

# Driver Drowsiness Detection using Machine Learning Approach

Priyanka Basavaraj Murdeshwar

Department of Computer Science and Engineering  
Alva's Institute Of Engineering and Technology  
Mijar, Moodbidri, India

Surekha Reddy

Department of Computer Science and Engineering  
Alva's Institute Of Engineering and Technology Mijar,  
Moodbidri, India

Dr. Manjunath Kotari

Department of Computer Science and Engineering  
Alva's Institute Of Engineering and Technology  
Mijar, Moodbidri, India

Shruthi Tharanath Salian

Department of Computer Science and Engineering  
Alva's Institute Of Engineering and Technology  
Mijar, Moodbidri, India

Sharath D S

Department of Computer Science and Engineering  
Alva's Institute Of Engineering and Technology  
Mijar, Moodbidri, India

**Abstract-**Drowsiness of drivers is one of the significant cause of road accidents. Every year, there is an increase in the amount of deaths and fatal injuries globally. By detecting the driver's drowsiness, road accidents can be reduced. This paper describes a machine learning approach for drowsiness detection. Face detection is employed to locate the regions of the driver's eyes, which are used as the templates for eye tracking in subsequent frames. Finally, the tracked eye's images are used for drowsiness detection in order to generate warning alarms. This proposed approach has three stages: detecting Face, detecting Eyes and detecting drowsiness. Image processing is used to recognize the face of the driver and then its extracts the image of the eyes of the driver for detection of drowsiness. The HAAR face detection algorithm takes as captured frames of image and then the detected face is considered as output. Next, CHT is used for tracking eyes from the detected face. Using EAR (Eye Aspect Ratio) the eye state is detected. The proposed system was tested by implementing the proposed approach on a Raspberry pi 3 Model B with 1GB RAM with use of Logitech HD Webcam C270. The system uses frames for face and eye tracking, and the average correct rate for eye location and tracking could achieve 95.0% based on some test videos. Thus, the proposed approach for a real-time of driver drowsiness detection is a low cost and effective solution method.

**Keywords:** Haar Face detection, AdaBoost, EAR (Eye Aspect Ratio), Raspberry pi3

## I. INTRODUCTION

Driver's drowsiness is one of the main reasons for road accidents. If the drowsiness of the driver can be predicted at initial stages, and if the driver can be alerted of the same then a number of accidents can be reduced. In this paper, a driver drowsiness detection system using machine learning approach is proposed. Frontal face detection and eye detection respectively are detected in the initial stage. There are three main contributions of the object detection framework have been used to achieve high frame rates. The first contribution of this paper, is a new image representation called an integral image that allows for very

fast feature evaluation. It use a set of features which are reminiscent of Haar Basis functions In order to compute these features very rapidly at many scales, the integral image representation for images is introduced. Using a few operations per pixel, the integral image can be computed from an image.

The second contribution is a method for constructing a classifier by selecting a small number of important features using AdaBoost. In order to ensure rapid classification, the learning process must exclude a large majority of the available features, and focus on a small set of essential features. This feature selection is achieved through a simple modification of the AdaBoost procedure.

The third contribution of this paper is a method for combining successively more complex classifiers in a cascade structure which dramatically rapids the speed of the detector by focusing attention on promising regions of the image. This object detection procedure classifies images based on the value of simple features. More specifically it is based on three kinds of features. A three-rectangle feature computes the sum within two outside rectangles subtracted from the sum in a center rectangle. Finally a four-rectangle feature computes the difference between diagonal pairs of rectangles.

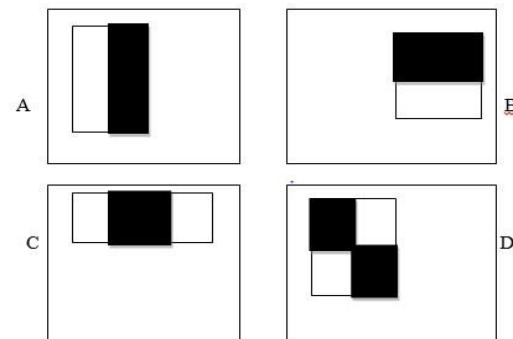


Figure 1.1 Examples for HAAR

As shown in figure 1.1 rectangle features relative to the enclosing detection window. The sums of the pixels that lie within the white rectangles are subtracted from the sum of pixels in the grey rectangles.

#### A. Integral Image

Rectangle features are computed rapidly using an intermediate representation for the image. The integral image at location  $z, y$  contains the sum of the pixels above and to the left of  $z, y$ , inclusive:  $ii(x, y) = \sum_{x' < x, y' < y} i(x', y')$  where  $ii(z, y)$  is the integral image and  $i(z, y)$  is the original image.

Using the following pair of recurrences:

$$s(x, y) = s(x, y - 1) + i(x, y) \quad (1) \\ ii(x, y) = ii(x - 1, y) + s(x, y) \quad (2)$$

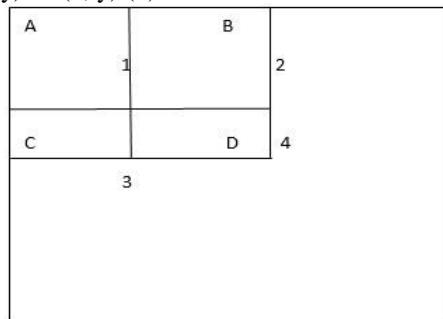


Figure 1.2 Pixels

(where  $s(z, y)$  is the cumulative row sum,  $s(z, 1) = 0$ , and  $ii(-1, y) = 0$ ) the integral image can be computed in one pass over the original image.

#### B. Learning Classification Functions

Given a feature set and a training set of positive and negative images, any number of machine learning approaches could be used to learn a classification function. In this system a variant of AdaBoost is used both to select a small set of features and train the classifier.

#### ALGORITHM 1: AdaBoost algorithm

- Given example images  $(x_1, y_1), \dots, (x_n, y_n)$  where  $y_i = 0, 1$  for negative and positive examples respectively.
- Initialize weights  $w_{1,i} = 1/m, 1/l$  for  $y_i = 0, 1$  respectively, where  $m, l$  are the number of negative and positive respectively.
- For  $t = 1, \dots, T$ :
  - Normalize the weights,  $W_t, i \leftarrow W_t, i / \sum_{j=1}^n W_t, j$   
So that  $w_t$  is the probability distribution.
  - For each feature,  $j$ , train the classifier  $h_j$  which is restricted to using single feature. The error is evaluated with respect to  $w_t$ ,  $e_j = \sum_i w_i |h_j(x_i) - y_j|$
  - Choose the classifier,  $h_t$ , with the lowest  $e_t$ .
  - Update the weights,  
 $W_{t+1, i} = W_t, i \beta_t^{1-e_t}$   
Where  $e_t = 0$  if example is classified correctly,  $e_t = 1$ , otherwise and  $\beta_t = e_t / (1 - e_t)$ . The final strong classifier is,

$$h(x) = \{1, \sum_{t=1}^T \frac{1}{2} \sum_{t=1}^T a_t \} a_t, h_t(x) \geq \sum_{t=1}^T a_t$$

0, otherwise  
where  $a_t = 1/\beta_t$

The AdaBoost algorithm for classifier learning is demonstrated. Each round of boosting selects one feature from the 180,000 potential features.

#### Objectives

- To detect the level of fatigue of the driver and alerting them.
- To detect the alcohol intoxication level of the driver

## II. PROBLEM STATEMENT

There are various reasons for accidents on road. The major reasons for accidents on road are fatigueness of driver.

## III. METHODOLOGY

#### A. Overview Design

The main concept of Driver Drowsiness Detection is to capture a driver's eyes from a camera and be able to accurately calculate the level of drowsiness in drivers with real-time processing. To achieve these requirements, proper materials have to be selected. For the base computer, the Raspberry Pi 3 Model B is selected.

#### B. System Design



Figure 1.3 Raspberry Pi 3 Model B

#### C. Software

Raspberry Pi 3 Model B and Raspberry Pi Camera is used because of the high performance of its CPU and higher frame rate. Raspberry Pi 3 Model B supports C++ and OpenCV library. OpenCV Version 3.1.0 is applied for various features of computer vision. The HAAR Cascade Classifier, warp Affine and template matching are supported in OpenCV library.

#### ALGORITHM 2:

#### DROWSINESS DETECTION ALGORITHM

The general flow of our drowsiness detection algorithm is fairly straightforward.

- First, a camera is setup that monitors a stream for faces.
- If a face is found, facial landmark detection is applied to extract the eye regions.

3. The eye regions are used to compute the eye aspect ratio to determine if the eyes are closed.
4. If the eye aspect ratio indicates that the eyes have been closed for a sufficiently long enough amount of time, sound an alarm is used to wake up the driver and the indicators will be on to indicate the pillion drivers.

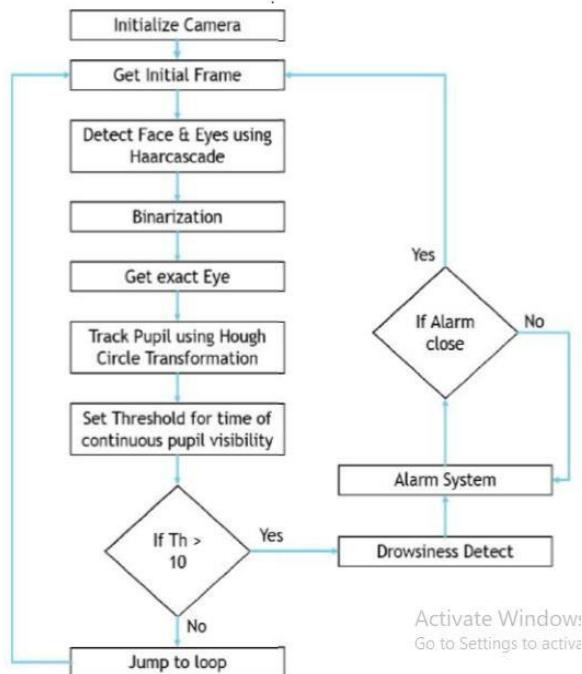


Figure 1.4 Algorithm flowchart

### C. Face Detection

The proposed system will start by capturing the video frames one by one. The system will detect the face in the frame image for each frame. This is achieved by making use of the HAAR algorithm for face detection. Firstly load the cascade file and then pass the acquired frame to an edge detection function, which detects all the possible objects of different sizes in the frame. Instead of detecting objects of all possible sizes, specify the edge detector to detect only objects of a particular size i.e. for face region. The output of this module is a frame with face detected in it.

#### 1. HAAR Cascade Classifier

It is one of the first tools utilized to detect a driver's face. It is one of the few object detection methods with the ability to detect faces. Paul Viola and Michael Jones invented this method. It is trained with thousands of faces in different light conditions. Ten people were tested with the HAAR Cascade Classifier

#### 2. Region of Interest

Region of interest (ROI) can detect a driver's face with increased accuracy. The formula for variable temp can be written as:

$$T = (100 - |FC/2|)/100 \text{ where } T \text{ is the temp value and } FC \text{ is the face degree.}$$

#### D. Eye Detection

After the face detection function has detected the face, the eyes detection function tries to detect the automobile driver's eyes. After face detection, find eye region by

considering eyes are present only in upper part of the face and from top edge of the face, extract eyes Region Of Interest (ROI) by cropping mouth and hair, mark it the region of interest. Speed up the processing for getting exact eyes.



Figure 1.5 Eye detection result

### E. Eye State and Drowsiness Detection

To identify drowsiness, it is first necessary to detect whether the eyes are close or open through the time. Eye Aspect Ratio (EAR) is used to detect whether the eyes of the driver is open or close. To calculate EAR, Euclidean formula is used to calculate the distance between the key points on the eyes.

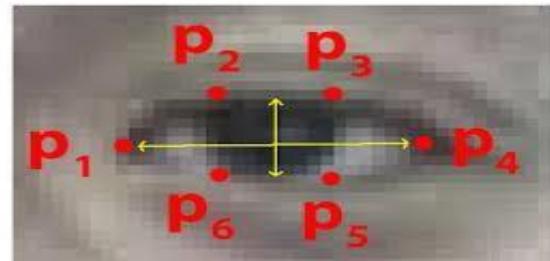


Figure 1.6 EAR Key points

The equation that reflects this relation called the Eye Aspect Ratio (EAR), which is derived from:

$$EAR = \frac{\|p2-p6\| + \|p3-p5\|}{2\|p1-p4\|}$$

The numerator of this equation computes the distance between the vertical eye landmarks while the denominator computes the distance between horizontal eye landmarks. In this proposed model, the threshold value is set to 300. If EAR is below 300 then the driver is detected as drowsy.

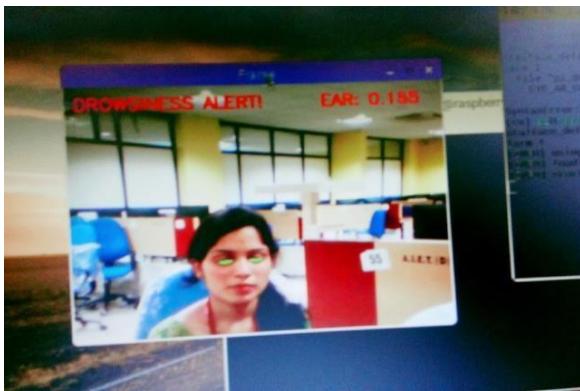


Figure 1.7 Drowsiness detected

#### IV. CONCLUSION

Driver Drowsiness Detection was built to help a driver stay awake while driving in order to reduce car accidents caused by drowsiness. This paper experimented in a bright room with constant light. In addition, there were several limitations including light conditions and the darkness of the skin. The embedded device can calculate a drowsiness level from the driver using a combination of Raspberry Pi 3 Model B and Raspberry Pi Camera. Raspberry Pi 3 Model B is a processor to calculate whether or not a driver is drowsy.

#### REFERENCES

- [1] Tejasweeni Musaleand Pansambal, DRIVER DROWSINESS DETECTION TECHNIQUE USING RASPBERRY PI , B.H. Bhavarabai Sawant College of Engineering and Research, Narhe, Pune, India ARTICLE.
- [2] Oraan Khunpisuth, Taweechai Chotchinasri, "DRIVER DROWSINESS DETECTION USING EYE-CLOSENESS DETECTION", Varakorn Koschakosai Mahidol University International College, Department of Computer Engineering, Faculty of Engineering, Mahidol University, NakhonPathom, Thailand Narit Hnoohom Image, Information and Intelligence Laboratory, Department of Computer Engineering, Faculty of Engineering, Mahidol University, Nakhon Pathom, Thailand
- [3] Marco Javier Flores, José María Armingol and Arturo de la Escalera, "REAL-TIME DROWSINESS DETECTION SYSTEM FOR AN INTELLIGENT VEHICLE" Member, IEEE
- [4] Belal ALSHAQAQI; Abdullah Salem BAQUHAIZEL, Mohamed El Amine OUIS, Meriem BOUMEHED; Abdelaziz OUAMRI, "DRIVER DROWSINESS DETECTION SYSTEM", Mokhtar KECHE, Laboratory signals and images (LSI) University of Sciences and Technology of Oran Mohamed Boudiaf (USTO-MB) Oran, Algeria {alshaqaqi@ieee.org}; Li Cuimei, Qi Zhiliang, Jia Nan, Wu Jianhua "HUMAN FACE DETECTION ALGORITHM VIA HAAR CASCADE CLASSIFIERS COMBINED WITH THREE ADDITIONAL CLASSIFIERS" @ IEEE 13<sup>th</sup> International Conference 1.School of Communication and Electronics, Jiangxi Science & Technology Normal University Nanchang 330013, China2. School of Information Engineering Nanchang University Nanchang 330031, China
- [5] T.D'Orazio, M. Leo, G.Cicirelli, "AN ALGORITHM FOR REAL TIME EYE DETECTION IN FACE IMAGES" A. Distante[dorazio, leo, cicirelli, distante] @ba.issia.cnr.it Institute of Intelligent Systems for Automation.
- [6] Viola P, Jones M J., "RAPID OBJECT DETECTION USING A BOOSTED CASCADE OF SIMPLE FEATURES", Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (ICCVPR 2001), vol. 1, pp. I-511-I-518, 8-14 Dec. 2001, Kauai, USA.