Eye Status Based on Eyelid Detection: A Driver Assistance System

Michał Daniluk[1], Mahdi Rezaei[2], Radu Nicolescu[2], and Reinhard Klette[2]

[1] Warsaw University of Technology, Poland
[2] The *.enpeda..* Project, The University of Auckland, New Zealand
`mdaniluk123@gmail.com`

Abstract. Fatigue and driver drowsiness monitoring is an important subject for designing driver assistance systems. The measurement of eye closure is a fundamental step for driver awareness detection. We propose a method which is based on eyelid detection and the measurement of the distance between the eyelids. First, the face and the eyes of the driver are localized. After extracting the eye region, the proposed algorithm detects eyelids and computes the percentage of eye closure. Experimental results are performed on the BioID database. Our comparisons show that the proposed method outperforms state-of-the-art methods.

1 Introduction and Related Work

Each year more than one million people is killed due to traffic accidents; for example, see the latest report [16] of the *World Health Organisation* (WHO) on road safety. In 2013 more than 1.2 million people died on the world's roads and another 50 million sustained nonfatal injuries as a result of road traffic crashes. The *National Highway Traffic Safety Administration* [1] (NHTSA) conservatively estimates that 100,000 police-reported crashes are the direct result of driver fatigue each year. All these statistics signal that nowadays drivers are often critically distracted and it is essential to develop a system which will detect driver drowsiness and decrease thus the number of car accidents.

In this paper we focus on the problem of detecting driver drowsiness through eye closure analysis to improve road safety. We detect the state of the eyes (open or closed), measure eye closure and compute blink frequency through video sequence analysis. All this information can warn a fatigued driver before a crash happens. Analysis of the driver's eye is a challenge due to the variety of lighting conditions while driving and irregular shape and color of human eyes.

Many techniques have been proposed already to measure the percentage of eye closure, or, at least to detect the eye status as being either open or closed. The analysis of the size of the iris surface [5] can be used to determine the state of the eye (open or closed); this method is based on template matching which has a high computational cost. Another study [2] detects the eyelids and iris using a circular Hough transform. The results of this approach show an accuracy rate of 88.7%, but the reported experiments have not been performed for a publicly available dataset. Since the shape of the eye varies for different subjects, the resulting locations are heavily affected by curvature of the eyelid, lighting conditions, and image contrast. The authors of [9] provided an eye-closure measurement method via analyzing the position of the upper and lower eyelids by searching for the change in average image intensity above and below the centre of an already detected eye. The change in image intensity, from dark eye region to light skin region, was captured by a simple vertical integral projection method. The authors report an average median error magnitude of 0.15 and an average 90^{th} percentile error magnitude 0.42 of eye closure, which is unreliable for differentiating eyes into states. The authors of [6] propose eye state-detection that uses statistical features such as sparseness and kurtosis of the histogram from the horizontal edge image of the eye.

We implemented and examined the method proposed in [6] and we noticed that the values of sparseness and kurtosis mainly depend on lighting conditions, and it also varies depending on the subjects. The method is a reasonable solution for determining open or closed eyes, but it is still difficult to measure the degree of eye closure. We also tested a method based on a circular Hough transform which proved to be unstable and worked only on high-resolution images.

Although, most methods are quite robust for non-challenging and normally illuminated scenes, we realized that due to frequent shadows and artificial lighting in day or night scenes, those methods are likely to fail. Furthermore, those methods are not yet tested on public databases, sources are not available, and it is difficult to compare them.

Another approach, which works under challenging lighting conditions, uses a nested cascade of classifiers for open or closed eye-status detection [11]. This detector can not only detect the eye status for frontal faces but also for rotated or tilted head poses in real-time driving applications. The authors obtained a high accuracy (of around 97%) for detecting the eye state, but the efficiency of this method highly depends on the training dataset, as well as on learning parameters.

The rest of this paper is structured as follows. The proposed system is outlined in Section 2. Section 3 provides experiments and obtained result. Section 4 concludes.

2 Proposed Method

The proposed method is based on eyelid detection and measurements of the distance between eyelids. The face and the eyes of a driver are first localized. After extracting the eye region, the proposed algorithm can detect eyelids and computes the percentage of eye closure. In the following sections, each stage of our algorithm is described in detail.

2.1 Face and Eye Detection

The first step of our algorithm is face and eye detection. Our approach uses the object detectors of Viola and Jones [15] and the application of Haar-like features. To determine the presence or absence of Haar-like features, we use *integral images*; see, e.g., [7]. For pixel location $p = (x, y)$, the *integral value*

$$I(p) = \sum_{1 \le i \le x \wedge 1 \le j \le y} P(i, j) \tag{1}$$

is the sum of all pixel values $P(q)$, where pixel location $q = (i, j)$ is not below, or not to the right of p assuming the origin in the upper left corner of the image carrier.

Considering $p_1, ..., p_4$ as the corners of a rectangle D, the sum of all pixel values for D equals

$$I(D) = I(p_4) + I(p_1) - I(p_2) - I(p_3) \tag{2}$$

See Fig. 1.

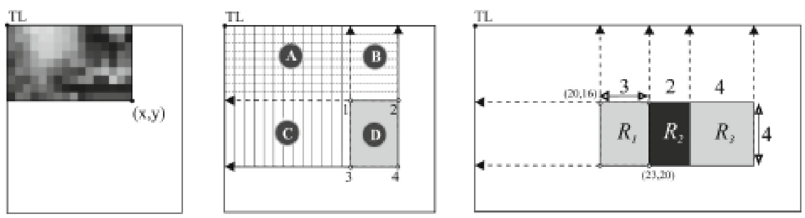

Fig. 1. Calculating integral values for a pixel location (x, y), a rectangle, and a Haar-like feature. Figure by courtesy of the authors of [13].

This technique is then also applied for time-efficient calculation of Haar-like features. Figure 1 illustrates for a Haar-like feature $R_1R_2R_3$. Values of contributing regions are weighted by w_i, thus creating a *feature value* for a given *Haar-like feature*. For the shown example, we have

$$I(F_k) = w_1 \cdot I(R_1) - w_2 \cdot I(R_2) + w_3 \cdot I(R_3) \tag{3}$$

The signs of w_i are opposite for light and dark regions.

First, the face is detected in the recorded image. Under non-ideal lighting conditions it is possible that one half of a face is darker than the another side. Inspired from [12] we divide a face into two halves and search for the eye region independently in each of them by adaptively changing detection parameters for the Haar-like object detector.

In order to detect an eyelid, we reduce the eye region of interest. After detection using Haar features, the eye's ROI includes useless information such as the eyebrow and the region between eyebrow and upper eyelid. First, we crop 40% of the upper part of the eye's ROI to remove the eyebrow from the cropped image window. The next step of the algorithm is binarization of the image window using the p-tile thresholding method of [4]. Inspired from [12], we experimentally obtained an adaptive p% value which depends on the intensity value I of the eye region by combining the *mode* M_o and the *mean* value as follows:

$$I = \frac{2}{3} \cdot M_o(E) + \frac{1}{3} \cdot \frac{1}{m} \cdot \sum_{i=1}^{m} E_i \tag{4}$$

Here, E is a eye region, and m is the total number of pixels in E.

After binarization, we detect vertical and horizontal borders in the binary image. Then, we select the largest border region in the image and calculated the bounding box of this region. We obtain a smaller eye region which includes only the iris, pupil, and eyelids. See Fig. 2.

Fig. 2. Example for finding a smaller eye region.

2.2 Eyelid Detection

Inspired from [8] we designed our own filters; see Fig. 3, for finding the position of eyelids. We used the following two simple facts in developing an algorithm for

1	...	...	1
-1	...	...	-1
-1	...	...	-1

Filter for upper eyelid

-1	...	...	-1
1	...	...	1

Filter for lower eyelid

Fig. 3. Designed filters for finding eyelids.

the detection of eyelids:

1. The upper eyelid is above the lower eyelid.
2. The surrounding region is brighter than the eye.

The eye region is now partitioned into five vertical strips. The filters are applied in each strip; we expect a maximum response at the eyelid position. We use the following median calculation to obtain the estimated position of each eyelid at time t:

$$V(t) = median\{P_1(t), P_2(t), P_3(t), P_4(t), P_5(t)\} \tag{5}$$

where P_i is the estimated position of an eyelid in each strip.

2.3 Calibration

The distance between eyelids depends on the shape of the eye and the distance between camera and face. We apply a calibration procedure during the recording of the first frames to estimate the distance between eyelids for fully open eyes and for closed eyes. During calibration, a user should look into straight into the camera and blink naturally without significant head movements. We experimentally obtained the value for the closed eye in the following way:

$$C = \frac{\frac{f_H}{35} + D[10\% \cdot S]}{2} \tag{6}$$

where f_H is the height of the face, D is a sorted vector of distances between eyelids in the first frames, and S is the size of this vector. We use the following equation to obtain the distance for a fully open eye:

$$O = D[85\% \cdot S] \tag{7}$$

2.4 Measurement of Eye Closure

The eye closure was measured by computing the distance between eyelids for both eyes. Since some people have long eyelashes, the position of lower eyelids moves further down when their eyes are closed. To avoid this problem we assumed that, during blinking, the lower eyelid does not change as much as the upper eyelid, so from the previous frames we obtained

$$\begin{aligned} L(t) = {} & 3\% \cdot P_{low}(t-5) + 7\% \cdot P_{low}(t-4) + 15\% \cdot P_{low}(t-3) \\ & +25\% \cdot P_{low}(t-2) + 50\% \cdot P_{low}(t-1) \end{aligned} \tag{8}$$

where $P_{low}(t)$ is a position of the lower eyelid at time t. Then, we obtained the position of the lower eyelid using the following equation:

$$P_{low}(t) = \frac{L(t) + P_{low}(t)}{2} \tag{9}$$

To improve our algorithm we assumed that both eyes give us the same information about eye closure. We use the following equation to measure the distance between eyelids. $D_R(t)$ and $D_L(t)$ are the distances between eyelids for the right and left eye at time t, respectively. We have that

$$D(t) = \begin{cases} max(D_R(t), D_L(t)) & \text{if } \frac{D_R(t)+D_L(t)}{2} > C \\ 0 & \text{if } \frac{D_R(t)+D_L(t)}{2} \leq C \end{cases} \tag{10}$$

The percentage of eye closure in previous frames supports the confidence for measuring the eye closure in the current frame. We obtained a value of eye closure from previous frames by using the following equation

$$\begin{aligned} M_p(t) = {} & 3\% \cdot M(t-5) + 7\% \cdot M(t-4) + 15\% \cdot M(t-3) \\ & +25\% \cdot M(t-2) + 50\% \cdot M(t-1) \end{aligned} \tag{11}$$

where $M(t)$ is the percentage of eye closure at time t. We assumed that the difference between the percentage of closure in the current frame and $M_p(t)$ should not be more than 50%. We compute a percentage of eye closure using the following equation:

$$M(t) = \begin{cases} \frac{D_R(t)}{O} \cdot 100 \text{ if } |M_R(t) - M_p(t)| \leq 50\% \wedge |M_L(t) - M_p(t)| > 50\% \\ \frac{D_L(t)}{O} \cdot 100 \text{ if } |M_L(t) - M_p(t)| \leq 50\% \wedge |M_R(t) - M_p(t)| > 50\% \\ \frac{D(t)}{O} \cdot 100 \quad \text{otherwise} \end{cases} \tag{12}$$

where $M_R(t)$ and $M_L(t)$ are the percentages of right and left eye closures at time t, respectively, and

$$D_L(t) = \begin{cases} D_L(t) \text{ if } D_L(t) > C \\ 0 \qquad \text{if } D_L \leq C \end{cases} \tag{13}$$

$$D_R(t) = \begin{cases} D_R(t) \text{ if } D_R(t) > C \\ 0 \qquad \text{if } D_R \leq C \end{cases} \tag{14}$$

3 Results

The proposed algorithm was tested on the BioID face database [14] and resulted in better eyelid detection rates than the method reported in [8], which also uses filters to locate positions of eyelids. The BioID face database consists of 1,521 grey-level images with a resolution of 384×286, which were taken under various lighting conditions before complex backgrounds, showing tilted and rotated faces. Our accuracy of eye detection is 97.8%. We analyzed all 1,521 images from the BioID database and marked positions of eyelids on them. We compared the results with true positions of eyelids and assumed that an algorithm gives a *good result* if the distance between eyelids for both eyes fulfils the following constraint:

$$|D_r - D_a| \leq 2 \tag{15}$$

where D_r is the true distance between eyelids and D_a is the calculated distance between eyelids obtained by an algorithm.

The proposed algorithm gives 93.1 % good results of measurement eye closure, compared to Ang Liu's method which gives 88.6 % good results. The *receiver operating characteristic* (ROC) curves for each experiment are shown in Fig. 4. The ROC is important for evaluating different methods for measurement of eye closure, and our proposed method gives better results than the other method.

We also tested the proposed algorithm on our own video sequences [3] under different lighting conditions with several people of different ages. For results for the detected eye status, see Table 1. Videos 1-5 were taken under day light conditions. The proposed algorithm has a good performance under day light conditions. However, analyzing video 6 under dark conditions gave a relatively poor result due to shadows caused by artificial lighting. See Fig. 5 for illustrations.

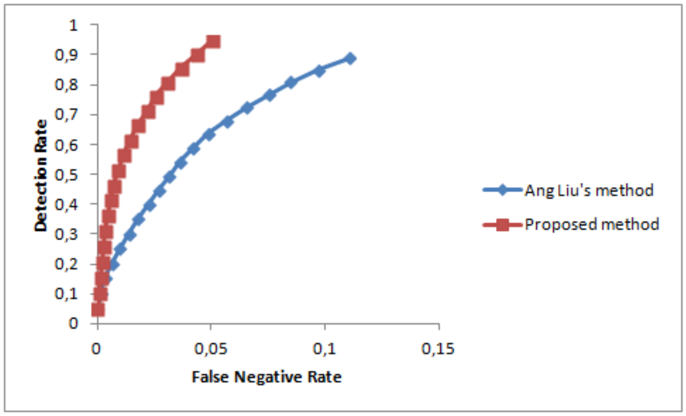

Fig. 4. ROC curve comparison between proposed method and Ang Liu's method.

Table 1. Accuracy of the proposed method in terms of detection rate.

	Open eye	Closed eye
	Detection rate	
Video 1	99.2%	94.4%
Video 2	99.7%	97.3%
Video 3	97.5%	91.4%
Video 4	99.2%	92.0%
Video 5	98.6%	97.8%
Video 6	96.0%	90.0%

4 Conclusions

In this study, we proposed a driver-monitoring system that can be used to raise a warning alarm in case of driver drowsiness. Accurate measurement of eye closure is a challenging topic in computer vision. Some of the challenges are that various subjects have different shapes of eyes or different lengths of eyelashes, and some use sunglasses while driving. Also, driving conditions are a difficult research context as images even from high-resolution camera might be noisy, which may prevent algorithms from working effectively. Under such conditions

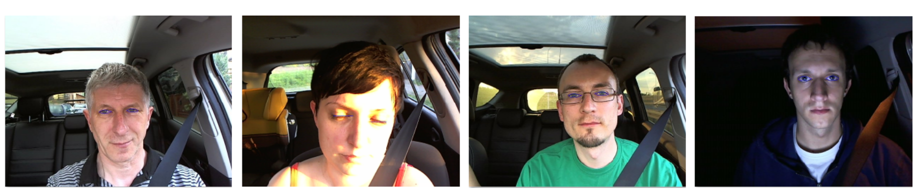

Fig. 5. Results of the proposed method under different lighting conditions.

we are sometimes not able to locate exactly the position of eyelids even by the naked eye. According to experimental results performed on low resolution images, our method provides more accurate results than other methods. Our method still requires further improvements for dark light conditions and also more experiments for subjects wearing sunglasses.

References

1. *NHTSA 2009, [online]*. http://www.nhtsa.dot.gov/.
2. Akrout, B., Mahdi, W.: A blinking measurement method for driver drowsiness detection. Advances in Intelligent Systems and Computing, 226:651–660 (2013)
3. Daniluk, M., Rezaei, M., Nicolescu, R., Klette, R.: Monocular driver monitoring under different lighting conditions. Available in enpeda image sequence analysis test site (EISATS), Set 11. www.mi.auckland.ac.nz/EISATS (2014)
4. Doyle, W.: Operations useful for similarity-invariant pattern recognition. J. ACM, 9(2):259–267 (1962)
5. Horng, W. B., Chen, C. Y., Chang, Y., Fan, C. H. : Driver fatigue detection based on eye tracking and dynamic, template matching. In Proc. Networking Sensing Control, volume 1, pp. 7–12 (2004)
6. Jo, J., Jung, H. G., Park, K. R., Kim, J., Lee, S. J. : Vision-based method for detecting driver drowsiness and distraction in driver monitoring system. Optical Engineering, 50(12):127202–127202 (2011)
7. Klette, R.: Concise Computer Vision. Springer, London (2014)
8. Liu, A., Li, Z., Wang, L., Zhao, Y. : A practical driver fatigue detection algorithm based on eye state. In Proc. Asia Pacific Conf. Postgraduate Research Microelectronics Electronics (PrimeAsia), pp. 235–238 (2010)
9. Malla, A. M., Davidson, P. R., Bones, P. J., Green, R., Jones, R. D.: Automated video-based measurement of eye closure for detecting behavioral microsleep. In Proc. Engineering Medicine Biology Society (EMBC), pp. 6741–6744 (2010)
10. Omidyeganeh, M., Javadtalab, A., Shirmohammadi, S.: Intelligent driver drowsiness detection through fusion of yawning and eye closure. In Proc. Virtual Environments Human-Computer Interfaces Measurement Systems (VECIMS), pp. 1–6 (2011)
11. Rezaei, M., Klette, R.: 3D cascade of classifiers for open and closed eye detection in driver distraction monitoring. In Proc. Computer Analysis Images Patterns, pp. 171–179 (2011)
12. Rezaei, M., Klette, R.: Adaptive Haar-like classifier for eye status detection under non-ideal lighting conditions. In Proc. Image Vision Computing New Zealand, pp. 521–526 (2012)
13. Rezaei, M.: Artistic rendering of human portraits paying attention to facial features. In Proc. Arts Technology, pp. 90–99 (2012)
14. *The BioID face database, [online]*.
http://www.bioid.com/downloads/facedb/facedatabase.html
15. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In Proc. Computer Vision Pattern Recognition, pp. 511–518 (2001)
16. World Health Organization: WHO global status report on road safety 2013: supporting a decade of action (2013)