Eye blink based fatigue detection for prevention of Computer Vision Syndrome

Matjaž Divjak
Inst. for Comp. Graphics and Vision, TU Graz
Inffeldgasse 16a, 8010 Graz, Austria
divjak@icg.tu-graz.ac.at

Horst Bischof
Inst. for Comp. Graphics and Vision, TU Graz
Inffeldgasse 16a, 8010 Graz, Austria
bischof@icg.tu-graz.ac.at

Abstract

Living in information society we spend a lot of time in front of visual display units such as computer monitors or TV screens. This has positive as well as negative effects on our lives. Negative effects are mostly health related and one major concern is the increasing number of people affected by the Computer Vision Syndrome (CVS) [1]. Most CVS-related problems can be avoided by suitable preventive actions, but the majority of computer users ignore them. To help remedy the situation we present a prototype of a computer vision system for real-time eye blink based detection of eye fatigue, which is a common symptom leading towards CVS. Our program records video of the user by a computer monitor mounted web-camera, processes the frames and detects hazardous behaviour based on user's eye dynamics and blink patterns. Experimental results show that the proposed system is capable of detecting common cases of fatigued behaviour linked with long-term computer use. We believe regular use of the proposed system could significantly reduce the CVS symptoms for long-time computer workers. However, the techniques used can also be applied to other similar applications, such as driver fatigue detection, human-computer interaction, etc.

1. Introduction

Long-term use of a visual display unit (VDU) device such as a computer monitor or a TV screen, can negatively affect user's health. Symptoms such as headaches, fatigue, blurred vision, eye strain, dry/irritated eyes or focusing difficulties are signs of Computer Vision Syndrome (CVS) and Asthenopia [2]. They are mostly caused by decreased blinking reflex and tightening of the inner eye muscles [3][4]. Recent studies show that 70 % of computer workers worldwide report having vision problems. More alarmingly, the number of affected people keeps increasing [1].

Most of CVS-related problems can be avoided by appropriate preventive measures, such as correct VDU setup, regular breaks and eye-muscle exercises. Unfortunately, the majority of computer users are not aware of the problem, and even those who are, find it very difficult to take regular breaks when they concentrate on work.

Several software packages exist to help the users manage their rest brakes and exercises [5][6], but they all have serious limitations in detecting the user's activity level. This activity level is usually determined from keyboard or mouse use and doesn't correctly represent the behaviour when user is statically observing the screen (for example, during reading).

To address those issues we present a prototype of an application that uses CV techniques to detect unhealthy behaviour during computer use. The application uses standard monitor-mounted web camera to track user's eye dynamics and blinking. Frequency and duration of blinks as well as eye closure speed are the primary indicators of eye fatigue. When those parameters reach critical levels, the system can warn the user and offer him instructions for relaxation as well as interactive eye-muscle exercises.

The next section reviews a few recent publications that are similar to our approach. Section 3 describes the process of detecting and tracking user's face and eyes, as well as steps involved in detecting blinks. Metrics for comparing blinking behaviour are presented in Section 4. Section 5 explains differences between normal blinking behaviour and eye fatigue. Performance of the proposed application is tested in Section 6, followed by discussion and conclusion.

2. Overview of existing approaches

Since our application is targeted for a general computer user, we decided to capture input data using a low cost web-camera, without any additional hardware (like infrared illumination). The application must be able to process video in real-time and under varying indoor conditions (typical office environment), achieving adequate accuracy for reliable detection of eye blinks.

Several eye tracking approaches that address similar issues were already published, but only a few of them address all of the mentioned constraints. For example, Morris et al. [7] proposed a blink detection system based on variance map calculation and eye corner analysis. It runs in real-time on 320 × 240 images. They report good blink detection results (95 % true positives), but head movements affect the variance map computation and cause a sharp drop in performance.

Sirohey et al. [8] presented an approach for determining eye blinks by locating eye corners, eyelids and iris and analyzing their movements. Motion information is estimated using normal flow. Head motion is modelled separately by an affine model and is used to decouple eye movements from the head movement. Authors claim their algorithm can track iris and eyelid motion correctly more than 90 % of the time, but not in real-time. In a later paper [9] they added a deterministic finite state machine to analyze the normal flow and estimate blink characteristics.

Chau and Betke [10] described a system that detects blinks in real-time by correlation with an open eye template. For 320 × 240 images obtained from a webcam they report 95 % overall blink detection accuracy. The main benefit of such template-based approach is its low computational complexity. The downside is that it only distinguishes between two eye states, open and closed. Any movement in-between is not well defined.

Pan et al. [11] use a boosted classifier to detect the degree of eye closure. The changing of eye states is modelled by a Hidden Markov Model. Their method operates in real time on 320 × 240 webcam images, detecting more than 96 % of eye blinks. Since examples of typical eye motion are used for training the model, the method should face difficulties when non-standard eye motion occurs, such as partial blinks.

Recently, Orozco et al. [12] proposed using two appearance-based trackers: the first one tracks iris movements while the second one focuses on eyelids and blinking. Using low resolution input video and a simple appearance model, the method reportedly runs in real-time, achieving good tracking results. Authors didn't try to detect eye blinks; however the method could be used for this purpose.

Most of the approaches described above detect blinks by locating separate eye parts such as iris, eyelids and eye corners. Quality of detection is directly linked to accuracy of feature localization. When low quality webcam images are used as input, sufficient accuracy is very difficult to achieve. Additionally, real-time constraint prevents us from using sophisticated eye localization techniques.

3. Detecting eye blinks

Our video source is a monitor-mounted web-camera, so we assume that a reasonably large face image is available. Each frame of the input video is processed by 3 different classifiers to detect locations of user's face, left and right eye [13]. The candidate regions are verified using a set of simple geometrical constraints. The classifiers were trained offline using OpenCV's version of AdaBoost [14]. If user's face is not in approximately frontal position, the classifier will fail to detect it. In such cases, a set of feature points [15] is detected inside the face region of the previous video frame. The features are tracked using standard Lucas-Kanade algorithm and their position in the current video frame is used to reconstruct the appropriate face region [13]. If no face is detected for several video frames, the tracking is reinitialized.

Our blink detection algorithm is partially based on work of Duric et al. [9], but with differences in optical flow estimation and different approach to extraction of final blink data. It consists of the following steps (Fig. 1):

1. calculation of optical flow for the face region, offloaded to Graphical Processing Unit (GPU) by help of the OFLib library [16];
2. compensation of optical flow of the eyes for global face movement;
3. optical flow normalization for accommodating differently sized faces;
4. rotation of flow vectors to correct for non-horizontal eye positions;
5. estimation of dominant vertical eye movement direction;
6. extraction of raw blinks by adaptive thresholding of the processed flow data.

This way we can detect closing and opening for each eye and each video frame separately. Additionally, results for left and right eyes are combined for increased robustness, giving us final blinking information.

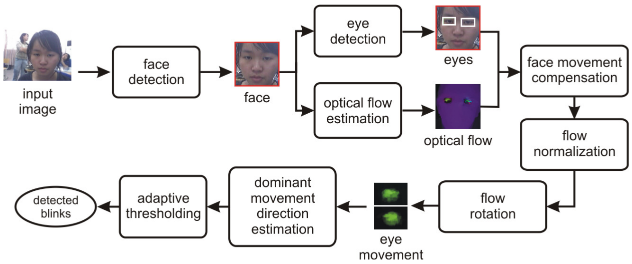

Figure 1: Overview of the eye blink detection process.

4. Eye blink analysis

This basic blinking information is used to calculate three metrics that describe changes in blink behaviour. After preliminary tests we concluded that decreased blinking reflex is best described by:

1. blink frequency (BF),
2. average blink duration (ABD),
3. average eye closure duration (AECD).

BF tells the number of blinks per given timeframe. ABD measures the mean duration of complete blinks in a given time interval. AECD metric measures the duration of only the first part of the blink when the eye is closing. In our program we repeatedly evaluate the BF, ABD and AECD metrics in 1-second intervals. Each time all the blinks from the past 1 minute are considered. This enables us to fairly precisely track the changes in blinking behaviour and quickly discover any significant deviations which might indicate eye fatigue.

5. Detecting eye fatigue

In order to quantitatively evaluate typical properties of normal and fatigued eye behaviour, we recorded two sets of test videos. The first set contains recordings of people who actively change their gaze during work and blink regularly. The second set contains recordings of people who read information from the screen for long periods of time, do not blink enough, squint their eyes or stare statically at the screen. Each set currently consist of 5 videos with approximately 2000 video frames each, with 3 different individuals.

Fig. 2 depicts a comparison between normal and fatigued behaviour of a typical individual. As expected, it is obvious that during fatigued behaviour the blinking frequency is significantly decreased (in our case BF drops by more than 40 %) and eye closes much slower than during normal behaviour (AECD is increased by 25 - 40 %). Blink duration is also increased, but not enough to be a reliable sign of fatigue. We noticed that people tend to fight eye fatigue by short bursts of rapid blinking, which could explain why the ABD is not greatly affected.

The differences in BF and AECD values observed from Fig. 2 can serve as relative thresholds that signal the onset of eye fatigue. Of course, good threshold values are critical for reliable performance of the system. The values shown give reasonably good performance in our tests, but for real-world use either the video sets must be expanded or each user should have the option to adapt those thresholds to his own behaviour patterns.

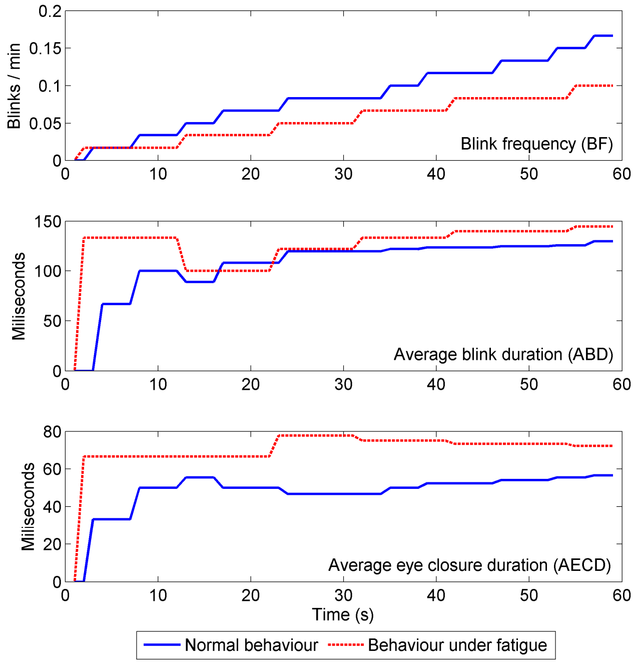

Figure 2: Illustration of differences between normal eye behaviour (solid line)) and fatigue (dashed line) in a typical test video. Blink data from the past 1 minute was evaluated with 1 second resolution.

6. Experiments

Performance evaluation of eye blink detection systems has one important problem: there are no standard evaluation videos with ground truth data that would enable researchers to compare results. Instead, algorithms are usually evaluated on privately recorded videos with unknown ground truth.

We were forced to do the same for videos depicting normal/fatigued eye behaviour used in previous section. All videos were recorded using monitor-mounted web-camera in a typical office environment, with resolution of 320 x 240 and 30 frames per second (10 videos, 20000 frames total, 3 individuals, 150 blinks total). Videos show upper body of a person approximately facing the camera, with moderate body movements. All videos were manually annotated with ground truth data.

For evaluation of general blink detection performance, two publicly available datasets are used. The ZJU database (80 short videos, 11800 frames total, 20 individuals, 255 blinks total) was recently used in [11] to evaluate their blink detection algorithm. Simple ground truth is available, but we extended it by additionally annotating beginnings and endings of blinks. The second dataset is the Talking Face Video [17] (1 video, 5000 frames total, 1 individual, 61 blinks total). Ground truth is provided, but doesn't include blink information; therefore we manually annotated the video in the same way as the ZJU dataset.

For easier comparison with existing methods, we also provide results of a "standard" blink detection algorithm based on correlation of detected eye region with offline templates of open and closed eye [10].

Current version of our program runs in real-time, at approximately 20 - 30 frames per second. This is enough to reliably detect even very short blinks, which are typically shorter than 100 ms.

6.1 Face and eye detection

Performance of classifier-based face and eye detectors used in our algorithm (Section 3) is summarized in Table 1. In cases where detectors failed the Lucas-Kanade tracker provided intermediate values, resulting in 100% detection of faces and eyes in all test videos, without false positives.

Table 1: Face and eye detection performance of classifiers used in our tracking algorithm, evaluated on four different datasets. Mean and std. dev. values are shown.

Dataset	Faces (%)	Left eyes (%)	Right eyes (%)
ZJU	100 ± 0	90 ± 11	98 ± 3
TALKING	100	99.8	99.6
Normal	95 ± 11	97 ± 5	97 ± 7
Fatigue	100 ± 0	99 ± 1	99 ± 1

6.2 Blink detection

Performance of blink detection was measured by comparing how many detected blink intervals intersect with ground truth values. Tables 2 and 3 show the average rate of true positive (TP) and false positive (FP) detections, along with calculated accuracy measure. Table 2 is for our flow-based blink detection, while Table 3 is for standard template-based detection.

Results show that our algorithm outperforms standard template-based approach and can detect blinks with reasonable accuracy (more than 90 %), but can have problems with false detections in cases where the eyes are rapidly moving up and down.

Table 2: Performance of our optical flow based blink detection algorithm, evaluated on four datasets.

Dataset	Mean TP rate (%)	Mean FP rate (%)	Mean accuracy
ZJU	95 ± 12	2 ± 6	97 ± 7
TALKING	95	19	88
Normal	98 ± 4	16 ± 12	91 ± 5
Fatigue	77 ± 9	7 ± 10	85 ± 1

Table 3: Performance of template-based blink detection (our implementation), evaluated on four datasets.

Dataset	Mean TP rate (%)	Mean FP rate (%)	Mean accuracy
ZJU	66 ± 31	17 ± 24	76 ± 17
TALKING	90	16	87
Normal	66 ± 30	26 ± 39	70 ± 25
Fatigue	47 ± 52	72 ± 1	36 ± 25

6.3 Analysis of blink behaviour

In this experiment all three metrics for analyzing blink behaviour (Section 4) were evaluated on all test videos. The time interval used was equal to the duration of each video. Table 4 presents ground truth values, calculated from manually annotated blink data. Table 5 presents results estimated by our flow-based method and Table 6 results of the standard template-based method.

All obtained results are within normal physiological limits. The values estimated by our method are much closer to reference values then results of the template-based method.

Table 4: Ground truth values for blink behaviour metrics. BF – blink frequency, ABD – avg. blink duration, AECD – avg. eye closure duration.

Dataset	BF (blinks/min)	ABD (ms)	AECD (ms)
ZJU	40 ± 16	391 ± 135	105 ± 45
TALKING	21	390 ± 254	84 ± 27
Normal	18 ± 3	290 ± 101	74 ± 21
Fatigue	8 ± 1	246 ± 73	75 ± 15

Table 5: Blink behaviour metrics obtained by our flow-based algorithm. BF – blink frequency, ABD – avg. blink duration, AECD – avg. eye closure duration.

Dataset	BF (blinks/min)	ABD (ms)	AECD (ms)
ZJU	40 ± 16	252 ± 148	68 ± 21
TALKING	25	475 ± 935	78 ± 46
Normal	22 ± 5	264 ± 207	66 ± 23
Fatigue	6 ± 1	262 ± 145	68 ± 11

Table 6: Blink behaviour metrics obtained by template-based blink detection. BF – blink frequency, ABD – avg. blink duration, AECD – avg. eye closure duration.

Dataset	BF (blinks/min)	ABD (ms)	AECD (ms)
ZJU	44 ± 21	161 ± 113	83 ± 63
TALKING	24	238 ± 339	113 ± 128
Normal	30 ± 28	142 ± 83	89 ± 46
Fatigue	20 ± 9	130 ± 111	76 ± 53

7. Discussion and conclusion

The results presented in Section 6 clearly show that our flow-based algorithm for detecting eye blinks significantly outperforms the popular template-based method. Comparison with more sophisticated methods is difficult due to lack of standard testing datasets. We expect that methods which identify separate eye parts will perform better that the proposed approach, but such methods are much more computationally intensive and currently unsuitable for real-time operation. When compared with approaches described in Section 2, our algorithm achieves approximately the same level of performance (more than 95 % TP detections), but allows more head movements and runs in real-time.

Our experiments demonstrate that we can detect the onset of eye fatigue and identify the moments when user needs a short intervention. This intervention should be in the form of a rest brake or exercise and should help the user to relax the eye muscles, increase the blinking rate and reduce the eye fatigue. We believe such interactive monitoring system can be used to detect dangerous eye behaviour during computer work and can help in timely prevention of CVS-related symptoms.

In the next stage of the project we plan to perform additional experiments with a number of rested and fatigued individuals. This way we will obtain good relative threshold values for detecting the onset of eye fatigue, as well as confirm the usefulness of the proposed approach.

Acknowledgement

This work was supported by a Marie Curie Intra-European Fellowship within the 6th European Community Framework Programme (project PRE-WORK, contract no. 041395).

References

[1] C. Blehm, et al.: “Computer Vision Syndrome: A Review”, *Survey of Ophthalmology*, Elsevier Inc., vol. 50, no. 3, 2005.

[2] R. Iribarren, A. Fornaciari, G. K. Hung: “Effect of cumulative nearwork on accommodative facility and asthenopia”, *International Ophthalmology*, Kluwer Academic Publishers, vol. 24, no. 4, pp. 205–212, 2001.

[3] J. Bali, N. Navrin, B. R. Thakur: “Computer vision syndrome: A study of the knowledge, attitudes and practices in Indian Ophthalmologists”, *Indian Journal of Ophthalmology*, vol. 55, no. 4, pp. 289-294, 2007.

[4] “Working with VDUs”, Health and Safety Executive Books, UK, December 2006.

[5] “WorkPace”, Wellnomics Ltd., http://www.workpace.com.

[6] “EyeShield”, Nimble Software, http://www.nimblesoftware.com/desktop/eyeshield/.

[7] T. Morris, P. Blenkhorn, F. Zaidi: “Blink detection for real-time eye tracking”, *Journal of Network and Computer Applications*, vol. 25, num. 2, pp. 129-143, April 2002.

[8] S. Sirohey, A. Rosenfeld, Z. Duric: “A method of detecting and tracking irises and eyelids in video”, *Pattern Recognition*, vol. 35, num. 6, pp. 1389-1401, 2002.

[9] Ric Heishman, Z. Duric: “Using image flow to detect eye blinks in color videos”, Proceedings of the 8th IEEE Workshop on Applications of Computer Vision, pp. 52, 2007.

[10] M. Chau, M. Betke: “Real Time Eye Tracking and Blink Detection with USB Cameras”, Boston University Computer Science Technical Report No. 2005-12, May 2005.

[11] G. Pan, et al.: “Eyeblink-based anti-spoofing in face recognition from a generic webcamera”, The 11th IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil, October 2007.

[12] J. Orozco, et al.: “Real-time gaze tracking with appearance-based models”, *Machine Vision Applications*, 2008.

[13] M. Divjak, H. Bischof: “Real-time video-based eye blink analysis for detection of low blink-rate during computer use”, 1st International Workshop on Tracking Humans for the Evaluation of their Motion in Image Sequences (THEMIS), Leeds, UK, pp. 99 – 107, September 2008.

[14] R. Lienhart, J. Maydt: “An Extended Set of Haar-like Features for Rapid Object Detection”, IEEE ICIP 2002, vol. 1, pp. 900-903, September 2002.

[15] E. Rosten, T. Drummond: “Machine learning for high-speed corner detection”, European Conference on Computer Vision, Graz, Austria, vol. 1, pp. 430-443, May 2006.

[16] C. Zach, T. Pock, H. Bischof: “A Duality Based Approach for Realtime TV-L1 Optical Flow”, Proceedings of the 29th DAGM Symposium on Pattern Recognition, Heidelberg, Germany, pp. 214-223, 2007.

[17] “Talking Face Video”, Face&Gesture Recognition Working group, IST-2000-26434, htttp://www-prima.inrialpes.fr/ /FGnet/data/01-TalkingFace/talking_face.html.