# Paper

# A Study on Tiredness Measurement using Computer Vision

MD. Alomgeer Hussein\*† , Md. Talal Bin Noman‡ Md. Atiqur Rahman Ahad†

(Received July 22, 2017, revised September 20, 2018)

Abstract: In this paper, we studied tiredness measurement based on several different detection methods in real time. We know that the driver tiredness is one of the major causes of traffic accidents. So tiredness detection can play a vital role for preventing road accidents. By developing an automatic solution for alerting drivers of tiredness before an accident occurs, this could reduce the number of traffic accidents. The Haar-cascade classifier is exploited based on Haar-like features to find the eyes. The main purpose of the Haar-cascade classifier is to classify closed or open state of the eyes. If we can notice that the eyes are closed for a predefined span of time, we consider the state of the eyes can be closed. Based on this closed-state of the eye, a notification (like alarm) is initiated to alert. We have detected only right eye for saving processing load on the system. The reason is that when a person closes his eyes he usually does not close one eye, but both eyes at the same time. Several steps are taken into account for this system; we first capture the frame from the webcam. Then we need to detect face as well as eye. To detect blinks, we process ROI (region of image) of pupil area. Our result is found to be satisfactory.

Keywords: Tiredness Measurement, Accidents Notification, Computer Vision

#### 1. Introduction

Computer vision-based activity recognition, behavior understanding, gait analysis, emotion deciphering, fall detection assessment, eye-blink study, face recognition, etc. are major application areas for the last decade in different fields [3] [4] [7] [9] [12] [13] [16] [23] [24]. Studies related to tiredness evaluation or eye-blinking rate becomes necessary in this arena for some important applications, especially in driver's alert system from micro-sleep. Driver alert system is very much required if possible [1] [2] [5] [35] [36]. This field is very challenging due to multiple factors related to the position of the eyes, head movements, background lights and vicinity, distance, etc. In this paper, we attempt to deal this important research area. A large number of road accidents are occurred due to the driver tiredness. So in the field of active safety research, developing a system for assessing driver's alertness level is becoming a central issue [1][2][5]. Eye detection and eye blink detection are very informative that has versatile applications. Eye blinking has a wide range of applications in human-computer interaction (HCI) like the eye typing applications as in [8], for disabled persons to operate all mouse operations, driving safety for general people like Google glass and so on.

The primary goal of this paper is to produce a system to detect tiredness using three different methods based on thresholding, distance from the eyelid and angle of the head. Thorough these three methods, the human eyes are the main aspect as eyes are one of the most important salient. The problem of human eye detection and blink detection has received significant attention during the past several years because of human-computer interaction and surveillance applications [14]. In this system high resolution cameras were used, for example in [10] with a focus on high resolution images of the eye-region which simplifies eye-features detection. In the past ten or fifteen years, the driver safety problem was the main concern for many countries all over the world and begun to pay attention to that problem. In accordance to do that they started to investigate the driver's mental states relating to driving safety. That is why they proposed some driver tiredness detection methods which can detect whether the driver is tired, such as drowsiness, and for generating some warning alarms to alert the driver [1][2]. Using computer vision technology of the existing approaches systems for extracting characteristics of a driver is very successful though but it is a challenging issue due to a variety of factors. External illumination interference, lighting conditions and the variety of eyes moving speed is the main reason [11].

The paper is organized as follows: after introduction we present some related works in Section 2. In Section 3, we presented the methodology of this work. Then we presented the experimental results and analysis in Section 5. Finally, we concluded the paper in Section 6 with a few future work guidelines.

<sup>\*</sup> Corresponding author: mdalomgeerhussein@gmail.com

<sup>†</sup> Department of Electrical and Electronic Engineering, University of Dhaka, Dhaka-1000, Bangladesh

<sup>&</sup>lt;sup>‡</sup> Department of Computer Science and Engineering, City University, Dhaka, Bangladesh

Table 1: Different systems and the corresponding references.

| Systems         | References               |
|-----------------|--------------------------|
| USB Based       | [14], [26]               |
| Webcam Based    | [8], [20]                |
| Real Time       |                          |
|                 | 2 3, 2 3                 |
| Special Devices | [21], [29]<br>[28], [30] |

#### 2. Literature Survey

In this section, we try to cover some important methods that work with eye movement and gaze tracking, especially with eye blink systems and its applications [1][2][5][19][22][27].

A real time method based on some captured video and image processing algorithms for eye blink detection is proposed by Mohammed et al. [21]. But it works on an offline captured video not online video. USB camera is used by Chau and Betke [14] as well as Magee et al. [26]. A blink detection system based on real time eye tracking with USB camera is proposed by Chau and Betke [14]. They used template matching approach to detect eyes and analyse eyeblinks. They achieved 95.3% blink detection accuracy. It can distinguish between two eye-states, open and closed. On the other hand Magee et al. [26] also implemented in USB camera. In their system, they used multi-scale template correlation for tracking the face, left and right eyes was exploited to detect if the computer user is looking on camera. To control computer applications they used the detected eye direction.

Gruman *et al.* [25] proposed two video based HCI tools called Blink Link and Eyebrow Clicker. The first tool can automatically detect eye blinks of a user's and it can measure their duration. A long blinks can trigger mouse clicks, while short blinks are ignored. Blink patterns are used for this system. The other tool can automatically detect trigger a mouse click when a user raises his or her eyebrows.

Webcam-based system is also available, e.g., in [8] [20]. Galab et al. [8] presented a webcam-based system for detecting eye blinks accurately without any restriction on the background. It automatically classifies the eye. The proposed system is tested with the users who wear glasses. Elahi et al. [20] proposed a webcam-based eye-tracker, especially for low power devices. They used mean of gradient vector algorithm to detect eye-ball center position. Tobii eyeX device [28] is one of the most fundamental things which can now be used to control PC by using eye blink. Uzma et al. [7] explored Tobii eye-tracker to visual face scanning and emotion perception analysis. Ref. [6] proposed a system to analyze the performance of eye-blink detection using android mobile phone.

Eye Tribe tracker [30], the first affordable eye tracker that precisely determines the on-screen gaze position. It also enables interact devices with human being by blinking. We can summarize this related field in the Table 1.

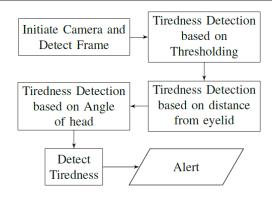


Figure 1: General flow diagram of the algorithm.

### 3. Methology

This paper is to produce a system to detect tiredness measurement using three different methods based on (A) thresholding, (B) distance from the eyelid, and (C) angle of the head in a real-time approach. In accordance to do this detection, first we need to detect eyes with Haar-cascade classifier using webcam and try to detect the presence of tiredness with these methods. The whole process is illustrated in the flowchart (see Fig. 1).

The real time application can be divided into six steps: capture image from live video webcam, face detection, eye detection, processing of ROI, blink detection and alert. The first step of the total process is the capturing frame from webcam. Afterwards, the colored frames will be converted to gray scale frames by extracting the luminance component [18]. When the program starts, at first, it searches for faces in front of the camera. If there is none then it starts monitoring changes of the image. If the change is significant then it checks for face in the image. The Haar classifier is used for face detection. Haar-cascade is feature-based face detection algorithm in cascaded form. It detects face features one after one in cascaded form. If one fails to detect it doesn't advance to the next feature. Feature detection also contains a value for relative percentage of matching. If the total percentage of matching of all key features is less than threshold then there is no face in the region. Haar classifier rapidly detects any object, based on detected feature not pixels [6] [15] [17]. The main advantage of Haar features is their computational efficiency [15]. The third step is to detect the eyes. Eye detection can be done by various approaches. We consider the Haar cascade classifier. We do not exploit any feature based eye detection algorithm. We just use the fact that, if the face is not out of the ratio eyes are placed in a certain position of face while detecting the eye (in Fig. 2).

ROI means region of image. We have taken only one fourth portion of the face and only a single eye for further analysis (Shown in Fig. 3).

Blinking is rapid closing of the eyelid. The detection of blinking is based solely on observation of the previous step using the template of the user's eye. In this approach, open eyelid is taken as a template for detecting eye blink. As the user's eye opens during the process of a blink we try to match by frame. From the flow chart (in Fig. 4) we can de-





(a) open state

(b) close state

Figure 2: Detection of face and eye in an open state and a close state.



Figure 3: ROI in open eye state.

duce some steps, are describing below in short form: Step 1: In the first step, we read data from webcam. Read data file containing feature details of face. Search for face feature match in every frame. Crop the face region. Step 2: Crop the eye-region from the face region. Eye region also contain eyebrows. Step 3: Reduce eye region, take ROI of pupil area and track the pupil. Step 4: Check the conditions for tiredness detection Step 5: If the assigned condition is true then the tiredness is detected and the algorithm will notify otherwise the system will find another sample frame from the webcam.

## 4. Experimental Setup

We developed a real-time webcam based eye-blink detection system using python programming language. The resolution of the webcam is  $1280 \times 720$  pixels. Python is an interpreted, object oriented high level programming language with dynamic semantics. And OpenCv is the most popular and portable open source computer vision library available for developers. OpenCV- Python makes use of Numpy, which is a highly optimized library for numerical operations with MATLAB-style syntax. All the OpenCV array structures are converted to and from Numpy arrays. This also makes it easier to integrate with other libraries that use Numpy such as SciPy and Matplotlib [32]. We used a personal computer with core i7 processor and 640 GB hard drive to run the python program. This system is very easy to configure and use as well as works with webcam and runs at a frame rate of 30 frames per second [14].

# **4.1 Tiredness Detection Based on Thresholding** If E(x, y) is a threshold version of f(x, y) at some threshold T.

$$E(x,y) = \begin{cases} 1, & \text{if } f(x,y) \ge T \\ 0, & \text{otherwise} \end{cases}$$
 (1)

Here f(x, y) is a function of black pixels. When the black

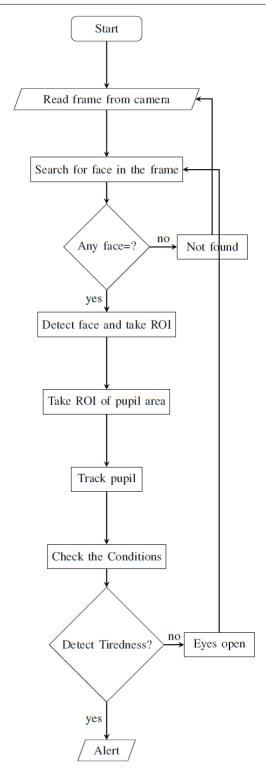


Figure 4: Flowchart of the system.

pixel is greater or equal than the threshold value then the eyes are open and when the black pixel is smaller than the threshold value then the eyes are closed. This system provides the necessary analysis: Step 1: In the first step, we read data from webcam. Read data file containing feature details of face. Search for face feature match in every frame. Crop the face region. Step 2: Crop the eye-region from the face region. Eye region also contain eyebrows. Step 3: Re-

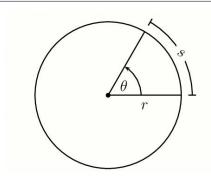


Figure 5: To measure tiredness using angle of head.



Figure 6: Comparison of tiredness at different angles of head.

duce eye region, take ROI of pupil area and take threshold data from ROI pupil. Step 4: Check the number of black pixel with threshold values, if the black pixel exceeds the threshold value then the eyes are open and if not then the eyes are closed. Step 5: If the minute limit is smaller than 'how long the eyes are closed' the system will show alert, and if the minute limit is greater than 'how long the eyes are closed' the system will find another sample frame from the webcam.

**4.2 Tiredness Detection Based on Angle of head** Tiredness detection based on Angle of Head is a real time detection approach. As the user's eye opens during the process of measure the angle of head we try to match the open eye template in the eye region rectangle area frame by frame. We can also mention a theorem of angle measurement.

Here is the relationship between the radius, and the central angle in radians (in Fig. 5). The formula is  $S = r\theta$  where s represents the arc length,  $S = r\theta$  represents the central angle in radians and r is the length of the radius. Firstly, we need to convert the value of  $\theta$  to degrees because the whole system works only in degrees. A fixed value of  $\theta$  is taken to detect tiredness of the driver. We compare the tiredness of the driver at different angle level of the head (in Fig. 6). We assume a right angle where  $\theta = 90^{\circ}$  as a straight head. There is 'tired' and 'heavy tired' notification when the value of  $\theta$  is  $3^{\circ}$  above the right angle ( $\theta = 93^{\circ}$  or  $\theta = 93^{\circ} + +$ ) and further increases.

This system also provides the necessary analysis using previously discussed method of face detection and eye blink detection. Step 1: In the first step, we read data from webcam. Read data file containing feature details of face. Search for face feature match in every frame. Crop the face region. Step 2: Crop the eye-region from the face region. Eye region also contain eyebrows. Step 3: Reduce eye re-

gion, take ROI of pupil area and take pixel data from ROI pupil. Step 4: Check the number of black pixel with predefined black pixel data, if the black pixel exceeds the predefined black pixel value then the eyes are closed and if not then the eyes are opened. Step 5: If the minute limit is smaller than 'how long the eyes are closed' the system will show alert, and if the minute limit is greater than 'how long the eyes are closed' the system will find another sample frame from the webcam.

#### 4.3 Tiredness Detection Based on Distance from Eye-

**lid** In this approach, open eyelid is taken as a template for detecting distance. As the user's eye opens during the process of measure the distance we try to match the open eye. Though we used a fixed balck pixel value (for this case minimum black pixel value is 80) to measure the tiredness. We can use the 'Euclidean distance' formula while measuring distance from eyelid. The Euclidean distance is a distance measure between two points based on Pythagoras' theorem. According to the Euclidean distance formula, the distance between two points in the plane with coordinates (x, y) and (a, b) is given by

$$d((x,y),(a,b)) = \sqrt{(x-a)^2 + (y-b)^2}$$
 (2)

This system also provides the necessary analysis using previously discussed method of face detection and eye blink detection. Step 1: In the first step, we read data from webcam. Read data file containing feature details of face. Search for face feature match in every frame. Crop the face region. Step 2: Crop the eye-region from the face region. Eye region also contain eyebrows. Step 3: Reduce eye region, take ROI of pupil area and take pixel data from ROI pupil. Step 4: Check the number of black pixel with predefined black pixel data, if the black pixel become smaller than the predefined black pixel value then the eyes are closed and if not then the eyes are opened. Step 5: If the minute limit is smaller than 'how long the eyes are closed' the system will show alert, and if the minute limit is greater than 'how long the eyes are closed' the system will find another sample frame from the webcam.

#### 5. Experimental Results and Analysis

For the analysis of the result, we divided this section into three categories (according to the methodology in Section 3): Tiredness detection based on (i) Thresholding, (ii) Angle of Head, and (iii) Distance from Eyelid We took 50 samples in lighting condition for detecting tiredness for every setup based on pixel computation.

**5.1 Thresholding** The overall performance was very satisfactory; we have not seen any error in face detection. However, after face detection, depending on the camera quality and lighting conditions sometimes the eye position cannot be found so getting centre position gives error. When the light source is behind the face the performance may degrade to some extends and may also detects multiple faces in the frame which creates some trouble. But most of the time it detects biggest face and work with it. In poor lighting condition the whole eye region may go dark. Here is





(a) open state

(b) close state





(c) open state

(d) close state

Figure 7: Various samples in night and day-light conditions.

Table 2: Experimental error in daylight condition.

| Category         | Samples | Errors | % of Errors |
|------------------|---------|--------|-------------|
| Light from Front | 50      | 5      | 10          |
| Light from Back  | 50      | 9      | 18          |
| Light from Side  | 50      | 7      | 14          |
| Dark skin        | 50      | 12     | 24          |
| Bright skin      | 50      | 6      | 12          |
| Gender- Boy      | 50      | 3      | 6           |
| Gender- Girl     | 50      | 4      | 8           |

the analysis of detected eye in different light conditions. We have taken some samples in night/indoor and some samples have taken in daylight conditions (see Fig. 7). We also have taken closed and open eye state samples. The accuracy is much better in lighting condition.

5.1.1 Experimenting in outdoor environment our experiment, we took the camera in different lighting conditions. At first we illuminated face of the person with light from the front, which is the most ideal case. We have got 5 errors out of 50 samples, which gave us 10 percentage of errors. Then we took the illumination from the back and side of the person. In this case, we have got 9 errors and 7 errors. Both gave us 18 and 14 percentage errors respectively. Our system is based on image threshold method so skin complexion is vital challenge. Hence we have tested our system for two different skin complexions. Dark skin complexion gave us 12 errors out of 50 which are 24% errors. For bright skinned persons we have got only 12 percents of errors which is 6 errors out of 50 samples. We also tested based on gender. The whole result is shown in the table (Table 2) and for comparative understanding chart diagram is provided (shown in Fig. 8).

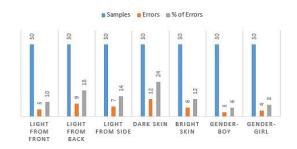


Figure 8: Graphical representation of error in daylight condition.

Table 3: Experimental error in indoor condition.

| Category         | Samples | Errors | % of Errors |
|------------------|---------|--------|-------------|
| Light from Front | 50      | 4      | 8           |
| Light from Back  | 50      | 12     | 24          |
| Light from Side  | 50      | 10     | 20          |
| Dark skin        | 50      | 16     | 32          |
| Bright skin      | 50      | 5      | 10          |
| Gender- Boy      | 50      | 5      | 10          |
| Gender- Girl     | 50      | 5      | 10          |

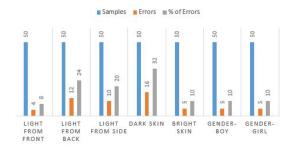
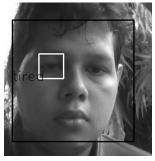


Figure 9: Graphical representation of error in indoor condition.

5.1.2 Experimenting in indoor environment For night simulation, we tested the system in indoor. As illumination level was very low, we had to use a table lamp for better reading. We took the camera in different lightening conditions. At first we illuminated face with light from the front, which is the most ideal case. We have got 4 errors out of 50 samples, which gave us 8 percentage of error. Then we took the illumination form the back and side of the person. In the first case, we have got 12 errors and for the side case we have got 10 errors. Both gave us 24 and 20 percentage of errors respectively. Persons with dark skin complexion gave us 16 errors out of 50 which is 32% error. For bright skinned persons, we have got only 10 percent of errors which is 5 errors out of 50. For both Boy and Girl, we have got 10 percent of errors. The whole result is shown in the table (in Table 3) and for comparative understanding chart diagram is provided (Fig. 9).

**5.2 Experiment on Angle of Head** For this approach, first we need to track eyes because we are dealing with only eyes, not head. When we move head we also move our eyes so if we track our eyes we can detect head angle. If our head angle became wider than predefined value











(c) Heavy Tired state

(d) Heavy Tired state

Figure 10: Tiredness detection based on distance (with 'tired' and 'heavy tired' notification).

Table 4: Experimental results of the 'Angle of head' at day-light/indoor condition.

#### At $(\theta > 3^{\circ})$ Condition:

| Category              | Samples | Errors | % of Errors |
|-----------------------|---------|--------|-------------|
| In Daylight condition | 50      | 9      | 18          |
| In Indoor condition   | 50      | 13     | 26          |

#### At $(\theta \gg 3^{\circ})$ Condition:

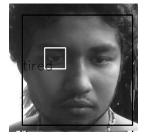
| Category              | Samples | Errors | % of Errors |
|-----------------------|---------|--------|-------------|
| In Daylight condition | 50      | 7      | 14          |
| In Indoor condition   | 50      | 12     | 24          |

then the alert notification will notify 'tired' (when  $\theta > 3^{\circ}$ ) or 'heavy tired' (when  $\theta \gg 3^{\circ}$ ) notification according to the angle of the head.

Let's define a predefined value ( $\theta = 3^{\circ}$ ) for the angle of the head of the driver. We need to divide the value into two parts if the angle of head exceeds the first part ( $\theta > 3^{\circ}$ ) then the alert notification will show only 'tired' notification. And if the angle of head exceeds second part ( $\theta \gg 3^{\circ}$ ) then the alert notification will show only 'heavy tired' notification (see Fig. 10).

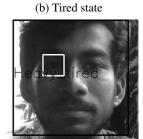
We took 50 samples in Daylight as well as Indoor condition for detecting tiredness. The result was enough satisfactory. This approach can detect (among 50 samples) almost 41 samples in daylight and 37 samples in indoor at the  $(\theta > 3^{\circ})$  condition and 43 samples in daylight as well as 38 in indoor at  $(\theta \gg 3^{\circ})$  condition. The errors and percentage of errors are listed in (Table 4).

**5.3** Experiment on Distance from Eyelid The main concern of this approach is the black pixel of the eye if the black pixel become smaller to smaller, the distance will be longer and longer, that is why the alert notification will notify 'tired' or 'heavy tired' notification. Let a predefined





(a) Tired state



(c) Heavy Tired state

(d) Heavy Tired state

Figure 11: Tiredness detection based on distance (with 'tired' and 'heavy tired' notification).

Table 5: Experimental results of 'distance from the eyelid' at daylight/indoor condition.

#### At (> 80) Condition:

| Category              | Samples | Errors | % of Errors |
|-----------------------|---------|--------|-------------|
| In Daylight condition | 50      | 9      | 18          |
| In Indoor condition   | 50      | 15     | 30          |

## At (≫ 80) Condition:

| Category              | Samples | Errors | % of Errors |
|-----------------------|---------|--------|-------------|
| In Daylight condition | 50      | 8      | 16          |
| In Indoor condition   | 50      | 11     | 22          |

value for the black pixel of the driver. We need to divide the value into two parts if the black pixel (in dpi) exceeds the first part (> 80) then the alert notification will show only 'tired' notification and if the black pixel (in dpi) exceeds second part (>> 80) from the eyelid then the alert notification will show only 'heavy tired' notification (see Fig. 11).

We took 50 samples in Daylight as well as Indoor condition for detecting tiredness. The result was enough satisfactory. This approach can detect (among 50 samples) almost 41 samples in daylight and 35 samples in indoor at the (>80) condition and 42 samples in daylight as well as 39 in indoor at (>80). The errors and percentage of errors are listed in (Table 5).

In this paper, we have implemented alarm alert system in our application by software, not on any hardware implementation. It works well though it has some error due to light conditions and bit execution processing time. It can be solved by using high configured processor on the computer while processing frame by frame. And we need to do more research work for getting better accuracy. We have developed a real time application for the purpose of safety-driving. One of the major reasons for vehicle accident is micro-sleep of a driver on highway. So we are trying to solve this problem by implementing microsleep alarm, alert system through this real time application. We took 50 sam-

ples in lighting condition for detecting tiredness based on pixel computation. The result can be improved in a better lighting condition with a better perform-able processing system. This kind of research can be applicable in future to understand the quality of workers in industries and offices. If some workers are found to be sleepy or tired at the end of the day, they may not get overtime. Countries like Japan, Korea as well as some European countries population are increasing very rapidly and hence, less young people to work for huge people. So selection of quality employees and extraction issues can be assessed through automatic analysis of tiredness of workers.

#### 6. Conclusion and Future Work

This paper presents a real time application on face detection and tiredness detection, which can be developed for a simple human-computer interaction system. It only requires a webcam with abundant light on face. So the objective of this paper is to develop a system to detect face and measure tiredness. In this paper, we produced a system to detect tiredness using three different methods based on thresholding, distance from the eyelid and angle of the head. The results were satisfactory in every method. In thresholding method the errors were quite reasonable and it was almost 6% to 8% while the error was almost 14% to 18% in the angle of head and distance from the eyelid respectively. We have done this experiment with 50 samples. Our algorithm performs well with little error, which can be acceptable considering the level of difficulties in computer vision. In future, we look forward to enhance performance by testing on different number of people in different lighting conditions. We may use cursor movement in future for general and disabled person to operate all mouse operation by eye movement and blink, driving safety for driver from micro-sleep and making life easier.

#### Acknowledgements

Authors are grateful to the subjects for their supports while doing the experiments. Authors acknowledge the supports of Center for Natural Science & Engineering Research (CNSER) for the support.

#### References

- [1] Q. Wang, J. Yang, M. Ren, and Y. Zheng, "Driver tiredness detection: A survey", *Proc. 6th World Congress on Intelligent Control and Automation (WCICA)*, vol. 2, pp. 8587-8591, 2006.
- [2] W. Horng, C. Chen, Y. Chang, and C.H. Fan., "Driver tiredness detection based on eye tracking and dynamic template matching", Proc. of the IEEE International Conference on Networking, Sensing & Control, pp. 7-12, 2004.
- [3] Md Atiqur Rahman Ahad, Computer Vision and Action Recognition: A Guide for Image Processing and Computer Vision Community for Action Understanding, ISBN: 978-94-91216-20-6, Springer, 2011.

- [4] Md Atiqur Rahman Ahad, Motion History Images for Action Recognition and Understanding, ISBN: 978-1-4471-4730-5, Springer, 2012.
- [5] Z. Zhang and J. Zhang, "A new real-time eye tracking based on nonlinear unscented Kalman filter for monitoring driver tiredness", Journal of Control Theory and Applications, 8(2), pp. 181-188, 2010. DOI 10.1007/s11768-010-8043-0
- [6] Md. Talal Bin Noman and Md Atiqur Rahman Ahad, "Mobile-based Eye-Blink Detection Performance Analysis on Android Platform", Frontiers ICT - Human-Media Interaction, 5(4), 2018. DOI: 10.3389/fict.2018.00004
- [7] U.H. Syeda, Ziaul Zafar, Zishan Zahidul Islam, Syed Mahir Tazwar, Miftahul Jannat Rasna, Koichi Kise, and Md Atiqur Rahman Ahad, "Visual face scanning and emotion perception analysis between Autistic and Typically Developing children", ACM UbiComp Workshop on Mental Health and Well-being: Sensing and Intervention, Hawaii, USA, 2017.
- [8] M.K. Galab, H.M. Abdalkader, and H.H. Zayed, "Adaptive Real Time Eye-Blink Detection System", *International Jour*nal of Computer Applications, 99(5), pp. 29-36, 2014.
- [9] Nafis Irtija, Mahsius Sami, and Md Atiqur Rahman Ahad, "Fatigue Detection Using Facial Landmarks", 4th Int. Symposium on Affective Science and Engineering, and the 29th Modern Artificial Intelligence and Cognitive Science Conference (ISASE-MAICS), 2018.
- [10] Mehmet Turkan, Montse Pardas, A. Enis Cetin, "Human eye localization using edge projections", Proc. of 2nd International Conference on Computer Vision Theory and Applications, 2007.
- [11] J. Wisniewska, M. Rezaei, and R. Klette, "Robust Eye Gaze Estimation", *Computer Vision and Graphics*, vol. 2, pp. 8587-859, 2006.
- [12] Y. Nishina, Md. Atiqur Rahman Ahad, JK Tan, H Kim and S Ishikawa, "A Robust Face Tracking Method Employing Color-based Particle Filter", Int. J. of Biomedical Soft Computing and Human Sciences, Vol. 16, No. 1, pp. 127-134, 2010.
- [13] F. P. Mahdi, M. M. Habib, Md Atiqur Rahman Ahad, S. Mckeever, ASM Moslehuddin, and P. Vasant, "Face Recognitionbased Real Time System for Surveillance", *Intelligent Deci*sion Technologies, IOS Press, 11(1), pp. 79-92, 2017.
- [14] Michael Chau and Margrit Betke, "Real Time Eye Tracking and Blink Detection with USB Cameras", Computer Science Department, Boston University, Boston, MA 02215, USA.
- [15] Paul Viola and Michael J. Jones, "Robust Real-Time Face Detection", *Proc. of 18th International Conference on Computer Vision*, 2, p. 747, 2001.
- [16] Tanmoy Paul, Ummul Afia Shammi, Mosabber Uddin Ahmed, Rashedur Rahman, Syoji Kobashi, and Md Atiqur Rahman Ahad, "A Study on Face Detection Using Viola-Jones Algorithm in Various Backgrounds, Angles and Distances", *Biomedical Soft Computing and Human Sciences*, 23(1), pp. 1-13, 2018.

- [17] Paul Viola and Michael J. Jones, "Robust Real-Time Face Detection", Proc. of 18th International Conference on Computer Vision, 57(2), pp. 137-154, 2004.
- [18] C.H. Morimoto and M. Flickner, "Real time multiple face detection using active illumination", 4th IEEE Int'l. Conf. on Automatic Face and Gesture Recognition (AFGR), pp. 8-13, 2000.
- [19] M. Betke, W. Mullally, and J. Magge, "Active detection of eye scleras in real time", IEEE CVPR Workshop on Human Modeling, Analysis and Synthesis, 2000.
- [20] H.M. Elahi, D. Islam, I. Ahmed, S. Kobashi and M.A.R. Ahad, 'Webcam-based Accurate Eye-central localization", Proc. of 2nd International Conference on Robot, Vision and Signal Processing, Japan, pp. 47-50, 2013.
- [21] A.A. Mohammed and S.A. Anwar, "Efficient Eye Blink Detection Method for disabled helping domain", *International Journal of Advanced Computer Science and Applications*, 5(5), pp. 202-206, 2014.
- [22] T. Morris, P. Blenkhron, and F. Zaidi, "Blink Detection for Real-Time Eye Tracking", *Journal of Network and Com*puter Applications, 25(22), pp. 129-143, 2002.
- [23] Zishan Zahidul Islam, Syed Mahir Tazwar, Md. Zahidul Islam, Seiichi Serikawa, and Md. Atiqur Rahman Ahad, "Automatic Fall Detection System of Unsupervised Elderly People Using Smartphone", Proc. of 5th IIAE International Conference on Intelligent Systems and Image Processing, 2017.
- [24] Md. Atiqur Rahman Ahad, "Gait analysis: an energy image-based approach", *International Journal of Intelligent Computing in Medical Sciences Image Processing, Taylor & Francis*, TSI Press, 5(1), pp. 81-91, 2013.
- [25] K. Grauman, M. Betke, J. Lombardi, J. Gips, and G.R. Bradski, "Communication via eye, blinks and eyebrow raises: Video-Based human-computer interfaces", *Universal Access Information Society*, 2(4), pp. 359-373, 2003.
- [26] J.J. Magee, M.R. Scott, B.N. Waber, and M. Betke, "Eyekeys: A real-time vision interface based on gaze detection from a low grade video camera", IEEE Workshop on Real-Time Vision for Human-Computer Interaction (RTV4HCI), 2004.
- [27] D.W. Hansen and Q. Ji, "In the eye of the beholder: A survey of models for eyes and gaze", *IEEE Trans. on PAMI*, 32(3), pp. 478-500, 2010.
- [28] https://www.youtube.com/watch?v=p4KXAMKvy1E, (Accessed in June. 2017).
- [29] A. Frigerio, P. Cavallari, M. frigeri, "Surface Electromyographic mapping of the orbicularis Oculi Muscle for Real-Time Blink Detection", *JAMA Facial Plast Surg*, 16(5), pp. 335-342, 2014.
- [30] https://www.youtube.com/watch?v=2q9DarPET0o, (Accessed in June. 2017).
- [31] https://www.youtube.com/watch?v=gAAXTbv-J8E, (Accessed in June. 2017).
- [32] http://docs.opencv.org/2.4/doc/tutorials/ imgproc/threshold/threshold.html.

- [33] F. Baronti, F. Lenzi, R. Roncella, and R. Saletti, "Distributed sensor for steering wheel grip force measurement in driver tiredness detection", *Proc. of Conference on Design, Au*tomation and Test in Europe, pp. 894-897, 2009.
- [34] T. Chieh, M. Mustafa, A. Hussain, E. Zahedi, and B. Majlis, "Driver tiredness detection using steering grip force", *IEEE Student Conference on Research and Development*, pp. 45-48, 2003.
- [35] M. Devi and P. Bajaj, "Driver tiredness detection based on eye tracking", Proc. of 1st International Conference on Emerging Trends in Engineering and Technology, pp. 649-652, 2008.
- [36] W. Wierwille, L. Ellsworth, S. Wreggit, R. Fairbanks, and C. Kirn, "Research on Vehicle-Based Driver Status/Performance Monitoring Development, Validation, and Refinement of Algorithms for Detection of Driver Drowsiness", Technical Report: DOT HS 808 247, National Highway Traffic Safety Administration, USA, Dec. 1994.