

# **DRIVER DROWSINESS DETECTION MINI REPORT 6**

**BASU VERMA**  
(142002007)

*under the guidance of*

**Dr Satyajit Das**



---

**IIT PALAKKAD**

**INDIAN INSTITUTE OF TECHNOLOGY PALAKKAD  
PALAKKAD - 678557, KERALA**

---

## *Contents*

---

<b>List of Figures</b>	<b>ii</b>
<b>1 Eye Detection and classification</b>	<b>1</b>
1.0.1 Drowsiness Detection . . . . .	1
1.0.2 Implementation . . . . .	1
1.0.3 Conclusion . . . . .	3
<b>References</b>	<b>4</b>

---

## *List of Figures*

---

1.1	Detection of eye in drowsy data set. . . . .	1
1.2	Open-eye detection is drowzy dataset. . . . .	2
1.3	Open eye detection and classification . . . . .	2
1.4	Closed eye detection and classification . . . . .	2
1.5	One-eye closed and another eye open classification . . . . .	3

## *Part. 1*

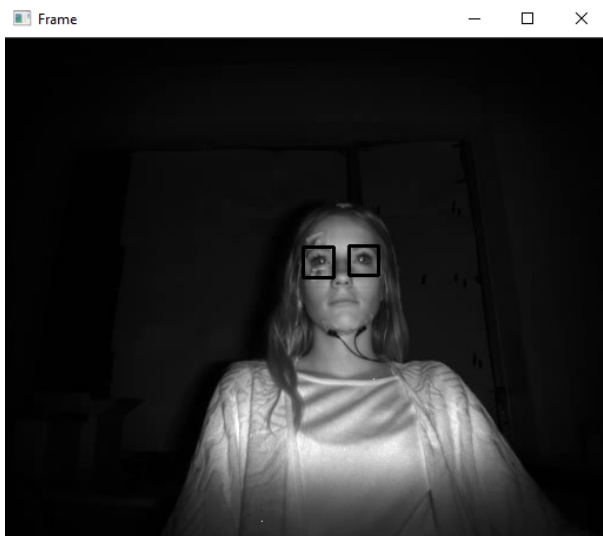
---

### *Eye Detection and classification*

---

#### 1.0.1 Drowsiness Detection

Detection of drowsy state of a person from desktop web cam. Haar Cascade classifier is used for eye detection in the video and CNN model is used for the classification of eye state as open or close. The video frame rate is 30fps.

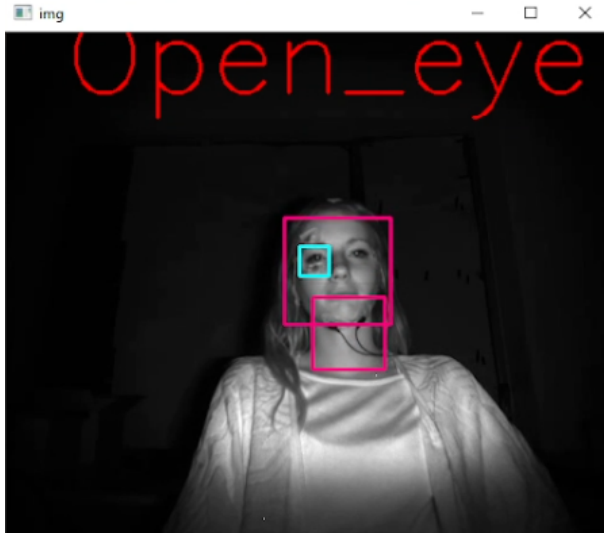


**Fig. 1.1** Detection of eye in drowsy data set.

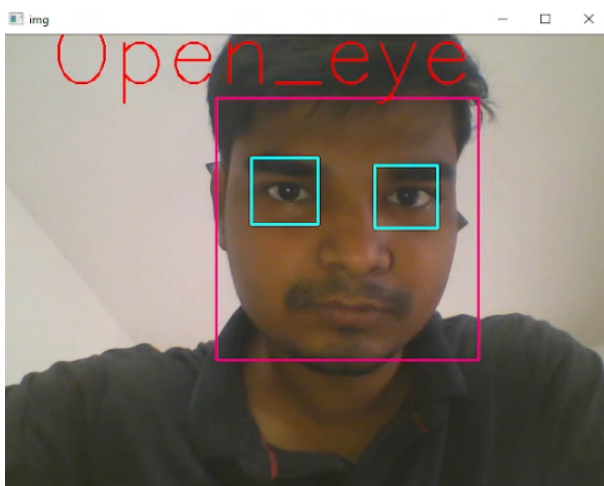
#### 1.0.2 Implementation

For implementation of the eye detection using Haar-like features [1], I used "**haar-cascade\_eye.xml**" file and read the drowsy videos frames continuously (30fps) and video from web cam (30fps). Converted the frame to gray scale and applied pretrained eye\_cascade model on it to detect the eye. The following result was recorded.

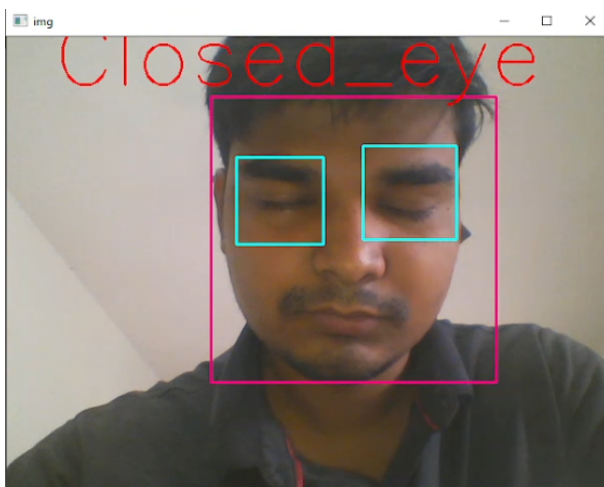
After detection of eye in the video, the eye area is cutt and fed to the CNN model(using VGG16 pretrained model), the CNN model is making the prediction whether the state is open or close.[2]



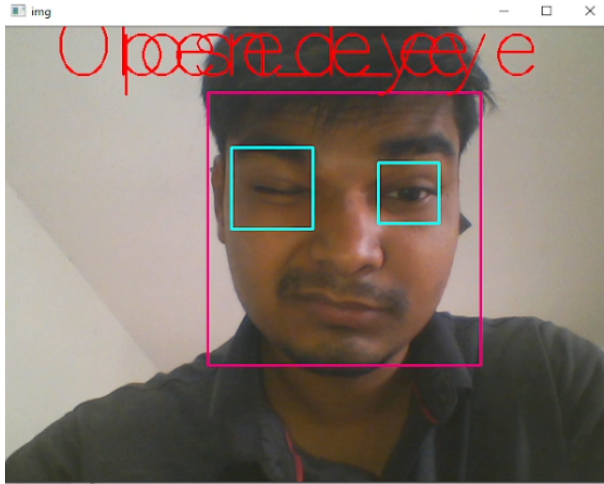
**Fig. 1.2** Open-eye detection is drowzy dataset.



**Fig. 1.3** Open eye detection and classification



**Fig. 1.4** Closed eye detection and classification



**Fig. 1.5** One-eye closed and another eye open classification

### 1.0.3 Conclusion

In drowzy dataset (the video is 30 fps), above model is successfully detecting the eye portion in the video when the eye state is open and it is also successfully classifying the eye state into open or close state but it is not detecting eye portion when the eye state is closed.

For the video from desktop web cam (this video is also 30fps), above model is successfully detecting the eye portion in the video whether it is open or close and also classifying it into proper category of open or close eye state with the condition that the room should have proper lighting and illumination.

---

## *References*

---

- [1] J. Zhu and Z. Chen, “Real time face detection system using adaboost and haar-like features,” in *2015 2nd International Conference on Information Science and Control Engineering*, 2015, pp. 404–407.
- [2] V. Chirra, U. S. Reddy, and V. KishoreKolli, “Deep cnn: A machine learning approach for driver drowsiness detection based on eye state,” *Revue d’Intelligence Artificielle*, vol. 33, pp. 461–466, 12 2019.