# *Android OpenCV Based Effective Driver Fatigue and Distraction Monitoring System*

R.Manoharan
Department of Computer Science and Engineering
Rajalakshmi Engineering College
Chennai, India
haran.mano24@gmail.com,
manoharan.r.2013.mecse@rajalakshmi.edu.in.

S.Chandrakala
Department of Computer Science and Engineering
Rajalakshmi Engineering College
Chennai, India
sckala@gmail.com,
chandrakala.s@rajalakshmi.edu.in.

*Abstract*— **Driver fatigue and distraction during travel are the major causes for the road accidents. Many driver monitoring systems have been proposed in recent years for monitoring driver activities to avoid accidents. Most of the existing systems are in the form of specialized embedded hardware, majorly present in luxurious vehicles. This paper presents an effective driver fatigue and distraction monitoring system for Android Automobiles. An intelligent system for monitoring driver fatigue and distraction during travel using Adaptive Template Matching and Adaptive Boosting is designed and implemented here. A novel approach of detecting eye rub due to irritation in eye and yawning detection through intensity sum of facial region is also proposed. Experiments are conducted using android OpenCV which can be installed in low cost smartphones as well as in Android Auto. Experiment results shows that a high accuracy of driver distraction is detected in different vehicles and camera locations.**

*Index Terms*— **Android OpenCV; Machine Learning; Driver Monitoring; Android Auto;**

## I. Introduction

Road safety is one of the main objectives in designing driver assistance systems. On average, every 30 s, one person dies somewhere in the world due to a car crash. The cost of accidents in the United States is estimated to be about $300 billion annually [1], i.e., about 2% of its gross domestic product. Conservative estimates suggest that a high proportion of fatalities and injuries due to traffic accidents involve impaired drivers. It is projected that these figures could be increased by 65% in the next 20 years, unless novel driving risk reduction methods are leveraged [2]. Among all fatal traffic accidents, 95% are caused by human errors [3]. The three major causes of these human errors, which are often referred to as the "Big Three," are alcohol, drowsiness, and inattention [4]. Statistics show that 25% of fatal accidents in Europe [5], 32% in the United States [6], and 38% in Canada [7] are caused by drunk drivers. The National Highway Traffic Safety Administration (NHTSA) reports that over 3000 fatalities from automobile accidents are caused by drowsiness or distraction [8].

Drowsiness is common during travel. Driver will either resist the drowsiness or stop the vehicle aside and continue the journey after a while. But sometimes driver may be unaware that they are actually feeling fatigue and it is the cause for major accidents. To avoid this, first driver fatigue has to be identified and alerted. Some external factors like dusts can cause irritation in eye and divert a driver during travel. Even an experienced driver can also be diverted from driving which eventually leads to an accident. So driver distractions during travel also have to be identified and alerted. Google designed an android based automobile operating system for cars which will be available in markets by middle of 2015. This operating system will be available in cars across 24 top and medium level car manufacturers in name of Android Auto. With a simple and intuitive interface, integrated steering wheel controls, and powerful new voice actions, it's designed to minimize distraction so driver can stay focused on the road [9]. So it is wise to propose a system which can be installed as software instead of an external hardware.

This paper presents an effective driver fatigue monitoring system for Android Automobiles. In this work various driver monitoring methods are explored for Android Automobiles. An intelligent system for monitoring driver distraction and fatigue during travel using adaptive template matching and adaptive boosting is designed and implemented. A novel approach of detecting eye rub due to irritation in eye and yawning through intensity sum of facial region is also proposed here. Experiments are conducted using low cost xiaomi redmi 1s smartphone powered by 1.6GHz processor with 1GB RAM. The application can be easily installed in Android Auto with small modifications. Experiment results shows that a high accuracy of driver distraction is detected in different scenarios with different vehicles and camera locations. Preprocessing of frames is done to avoid noise obtained during video acquisition.

## II. Related Work

Many driver monitoring systems have been proposed in recent years for monitoring driver activities to avoid accidents. Vision based intelligent system through smartphones offers rich potential for application development as well as research. Jaeik Jo et al [10] proposed four new methods to identify driver distraction and drowsiness. First driver drowsiness is identified. Second eye detection algorithm is proposed to avoid unclear image due to reflection in glasses. Third eye detection accuracy is enhanced by applying eye validation after initial detection of eye. Fourth a novel eye state detection algorithm is proposed. Though it has innovative approaches a laptop is required to process the frames.

Boon Giin Lee et al [11] proposed a method to monitor driver safety by analyzing information related to fatigue using eye movement monitoring and bio signal processing. It is implemented using a smartphone and sensors are embedded through bluetooth. But the processing time is slow because of the native camera feature. Boon Giin Lee enhanced the [11] and proposed a new system in [12] which combined various bio sensors for effective monitoring of driver distraction and fatigue. To indicate the driver capability level Fuzzy Bayesian framework is designed. But the Processing speed is slow because of the limited resource and driver alerting mechanism is not effective.

In [13] smartphone based monitoring of driver fatigue is proposed. Because of the limitations of android OpenCV and illumination conditions effective monitoring is not guaranteed. Estimating gaze direction of vehicle drivers using a smartphone camera is proposed in [14] which use SVM gaze classifier to identify where the driver is looking. An in-situ approach for gathering training data also proposed in this research work. But it is not implemented using the smartphone. Just the smartphone camera is used to capture the video and processed in PC.

## III. Proposed Work

An intelligent system for monitoring driver distraction and fatigue during travel using adaptive template matching and adaptive boosting is designed. A novel approach of detecting eye rub due to irritation in eye and yawning through intensity sum of facial region is also proposed.

### *A. System Design*

***Work Flow***

Figure 1 shows the complete work flow of the proposed Android OpenCV Based Driver Monitoring System. The driver fatigue and eye rub detection is carried out in several stages. Initially the Input video is split to frames. Image preprocessing (size reduction) is done to reduce the processing time on each frame. Haar – like feature detector is used to detect the face.

After the detection of face, eye and mouth are detected in parallel. Eyes are detected on the upper portion of the face and the mouth is detected on the lower portion of the face to increase the processing speed. Eyes are analyzed for eye rubs due to the irritation in eye caused by dust or other factors and drowsiness. If eye rub or drowsiness is found, then the driver is alerted with possible recorded voice suggestions. From the detected mouth, yawns are detected and counted. If number of yawns exceeds the threshold value then the driver is alerted with recorded voice.

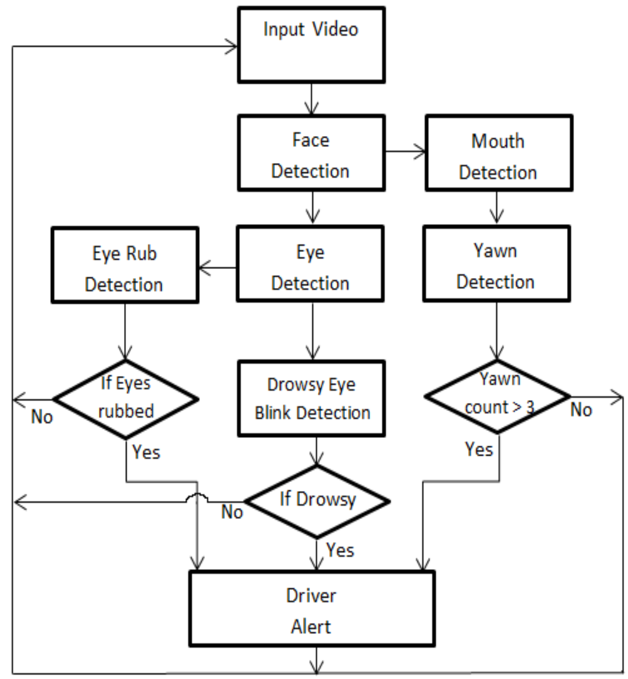


Fig. 1. Work Flow of Proposed Driver Monitoring System

Smartphone camera captures video at the resolution of 1280x720. The Video is recorded at the rate of 6frames per second. These frames resolutions are downsized to 320x180 and the color images are converted in to grey scale images to reduce the processing power, which in turn results in high speed processing.

***Face detection***

Haar Classifier Object detection is used to detect the face [15,16]. By combining adaptive boosting with adaptive template matching, Driver face is detected even during head rotation. Initially first five frames are used for training and further the face is detected based on those training frames. These training frames are adaptive and system is trained through these new frames which made face detection during head rotation possible.

***Eye detection***

Various algorithms are used for detecting the eye. In [13] Ada boost algorithm using the Haar – like feature detector is used to detect the eyes. But it fails when the driver head is rotated. So to overcome this issue, adaptive template matching and adaptive boosting are combined in this paper. In Figure *5, A, B* and *C* shows that eyes are detected even when the driver face is rotated. To improve efficiency of eye detection system, the upper portion of the detected face alone is used for searching the eye.

***Drowsy Blink Detection***

Eye blinks occurs mostly in a proper rhythm. First the normal blinks and the drowsiness blinks have to be differentiated. The average duration of eye closure is 400ms and the minimum duration is 75ms.Therefore if driver eyes experiences closure more than 400ms then it is considered as drowsiness. Change of black pixel in the eye region is used to detect the eye blink in this paper. As shown in the Figure *2*, open eye region *A* is converted in to corresponding binary image *B*. It shows that the larger visible pupil gives more black pixels. Closed eye region *C* is converted in to binary image *D*. Thus the open eye has more black pixels and the ratio of black pixels in *B* and *D* is used to detect the open eye and closed eye.

The number of black pixels in the eye region is calculated using the below formula,

$$R = \frac{B}{B+W} \text{ x } 100\%$$

where *R* is the Ratio of black pixels, *B* is the number of black pixels and W is the number of white pixels. If the value of *R* is lesser than 30% then it is considered as closed eye. The Percentage of Eye Closure (*PERCLOS*) is used to determine whether the driver is feeling fatigue or not. A fatigue driver may exhibit slow eye movements compared to normal state. *PERCLOS* is defined as the percent of time eye are closed in a short time window (often 30 seconds) [13]. An eye is considered as closed if the visible pupil is below 30% of its minimum opening

$$\text{PERCLOS} = \frac{Nblink}{30XS}$$

where *Nblink* denotes the number of eye blinks in the past 30 seconds and S is the sampling rate.

***Eye Rub detection***

Eye rubs or squeezes are one of the major reasons for accidents. Even a skilled driver is distracted by irritation in eye caused by external factors like dust. This reduces the concentration of the driver on road and may lead to an accident. The system proposed here is designed in a way to detect these kinds of distractions. When such distractions are found, the system will immediately alert the driver with some possible suggestions through voice alert. Eye rubs are detected by the difference in the black pixels of the eye regions

$$ERP = RR - RL$$

where *ERP* is the Eye Rub Percentage, *RR* is the Ratio of black pixels in Right eye and *RL* is the Ratio of black pixels in the Left eye. If the value of *ERP* is greater than 5% then it is considered as eye rub and driver will be alerted with recorded voice suggestions. In Figure *3*, the *A* and *C* show the various rub positions and *B* and *D* shows the corresponding binary image of it. Eyes are detected separately using Haar like feature. The sum of the black pixels in the left eye is lesser than the sum of the black pixels in the right eye. This will ring the alarm and provide suggestions to the driver through voice alert.

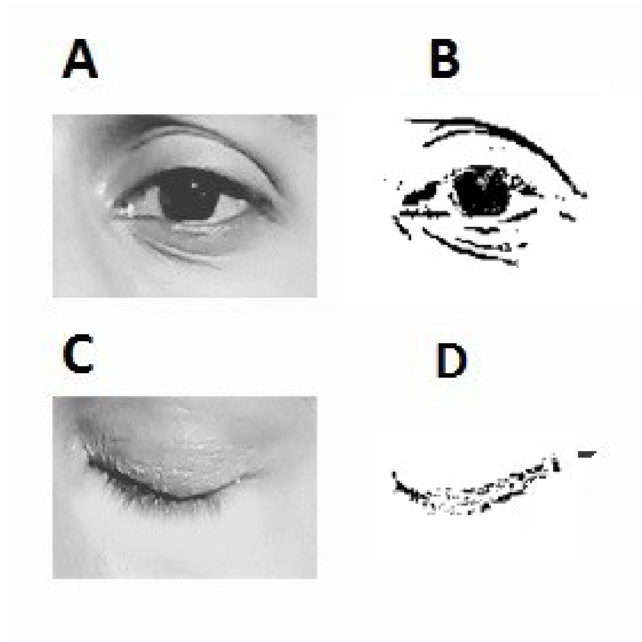


Fig. 2. Blink detection: (A) open eye region; (B) Binary image of (A); (C) Closed eye region; (D) Binary image of (C).

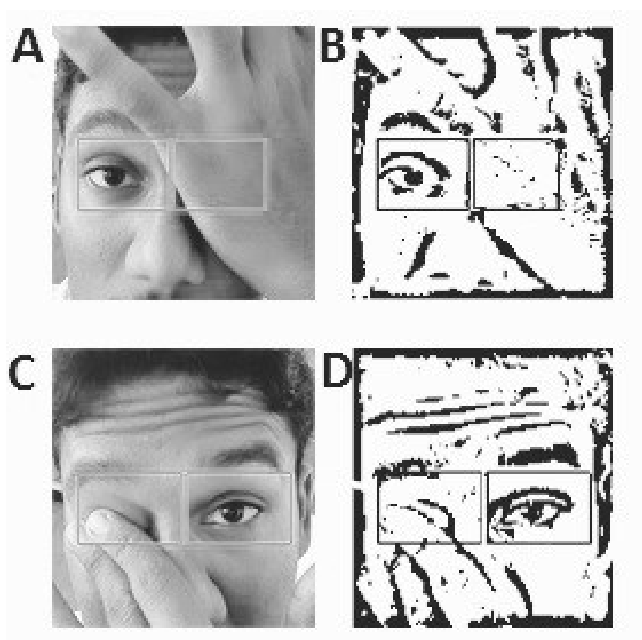


Fig. 3. Rub detection: (A), (B) Various Rub positions; (B) Binary image of (A); (D) Binary image of (C).

***Mouth Detection***

Haar like feature detector is used to detect the mouth. To increase the computational efficiency, the lower region of the face alone searched for mouth. Adaptive template matching and adaptive boosting is used to detect the mouth even when the face is rotated.

***Yawning Detection***

Yawning is not only a sign of tiredness but also a sign of changing body conditions. Repeated Yawning during travel is an indication of driver fatigue. Using the change of black pixels in mouth region yawning is detected. An open mouth is often darker than the closed mouth. Thus an open moth should have more black pixels compared to the closed.

In the detected mouth region, the image is converted into binary image as shown in the Figure *4*. Compared to the *B*, *D* has more black pixels which indicate the open mouth. It is important to not misclassify a partial open mouth as yawn. So a threshold value is fixed for Yawning and number of yawn count is logged. The Fatigue detection system will analyze the yawn counts and eye blinks to make the final decision. If the number of yawn count exceeds 3 then the driver is alerted with recorded voice.

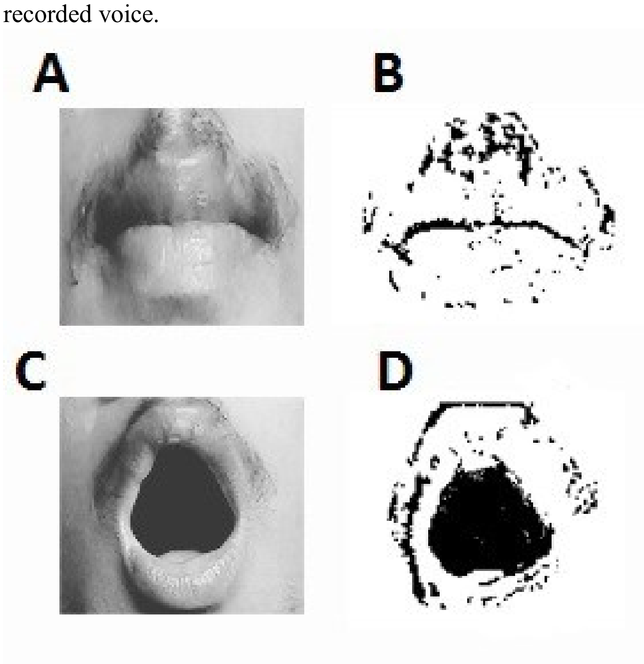


Fig. 4. Yawn detection: (A) Closed mouth region; (B) Binary image of (A); (C) Open mouth region; (D) Binary image of (C).

## IV. Experimentation Result

Experiments are conducted using a low cost xiaomi redmi 1s smartphone powered by 1.6GHz processor with 1GB RAM. It runs on top of the Android 4.3 Jelly Bean version. *E* and *F* in Figure 5 shows the optimal positions in which the monitoring is effective compared to all other positions. *F* view is positioned from the sun shield in the car. This view will be used for further processing in this research work.


Fig. 5. Various camera positions for monitoring the driver face

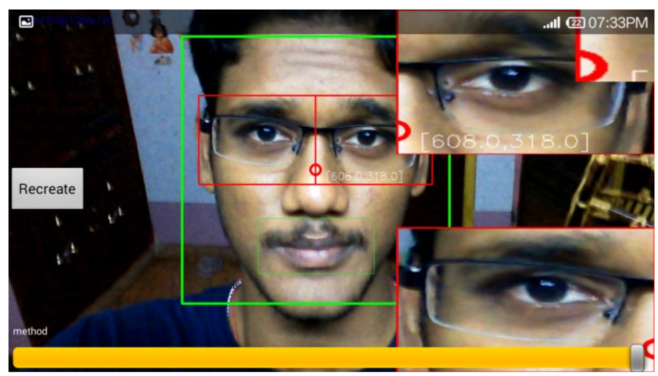


Fig. 6. Screen shot of Android OpenCV based effective driver fatigue and distraction monitoring system in normal mode.

Figure *6* shows the Screen shot of Android OpenCV Based Effective driver Fatigue monitoring system in normal mode. Driver eyes and mouth are monitored continuously for detecting any abnormality (fatigue). Figure *7* shows the Screen shot of Android OpenCV based effective fatigue monitoring system in alert mode. System gives continuous alert when any abnormalities are detected. This proposed drive fatigue system can be easily installed in Android Auto with small modifications. Performance of the system is measured based on the CPU load and the battery consumption.

The effective design of this system placed very low load on cpu and consumed only 12% of battery use of continuous use for 1hour.

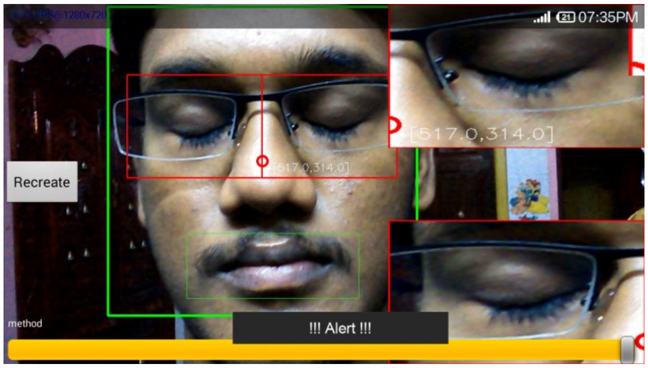


Fig. 7. Screen shot of Android OpenCV based effective driver fatigue and distraction monitoring system in alert mode.

## V. Conclusion and Future Work

To reduce the accidents due to the driver fatigue, an effective non-intrusive driver fatigue monitoring system is proposed in this paper. This fatigue monitoring system offers easy portability from Android mobile to Android Auto with few modifications. This paper is an initial work of the driver monitoring system for Android Auto. In this paper no IR illuminators are used which in turn will not provide best results under low/no light conditions. In near future, extraction of features from face as well as the body posture during driving with IR illuminators will be addressed.

## References

[1] "Crashes vs. congestion, what is the cost to the society?" American Automobile Association (AAA), Heathrow, FL, USA, 2011.

[2] A. Doshi andM.M. Trivedi, "On the roles of eye gaze and head dynamics in predicting driver's intent to change lanes," IEEE Trans. Intell. Transp.Syst., vol. 10, no. 3, pp. 453–465, Sep. 2009.

[3] A. Amditis, M. Bimpas, G. Thomaidis,M. Tsogas, M. Netto, S.Mammar, A. Beutner, N. Möhler, T.Wirthgen, S. Zipser, A. Etemad,M. Da Lio, and R. Cicilloni, "A situation-adaptive lane-keeping support system: Overview of the SAFELANE approach," IEEE Trans. Intell. Transp. Syst., vol. 11, no. 3, pp. 617–629, Sep. 2010.

[4] K. Murata, E. Fujita, S. Kojima, S. Maeda, Y. Ogura, T. Kamei, T. Tsuji, S. Kaneko, M. Yoshizumi, and N. Suzuki, "Noninvasive biological sensor system for detection of drunk driving," IEEE Trans. Inf. Technol. Biomed., vol. 15, no. 1, pp. 19–25, Jan. 2011.

[5] L.-W. Zhu, Z.-Y. Zhang, Z.-J. Bao, and Y. Sun, "Study on the drink driving behavior of drivers in Beijing based on the theory of plan behavior," in Proc. LEITS, 2010, pp. 1–5.

[6] D. Jiangpeng, T. Jin, B. Xiaole, S. Zhaohui, and X. Dong, "Mobile phone based drunk driving detection," in 4th Int. Conf. PervasiveHealth, Munich, Germany, 2010, pp. 1–8.

[7] Y. C. Liu and C. H. Ho, "The effects of different breath alcohol concentration and post alcohol upon drivers driving performance," in Proc. IEEE IEEM, 2007, pp. 505–509.

[8] "Bludeprint for ending distracted driving", NHTSA Report, 2012. Available: http://www.distraction.gov

[9] Available: http://www.android.com/auto/

[10] Jaeik Jo, Ho Gi Jung, Kang Ryoung, Jaihie Kim, "Vision – based method for detecting driver drowsiness and distraction in driver monitoring system" Optical Engineering SPIE Vol 50 Dec 2011.

[11] Boon – Giin Lee and Wan – Young Chung, "Driver alertness monitoring using fusion of facial features and bio signals" IEEE SENSORS JOURNAL Vol. 12 july 2012.

[12] Boon – Ginn Lee and Wan – Young Chung, " A smart phone based driver safety monitoring system using data fusion" Sensors journal, Vol 12, 2012.

[13] He J, Roberson S, Fields B, Peng J, Cielocha S, et al (2013) Fatigue detection using smartphones J Ergonomics 3: p.p 120.

[14] Meng – che chuang, Raja bala, Edgar A. Bernal, Peter Paul, Aaron Burry, "Estimating Gaze Direction of Vehicle Drivers using a Smartphone Camera" CVPR workshop paper 2014.

[15] Viola P, Jones M(2004) Robust real-time face detection. Int J Computer Vision 57: 137-154.

[16] Viola P, Jones M, Snow D (2003) Detecting pedestrians using patterns of motion and appearance. 9th IEEE international conference on computer vision (ICCV 2003), Nice, France: 734-741.