# FIXATION IDENTIFICATION FOR LOW-SAMPLE-RATE MOBILE EYE TRACKERS

*N. Anantrasirichai, Iain D. Gilchrist and David R. Bull*

Bristol Vision Institute, University of Bristol, Bristol BS8 1UB, UK

## ABSTRACT

This paper presents a novel method of fixation identification for mobile eye trackers. The most significant benefit of our method over the state-of-the-art is that it achieves high accuracy for low-sample-rate devices worn during locomotion. This in turn delivers higher quality datasets for further use in human behaviour research, robotics and the development of guidance aids for the visually impaired. The proposed method employs temporal characteristics of the eye positions combined with statistical visual features extracted using a deep convolutional neural network, inspired by models of the primate visual system, through the fovea and peripheral areas around the eye positions. The results show that the proposed method outperforms existing methods by up to 16 % in terms of classification accuracy.

***Index Terms***— eye data, fixation, saccade, convolutional neural network, classification

## 1. INTRODUCTION

Eye trackers are frequently employed in human behaviour research to capture instantaneous eye positions, revealing what visual information is important for different visual contexts. The major events of eye motion patterns consist of fixations of different temporal durations, saccades of different spatial length, direction and temporal length, and blinks. Fixations are generally considered to be the most interesting of the eye movement characteristics, since they indicate what and when visual information is most probably registered to the brain [1].

Most research on eye movements has been based on the presentation of static images or video clips with participants being asked to look at a sequence of images on a computer screen. The eye trackers developed for this purpose currently support ultra high sampling rates (up to 2000 Hz [2]), leading to high accuracy characterisation of eye movements. In contrast, the sampling-frequency capability of mobile eye trackers is limited by additional constraints, such as light weight size and low power consumption. With current technology, mobile eye trackers can provide sampling rates of up to 60 Hz [2, 3], although those with higher rate are expensive. Cheaper or older models provide a sampling rate of up to 30 Hz, which is lower than the requirement for fixation identification, since saccadic movements can be shorter than 50 ms (20 Hz) [4, 5], of which Nyquist frequency for this saccade tracking is 40 Hz. Also, the traditional fixation-based algorithms cannot be applied because the head movement and tracking behaviour (smooth downward movements to maintain fixation on an environmental feature during locomotion) cause dispersion to be too high.

A large number of fixation identification methods have been proposed (as described in Section 2). However, these methods can produce widely varying results, especially for data from mobile eye trackers [6, 7]. Fixation patterns occurring during locomotion are different to those when viewing a screen. Humans use oculomotor fixations when they are walking as the surrounding scene appears to be moving relative to them. These fixations encompass the ability to suppress ocular drifts whilst maintaining a steady retinal image of a single target of interest.

The main contribution of this paper is a method to achieve fixation identification, or noise removal, for low-sample-rate mobile eye trackers. We employ a learning method using a deep convolutional neural network (CNN) as it resembles the organization of information in the human visual system. Saccades, blinks and noise due to light interference and large head movements are classified as one class against another for fixations. First, the eye positions and temporal relationships between them are employed to select the fixation points to be used in the training process, as the number of fixations is significantly larger than that of saccades and noise combined. Then the areas around the selected points are employed to extract key features used in the final classifier.

The remainder of this paper is organised as follows. Section 2 presents existing work on fixation identification. Subsequently, we describe our proposed method in Section 3 and the results are shown in Section 4. Finally the conclusions and future work are set out in Section 5.

## 2. FIXATION IDENTIFICATION FOR EYE DATA

The most common algorithm for fixation identification is velocity-based thresholding (I-VT) [8]. This classifies eye movements based on the velocity (visual degrees per second °/s) of the directional shifts of the eye. If the velocity is lower than the threshold, the point is classified as a fixation. Subsequently, the classification result is further compared to that of the previous samples within the same class to create a

fixation or saccade. Komogortsev et. al. suggests a velocity threshold of 30°/s in [9]. Tobii Technology applies a window length of 20 ms to average velocity when signals contain high noise [8]. An adaptive threshold using mean and standard deviation was proposed in [6].

An alternative approach, using dispersion-based thresholds (I-DT) typically identifies gaze samples as belonging to a fixation if the samples are located within a spatially localised region (about 0.5°) for a minimum period of time: the minimum allowed fixation duration (80-150 msec) [10]. This simply follows the assumption that fixation points generally occur near one another. Saccades are then detected implicitly as everything else. Probability-based algorithms, such as Hidden Markov model [8], can be used to determine the most likely identifications for a given protocol. A two-step spatial dispersion threshold is proposed in [11].

The methods above exhibit poor performance in the detection of fixations and saccades in dynamic scenes. To address this, some fixation identification algorithms have been proposed for mobile eye trackers. These include a Bayesian mixture model for traffic hazard perception [12] and the combination of I-VT and duration sensitivity to track fixations [13].

## 3. PROPOSED SCHEME

The proposed framework is depicted in Fig. 1, where two classifiers are employed for fixation-point selection, described in Section 3.1, and fixation identification, described in Section 3.2. We identify each event as one of two classes: i) fixation or ii) saccade and noise. In the training process, the data from the eye trackers during eye blinks, occurring two to four times per minute under normal conditions, are removed.

### 3.1. Fixation-point selection

Generally, training a classifier requires a balanced number of samples from each class; otherwise, the predicted results can be biased towards the class with more training samples. Eye tracking data always contains a significantly larger proportion of fixation samples, due to the nature of the human visual system. Therefore, if the number of training saccades and noise is $N_t$, we also use $N_t$ fixations. Here, we select these $N_t$ fixations using a support vector machine [14], which creates a hyperplane between the randomly selected fixation points and the saccade points (plus noise). Then, the distances to the hyperplane are measured.

We employ eight features, $F^0$, produced from eye positions and their temporal relationships as follows.

- Eye position at $X_t = \{x_t, y_t\}$: two features indicate that eye positions obtained from the mobile eye tracker exhibit centre-bias behaviour – since the head is often moved to improve vision. The eye positions near the edge of the image are often noise occurring when the light outside interferes with the camera used for detecting pupils.

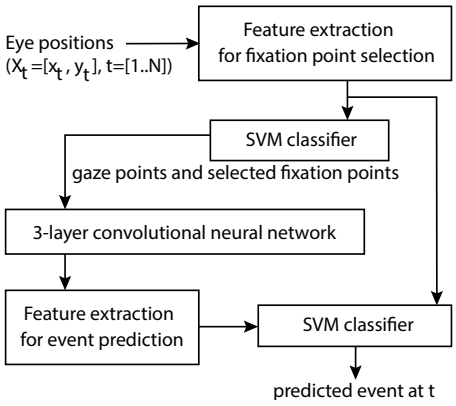


**Fig. 1**. Diagram of the proposed method for event prediction.

- Minimum distances to $D(X_{t-1})$ or $D(X_{t+1})$ (in both horizontal and vertical directions (two features)), where $D(m_n)$ is a warp function from position $m$ on frame $n$ to the current frame's geometry: if the eye position in the current frame is near to that of the adjacent frame, it is likely to be a fixation.
- 2-point angular velocities between $X_t$ and $D(X_{t-1})$, $v_{(t-1,t)} = \frac{|X_t - D(X_{t-1})|}{\Delta t}$, where $\Delta t = \frac{1}{30}$ sec for a 30-Hz eye tracker: if the speed relative to the previous eye position mapped to the current frame's geometry is small, the current point is likely to be a fixation.
- 2-point angular velocities between $X_t$ and $D(X_{t+1})$, $v_{(t,t+1)} = \frac{|X_t - D(X_{t+1})|}{\Delta t}$: if the speed relative to the next frame's eye position mapped to the current frame's geometry is small, the current point is likely to be a fixation.
- Angular acceleration, $a_t = \frac{v_{(t,t+1)} - v_{(t-1,t)}}{\Delta t}$: high accelerations indicate small saccades.
- Interframe angle change, $\theta_t = \arccos\left(\frac{v'_{(t,t+1)} \cdot v'_{(t-1,t)}}{\left|v'_{(t,t+1)}\right|\left|v'_{(t-1,t)}\right|}\right)$, where $v'_{(m,n)} = \frac{X_m - X_n}{m-n}$: if the angle between the vectors directed from the current eye position to the eye positions in the adjacent frames is close to 180°, the current eye position is likely to be a saccadic movement.

In this paper, ten randomly selected fixation groups were used for training and the average distance from each sample to these ten hyperplanes was computed. The distance of sample $x$ from the hyperplane is computed as shown in Eq. 1, where $w$ and $b$ are the hyperplane normal vector and the bias, respectively.

$$d_x = \frac{w^T F_x^0 + b}{|w|} \tag{1}$$

$N_t$ fixation points are selected by combining the $N_{inc}$ fixation points incorrectly classified as saccades with the $N_t - N_{inc}$ points that have the smallest distance to the hyperplane

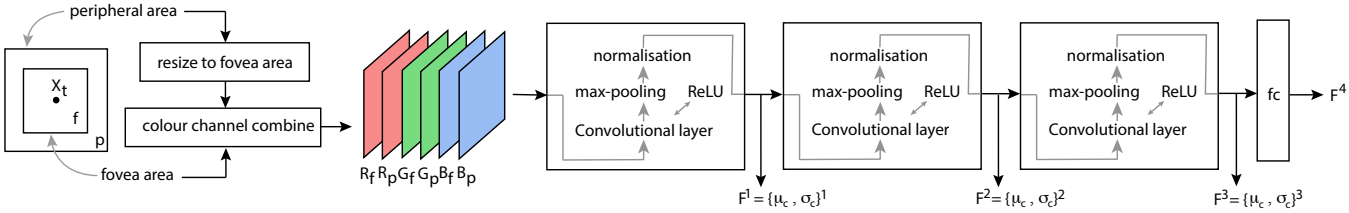


**Fig. 2**. Feature extraction from 3-layer CNN.

(since generally $N_{inc} < N_t$). These points are the weakest classifiers at this stage, so are ideal training data for the CNN. The correctly classified points with large distances to the hyperplane are easier to classify.

### 3.2. Local visual features by deep learning

The final event classifier employs the features $F^0$ and the statistical visual features extracted using the process shown in Fig. 2. The fovea (3°) and peripheral (6°) areas around the selected eye positions are used. The peripheral area is resized to the same size as the fovea area, $h_f \times h_f$ pixels. Then, these two areas, 6 colour channels in total, are combined into one three-dimensional input to train the CNN.

#### *3.2.1. Convolutional neural network*

A CNN is a biologically-inspired architecture that comprises multiple layers of neuron collections that have learnable weights and biases. Their results are tiled so that they overlap to obtain a better representation of the original image. The CNN creates its filters' values based on the task. Generally the CNN learns to detect edges from the raw pixels in the first layer, then uses the edges to detect simple shapes in the next layer. The higher layers produce higher-level features. Here, we develop our network using the Caffe [15] framework. The network consists of three layers of convolution, max-pooling, rectified linear unit (ReLU), and local normalisation, followed by a fully connected layer and a linear classifier at the top as shown in Fig. 2. Deeper networks may be used which generally give better performance. The three-layer network is employed here because of the limitations of our computational system.

#### *3.2.2. Feature extraction*

We compute the mean $\mu$ and the variance $\sigma^2$ of each channel of the output of each layer. Following the parameters recommended in Krizhevsky's ConvNet structure [15], the input with size of $(6 \times h_f \times h_f)$ produces the outputs of layers 1 to 3 with the sizes of $(32 \times \lfloor \frac{h_f}{2} \rfloor \times \lfloor \frac{h_f}{2} \rfloor)$, $(32 \times \lfloor \frac{h_f}{4} \rfloor \times \lfloor \frac{h_f}{4} \rfloor)$ and $(64 \times \lfloor \frac{h_f}{8} \rfloor \times \lfloor \frac{h_f}{8} \rfloor)$, respectively. That means, $F^1 = \{\mu_1^1, \sigma_1^1, \ldots, \mu_{32}^1, \sigma_{32}^1\}$, $F^2 = \{\mu_1^2, \sigma_1^2, \ldots, \mu_{32}^2, \sigma_{32}^2\}$ and $F^3 = \{\mu_1^3, \sigma_1^3, \ldots, \mu_{64}^3, \sigma_{64}^3\}$. The output vectors of the CNN are also employed, i.e. $F^4$ with the length of 10. Including $F^0$ described in Section 3.1, the set of features used for fixation identification is shown in Eq. 2 – a total of 274 dimensions, which are subsequently employed in the final SVM classifier.

$$F = \{F^0, F^1, F^2, F^3, F^4\} \tag{2}$$

## 4. RESULTS AND DISCUSSION

We first tested features $F^0$ using a high-sample-rate eye tracker to examine whether these features are suitable to be used for fixation-point selection. Subsequently, our framework was tested with a low-sample-rate mobile eye tracker, using sequences collected from participants during locomotion on different terrain types. We employ a support vector machine (LIBSVM) [14] to perform linear classification. The linear kernel is robust to overfitting and gives better speed than a non-linear kernel. We compare our method to i) velocity-based threshold (I-VT) [10], ii) hidden Markov model identification [16], and iii) the EyeMMV [11].

### 4.1. 1000-Hz EyeLink 1000

The features for fixation-point selection were first tested with the down-sampled data of the 1000-Hz EyeLink eye tracker [17]. The events marked by this device were used as ground truth. 30-Hz samples were created by subsampling the 1000Hz data using an average low-pass filter with the length of $\lfloor \frac{1000}{30} \rfloor$. This replicates how cameras function by opening and closing the shutter for the duration of exposure. Four sequences were captured while 4 subjects were watching a short film. One sequence was used for training, and the other three were used for testing. The average number of fixations and saccades for one sequence are approximately 10,000 and 1,100 respectively, so 1,100 fixation points were randomly selected to ensure that the number of both classes were balanced. There were 40 different random fixation sets creating 40 tests in total and the results were averaged.

The precision, recall and classification accuracy of fixation identification are shown in Table 1. High precision means that fixations are rarely misidentified as saccades, whilst high recall means all the true fixations are correctly identified. High accuracy implies that most fixations and saccades are predicted correctly. Table 1 shows that our simple feature set $F^0$ provides excellent performance – both fast (less than

**Fig. 3**. Eye tracking sequences containing a variety of terrain types. The cycles show eye positions which may be fixations or saccades or noises

**Table 1**. Classification performance (%) for down-sampled data (30 Hz) from the 1000-Hz Eye link eye tracker

| method | precision | recall | accuracy |
|---|---|---|---|
| I-VT | **94.47** | 95.68 | 92.07 |
| I-HMM | 94.30 | 94.67 | 90.97 |
| EyeMMV | 93.44 | 96.02 | 90.50 |
| $F^0$ | 94.40 | **98.42** | **94.07** |

**Table 2**. Classification performance (%) for 30-Hz SMI mobile eye tracker

| method | precision | recall | accuracy |
|---|---|---|---|
| I-VT | 68.84 | 80.52 | 69.41 |
| I-HMM | 70.33 | 83.62 | 70.33 |
| EyeMMV | 66.85 | 78.23 | 68.82 |
| $F^0$ | 73.02 | 84.06 | 73.16 |
| CNN ($F^4$) | 78.84 | 91.19 | 80.29 |
| $F^1$-$F^4$ | 84.92 | 90.81 | 85.08 |
| $F$ | **87.32** | **91.24** | **86.30** |
| $F$ w/o fix-sel | 85.83 | 90.45 | 84.41 |

0.25 $\mu$s/frame using Matlab on i7 CPU 2.8GHz) and highly accurate. Incorrect predictions were observed to occur at the beginning of the fixation, where our eyes are slightly shaky whilst fixating.

### 4.2. 30-Hz SMI mobile eye tracker

The test sequences were acquired with the SensoMotoric Instruments (SMI) Eye Tracking Glasses. These produce a point of view video at a resolution of 1280×960 pixels ($W \times H$) at 30 fps. The system provides a scene field of view of 60° horizontally and 46° vertically. Hence, the foveal and peripheral areas used in our method are approximately $64 \times 64$ pixels ($h_f = 64$) and $128 \times 128$ pixels, respectively.

We tested the proposed scheme using 12 sequences of mobile eye tracking data from 6 participants. Six sequences containing 4 distinct terrain types were used: flat concrete, slanted cobbles, stepping stones and rocks (shown in Fig. 3 columns 1–4). The other 6 sequences contain mixed materials including dirt, rocks, grass and wood on sloped terrain as shown in Fig. 3 columns 5 and 6. The ground truths were manually marked. The average fixations and saccades for one sequence are approximately 8,850 and 950, respectively. All sequences are between 4-6 minutes in duration.

A 2-fold cross validation was employed - 6 sequences were used for training and the other 6 were used for evaluation. We ran 10 tests with randomly selected training sequences and the results were averaged. Table 2 shows the performances of fixation identification. Our proposed method ($F$) gives the best results with improvements in precision, recall and classification accuracy of approximately 18 %, 10 % and 16 %, respectively. The performances of the I-VT, I-HMM and EyeMMV are dramatically lower than those of the static eye tracker (Section 4.1). This clearly shows that these methods are not suitable for mobile eye trackers. Prediction on the mobile eye tracker is significantly more difficult than on the static eye tracker because of outdoor conditions and '*track and return*' behaviour [18], where our eyes fixate at a particular location tracking it back as walking forwards, then saccading ahead again to fixate at the next location. When the features $F^1$-$F^4$ were included, the precision, recall and classification accuracy were improved from using only $F^0$ by up to 14 %, 7 % and 13 %, respectively. This implies that the local visual features play a very important role in attracting human attention during locomotion.

The last row shows the results when the fixation-point selection process (i.e. sample point balancing in Section 3.1) was not applied, i.e. the fixated points fed to the CNN were randomly selected. This reveals that our fixation-point selection method improves the classification performance up to 2 %.

## 5. CONCLUSIONS AND FUTURE WORK

This paper presents a new supervised learning method for fixation identification of the eye data. The proposed method employs temporal characteristics of the eye positions and local visual features extracted by a deep convolutional neural network (CNN), and then classifies the eye events via a support vector machine (SVM). Our method offers higher classification accuracy, precision and recall of fixations than the existing methods. The local visual features clearly enhance the performance of the fixation identification method for low-sample-rate mobile eye trackers. The fixation points predicted by our method can then be used in human behaviour research and to improve the performance of bio-inspired machines. Temporal visual features and a deeper CNN will be employed in future work.

## 6. REFERENCES


[1] K Rayner, "Eye movements in reading and information processing: 20 years of research," *Psychological Bulletin*, vol. 124, no. 3, pp. 372–422, 1998.

[2] Eye Tracking, "Hardware: Eye tracking systems," Tech. Rep., http://www.eyetracking.com/Hardware/Eye-Tracker-List, 2016.

[3] SensoMotonic Instruments, "Smi eye tracking glasses," Tech. Rep., http://eyetracking-glasses.com, 2016.

[4] B. Fischer and E. Ramsperger, "Human express saccades: extremely short reaction times of goal directed eye movements," *Experimental Brain Research*, vol. 57, pp. 191–195, 1984.

[5] H. Collewijn, C. J. Erkelens, and R. M. Steinman, "Binocular co-ordination of human vertical saccadic eye movements," *Journal of Physiology*, vol. 404, pp. 183–197, 1988.

[6] Marcus Nyström and Kenneth Holmqvist, "An adaptive algorithm for fixation, saccade, and glissade detection in eyetracking data," *Behavior Research Methods*, vol. 42, no. 1, pp. 188–204, 2010.

[7] C. Berger, M. Winkels, A. Lischke, and J. Hoppner, "Gazealyze: a matlab toolbox for the analysis of eye movement data," *Behavior Research Methods*, vol. 44, no. 2, pp. 404419, 2012.

[8] Dario D. Salvucci and Joseph H. Goldberg, "Identifying fixations and saccades in eye-tracking protocols," in *Proceedings of the 2000 Symposium on Eye Tracking Research & Applications*, New York, NY, USA, 2000, ETRA '00, pp. 71–78, ACM.

[9] O.V. Komogortsev, D.V. Gobert, S. Jayarathna, Do Hyong Koh, and S.M. Gowda, "Standardization of automated analyses of oculomotor fixation and saccadic behaviors," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 11, pp. 2635–2645, Nov 2010.

[10] Anneli Olsen, "The Tobii I-VT fixation filter: Algorithm description," Tech. Rep., Tobii, 2012.

[11] V. Krassanakis, V. Filippakopoulou, and B. Nakos, "Eyemmv toolbox: An eye movement post-analysis tool based on a two-step spatial dispersion threshold for fixation identification," *Journal of Eye Movement Research*, vol. 7, no. 1, pp. 1–10, 2014.

[12] Enkelejda Tafaj, Thomas C. Kubler, Gjergji Kasneci, Wolfgang Rosenstiel, and Martin Bogdan, "Online classification of eye tracking data for automated analysis of traffic hazard perception," *Artificial Neural Networks and Machine Learning*, vol. 8131, pp. 442–450, 2013.

[13] Susan M. Munn, Leanne Stefano, and Jeff B. Pelz, "Fixation-identification in dynamic scenes: Comparing an automated algorithm to manual coding," in *Proceedings of the 5th Symposium on Applied Perception in Graphics and Visualization*, 2008, pp. 33–42.

[14] C.C. Chang and C.J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011.

[15] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[16] O. V. Komogortsev and A. Karpov, "Automated classification and scoring of smooth pursuit eye movements in presence of fixations and saccades," *Journal of Behavioral Research Methods*, pp. 1–13, 2012.

[17] SR Research, "The eyelink," Tech. Rep., http://www.sr-research.com/eyelink1000.html, 2016.

[18] B.M. Hart and Einhäuser, "Mind the step: complementary effects of an implicit task on eye and head movements in real-life gaze allocation," *Experimental Brain Research*, vol. 223, pp. 233–249, 2012.