acknowledgments

This work was supported by the Swedish strategic research programme eSSENCE.

References

[1] B. Tatler, The central fixation bias in scene viewing: selecting an optimal viewing position independently of motor biases and image feature distributions, J. Vis. 7 (14) (2007) 1–17.
[2] J. Henderson, A. Hollingworth, High-level scene perception, Annu. Rev. Psychol. 50 (1999) 243–271.
[3] K. Gidlöf, A. Wallin, R. Dewhurst, K. Holmqvist, Gaze behavior during decision making in a natural environment, J. Eye Mov. Res. 6 (1) (2013) 1–14.
[4] K.-M. Flechtner, B. Steinacher, R. Sauer, A. Mackert, Smooth pursuit eye movements in schizophrenia and affective disorder, Psychol. Med. 27 (1997) 1411–1419.
[5] M. Dorr, T. Martinez, K. Gegenfurtner, E. Barth, Variability of eye movements when viewing dynamic natural scenes, J. Vis. 10 (28) (2010) 1–17.
[6] E. Kowler, Eye movements: the past 25 years, Vis. Res. 51 (2011) 1457–1483.
[7] P. Mital, T. Smith, R. Hill, J. Henderson, Clustering of gaze during dynamic scene viewing is predicted by motion, Cogn. Comput. 3 (1) (2010) 5–24.
[8] K. Holmqvist, M. Nyström, R. Andersson, R. Dewhurst, H. Jarodzka, J. van de Weijer, Eye Tracking: A Comprehensive Guide to Methods and Measures, Oxford University Press, 2011.
[9] M. Vidal, A. Bulling, H. Gellersen, Analysing EOG signal features for the discrimination of eye movements with wearable devices, in: Proceedings of the First International Workshop on Pervasive Eye Tracking & Mobile Eye-Based Interaction, ACM, 2011, pp. 15–20.
[10] O. Komogortsev, A. Karpov, Automated classification and scoring of smooth pursuit eye movements in the presence of fixations and saccades, Behav. Res. Methods 45 (1) (2013) 203–215.
[11] D.J. Berg, S.E. Boehnke, R.A. Marino, D.P. Munoz, L. Itti, Free viewing of dynamic stimuli by humans and monkeys, J. Vis. 9 (5) (2009) 1–15.
[12] M. Vidal, A. Bulling, H. Gellersen, Detection of smooth pursuits eye movement using shape features, in: ETRA '12 Proceedings of the Symposium on Eye Tracking Research and Applications, ACM, New York, 2012, pp. 177–180.
[13] L. Larsson, M. Nyström, M. Stridh, Detection of saccades and post-saccadic oscillations in the presence of smooth pursuit, IEEE Trans. Biomed. Eng. 60 (9) (2013) 2484–2493.
[14] C. Meyer, A. Lasker, D. Robinson, The upper limit of human smooth pursuit, Vis. Res. 25 (4) (1985) 561–563.
[15] P. Berens, Circstat: a matlab toolbox for circular statistics, J. Stat. Softw. 31 (10) (2009) 1–21.
[16] L. Sörnmo, P. Laguna, Bioelectrical Signal Processing in Cardiac and Neurological Applications, Elsevier, 2005.
[17] O. Komogortsev, D. Gobert, S. Jayarathna, D. Hyong Koh, S. Gowda, Standardization of automated analyses of oculomotor fixation and saccadic behaviors, IEEE Trans. Biomed. Eng. 57 (11) (2010) 2635–2645.
[18] S. Munn, L. Stefano, J.B. Pelz, Fixation-identification in dynamic scenes: comparing an automated algorithm to manual coding, in: ETRA '08 Proceedings of the Symposium on Eye Tracking Research and Applications, ACM, 2008, pp. 33–42.
[19] J.S. Agustin, Off-the-shelf gaze interaction, The IT University of Copenhagen, 2009, Ph.D. thesis.
[20] M. Vidal, A. Bulling, H. Gellersen, Pursuits: spontaneous interaction with displays based on smooth pursuit eye movement and moving targets, in: Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing, ACM, 2013, pp. 439–448.
[21] H. Jarodzka, K. Scheiter, P. Gerjets, T. van Gog, In the eyes of the beholder: how experts and novices interpret dynamic stimuli, Learn. Instr. 20 (2) (2010) 146–154.
[22] M. Vidal, J. Turner, A. Bulling, H. Gellersen, Wearable eye tracking for mental health monitoring, Comput. Commun. 35 (11) (2012) 1306–1311.