



Driver Drowsiness Monitoring System Using Visual Behaviour and Machine Learning

¹Jayashree J, ²Nisarga M G, ³Ranjitha N, ⁴Supritha A V, ⁵Mr. Mohan Kumar K S

¹⁻⁴Student, Electronics and Communication Department, B G S Institute of Technology, Mandya, India

⁵Asst. Prof, Dept. of ECE, Electronics and Communication Department, B G S Institute of Technology, Mandya, India

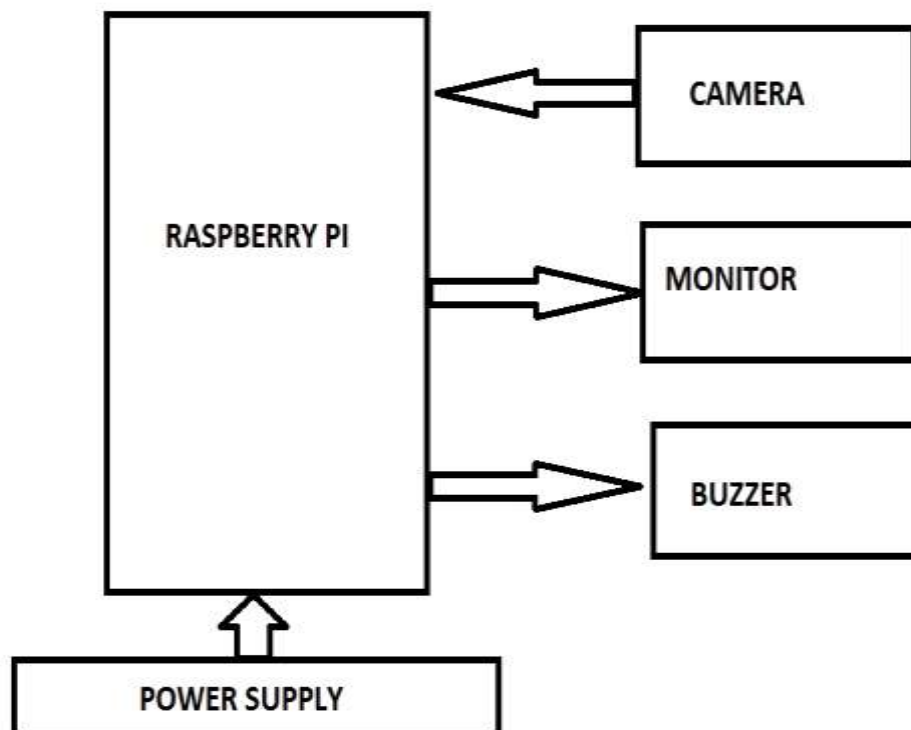
ABSTRACT: A countless number of people drive on the highway day and night. Taxi drivers, bus drivers, truck drivers and people traveling long-distance suffer from lack of sleep. Due to which it becomes very dangerous to drive when feeling sleepy. Drowsiness detecting device is solution for avoiding accidents.

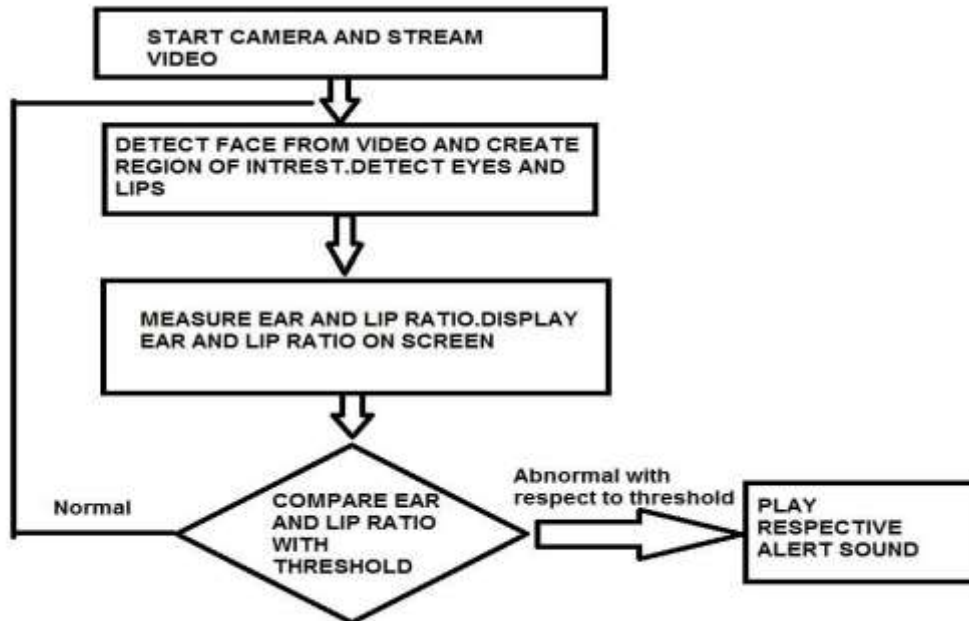
Key word: Drowsiness Detecting, Machine Learning.

I. INTRODUCTION

The majority of accidents happen due to the drowsiness of the driver. So, to prevent these accidents we will build a system using Python, OpenCV, and Keras which will alert the driver when he feels sleepy. Drowsiness detection is a safety technology that can prevent accidents that are caused by drivers who fell asleep while driving. The objective of this project is to build a drowsiness detection system that will detect that a person's eyes are closed for a few seconds. This system will alert the driver when drowsiness is detected. The requirement for this Python project is a webcam through which we will capture images. You need to have Python (3.6 version recommended) installed on your system, then using pip, you can install the necessary packages.

II. METHODOLOGY



**Flow Diagram:****Step 1 – Start camera and Stream video**

With a webcam, we will take video as input. So to access the webcam, we made an infinite loop that will capture each frame. We use the method provided by OpenCV, `cv2.VideoCapture(0)` to access the camera and set the capture object (`cap`). `cap.read()` will read each frame and we store the image in a frame variable

Step 2 – Detect the face from video and create a Region of Interest (ROI). Detect eyes and Lips.

To detect the face from video, we need to first convert the image into grayscale as the OpenCV algorithm for object detection takes gray images in the input. We don't need color information to detect the objects. We will be using haar cascade classifier to detect faces. This line is used to set our classifier `face = cv2.CascadeClassifier('path to our haar cascade xml file')`. Then we perform the detection using `faces = face.detectMultiScale(gray)`. It returns an array of detections with x,y coordinates, and height, the width of the boundary box of the object. Now we can iterate over the faces and draw boundary boxes for each face.

Step 3 – Measure Ear and Lip Ratio. Display Ear and Lip Ratio on Screen.

The same procedure to detect faces is used to detect eyes and lip. First, we set the cascade classifier for eyes in `leye` and `reye` respectively then detect the eyes using `left_eye = leye.detectMultiScale(gray)`. Now we need to extract only the eyes and lip data from the full image. This can be achieved by extracting the boundary box of the eye and lip and then we can pull out the eye and lip image from the frame with this code. `l_eye` only contains the image data of the eye. This will be fed into our CNN classifier which will predict if eyes are open or closed. Similarly, we will be extracting the right eye into `r_eye`.

Step 4 – Compare Ear and Lip Ratio with Threshold.

We are using CNN classifier for predicting the eye status. To feed our image into the model, we need to perform certain operations because the model needs the correct dimensions to start with. The score is basically a value we will use to determine how long the person has closed his eyes and lip. So if both eyes are closed, we will keep on increasing score and when eyes are open, we decrease the score. We are drawing the result on the screen using `cv2.putText()` function which will display real time status of the person. When the person is drowsy, he tends to yawn. By constantly measuring the distance between upper and lower lip, if person yawns the distance between upper and lower lip increases. by when the distance is greater than threshold.

Step 5 – Play Respective alert sound.

A threshold is defined for example if score becomes greater than 15 that means the person's eyes are closed for a long period of time then it play alert sound.



III. RESULT AND ANALYSIS



Figure 1 : Ear and Yawn Ratio



Figure 2 ; Yawn Alert

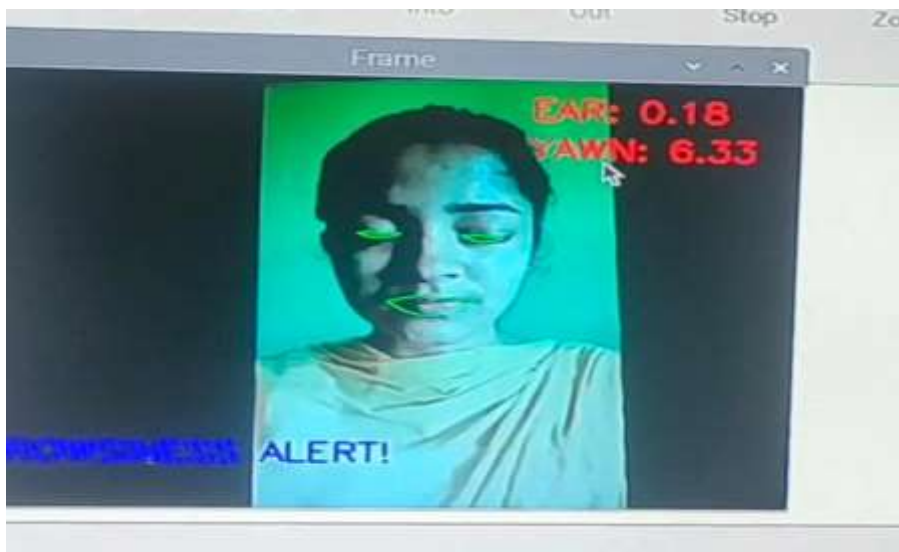


Figure 3 : Drowsiness Alert



IV. CONCLUSION

Our project is cost effective solution for preventing accident due to drowsiness. The device is affordable hence majority of the people can afford to purchase. The device is easy to manufacture and assemble. The device has potential to save many lives. In this paper, a low cost, real time driver drowsiness monitoring system has been proposed based on visual behavior and machine learning. Here, visual behavior features like eye aspect ratio, mouth opening ratio and nose length ratio are computed from the streaming video, captured by a webcam. An adaptive thresholding technique has been developed to detect driver drowsiness in real time.

REFERENCES

- [1] WHO (2009). Global Status Report on Road Safety Geneva, Switzerland. https://afro.who.int/sites/default/files/2017-06/vid_global_status_report_en.pdf [PubMed]
- [2] Rau P. (2005). Drowsy Driver Detection and Warning System for Commercial Vehicle Drivers: Field Operational Test Design, Analysis, and Progress. National Highway Traffic Safety Administration; Washington, DC, USA. [Google Scholar]
- [3] Drivers Beware Getting Enough Sleep Can Save Your Life This Memorial Day. National Sleep Foundation (NSF); Arlington, VA, USA: 2010. [Google Scholar]
- [4] Husar P. (2012). Eyetracker Warns against Momentary Driver Drowsiness. <http://www.fraunhofer.de/en/press/research-news/2010/10/eye-tracker-driver-drowsiness.html>
- [5] Poursadeghiyan M, Mazloumi A, Saraji GN, et al. (2017). Determination the levels of subjective and observer rating of drowsiness and their associations with facial dynamic changes. *Iran J Public Health*, 46(1):93–102 [PMC free article] [PubMed] [Google Scholar]
- [6] Karchani M, Mazloumi A, Saraji GN, et al. (2015). Relationship between Subjective Sleepiness and Demographic Characteristics in Night Work Drivers. *Advances in Environmental Biology*, 9(3):1012–1015 [Google Scholar]
- [7] Ying Y, Jing S, Wei Z. (2007). The Monitoring Method of Driver's Fatigue Based on Neural Network. *Proc. International Conf. on Mechatronics and Automation, China* pp.3555–9. [Google Scholar]
- [8] Lopar M, Ribarić S. (2013). An Overview and Evaluation of Various Face and Eyes Detection Algorithms for Driver Fatigue Monitoring Systems. *Proceedings of the Croatian Computer Vision Workshop, Year 1 Zagreb, Croatia*, pp.15–18 <https://arxiv.org/ftp/arxiv/papers/1310/1310.0317.pdf> [Google Scholar]
- [9] Khan MI, Mansoor AB. (2008). Real Time Eyes Tracking and Classification for Driver Fatigue Detection, *ICIAR 2008, LNCS 5112, Image Analysis and Recognition*, pp: 729–738 [Google Scholar]
- [10] Bergasa LM, Nuevo J, Sotelo MA, Barea R, Lopez ME. (2006). Real-Time system for monitoring driver vigilance. *IEEE trans IntellTranspSyst*, 7 (1): 63–77. [Google Scholar]