iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** VI **Month of publication:** June 2021

DOI: https://doi.org/10.22214/ijraset.2021.35878

www.ijraset.com
Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI Jun 2021- Available at www.ijraset.com

A Vision-based Driver Night Time Assistance and Surveillance System

Sahil Sayyad[1], Kaustubh Patil[2], Digvijay Belan[3], Jayesh Gosavi[4], Arati Kale[5]
[1, 2, 3, 4, 5] Jayawantrao Sawant College of Engineering

Abstract: In India round 1.5 lakh humans died per year in avenue twist of fate because of road accidents most of them were due to low vision and weariness problems. Weariness (Extreme Tiredness) or Fatigue is a major purpose of avenue accidents and has extensive implications for street safety. several deadly accidents can be avoided if the drowsy drivers are warned in time. In many cases it is observed that a car hits some object/obstacle on road due to low vision in that case an object detection and warning system can help to avoid such accidents. Basically, Weariness is a state of sleepiness which abnormally happen when we are very tired or whilst drunken. A spread of drowsiness detection strategies exist that monitor the driving force's drowsiness state at the same time as driving and alarm the drivers if they're no longer concentrating on driving. The relevant features can be extracted from facial expressions including yawning, eye closure and head actions for inferring the level of weariness. The organic condition of driver's body is analyzed for driver weariness detection.

So, this utility overcomes the trouble of driver weariness detection and object/obstacle detection & warning whilst driving using eye extraction, facial extraction, object and its distance detection using different algorithms.

Key Terms: Eye extraction, Facial Extraction, Drowsiness, Weariness, Low vision

I. INTRODUCTION

Every year approximately 1.5 lakh people die in India due road accidents. Now a days traffic accidents have become a major cause of death. Most traffic accidents are caused by driver carelessness under traffic conditions. Therefore, detecting on-road traffic conditions for assisting drivers is need of hour to help drivers take safe driving precautions.This study proposes a vision-based intelligent nighttime driver assistance and surveillance (VIDASS) system. The proposed VIDASS system includes the computer vision and image sensing techniques of night and day time vehicle detection, collision warning determination, and traffic event recording functions by processing the road-scene frames in front of the host car, which are captured from the cameras mounted inside & outside the host vehicle. These proposed vision-based sensing and processing technologies are implemented as a set of different advanced Machine Learning (ML) algorithms like CNN (Convolutional Neural Network). Peripheral devices, including image grabbing devices, network communication modules, and other in-vehicle control devices, are also integrated to produce an in-vehicle embedded vision-based driver assistance and surveillance system. The driver weariness detection system proposed here collects the photo of human from camera mounted inside the car, and explores how these statistics can be used to improve the security at the same time as riding. The photos from the camera are stored together and different machine learning algorithms are applied hence to obtain result from facial and eye movement extraction. If driver seems to be drowsy then it plays the buzzer alarm and growth buzzer sound in each 2 sec. It also has the functionality to send an email or SMS to the family member of driver, the Cab company, or the passenger's family regarding the situation of driver. It could help to prevent severe injury even death due to road accidents. Image detection and identification is one of the most common applications of machine learning. The system proposed here can be used in different types of road vehicles specially in cabs and transport vehicle. Our system can be used for safe and efficient driving experience while preventing severe injury and death due to dangerous road accidents.

II. RELATED WORK

Driver Assistance System for Lane Detection and Vehicle Recognition with Night Vision Chun-Che Wang1,Shih-Shinh Huang and Li-Chen Fu The objective of this research is to develop a vision-based driver assistance system to enhance the driver's safety in the night-time. The proposed system performs both lane detection and vehicle recognition. In lane detection, three features including lane markers, brightness, slenderness and proximity are applied to detect the positions of lane markers in the image. On the other hand, vehicle recognition is achieved by using an evident feature which are extracted through three four steps: taillight standing-out process, adaptive thresholding, centroid detection, and taillight pairing algorithm. Besides, an automatic method is also provided to calculate the tilt and the pan of the camera by using the position of vanishing point which is detected in the image by applying Canny edge detection, Hough transform, major straight line extraction and vanishing point estimation.

Experimental results for thousands of images are provided to demonstrate the effectiveness of the proposed approach in the nighttime. The lane detection rate is nearly 99%, and the vehicle recognition rate is about 91%. Furthermore, our system can process the image in almost real time. Advanced Driver Assistance Systems Information Management and Presentation Alexandre Dugarry: With the development of advanced driving assistance systems, in-vehicle communication and information systems, there are situations where the driver becomes overloaded by information, creating potentially dangerous conditions .In this Thesis a novel strategy is proposed, to priorities and present information. Firstly two main criteria are extracted, that allow the ability to rank messages: the risk associated with the non-presentation of the message, and its relevance to the environment. Fuzzy cognitive maps enable to represent expert knowledge and model these relationships.

Secondly, a strategy to present information is proposed. Using an importance index, calculated from the previous risk and relevance indices, but also information nature, time constraints and access frequency, a set of best interfaces is selected. Furthermore design a model of driver workload is designed, based on the multiple resources theory. By estimating in real time the workload of the driver, the system enables to choose an optimal interface, that should prevent overload .This Thesis presents then the tools developed for the implementation and testing of the model. A video capture and data transfer program, based on the IEEE-1394 bus, enable in-vehicle real-time data capture and collection.

Moreover, a software package for replay of the acquired data, analysis and simulation is developed. Finally, the implementation of the prioritization and presentation strategy is outlined. The last part of this work is dedicated to the experiments and results. Using an experimental vehicle, data in different driving conditions are collected. The experiment is completed by creating data to simulate potentially dangerous situations, where driver is overloaded with information.

Real Time Driver Drowsiness Detection Based on Driver'sFace Image Behavior Using a System of Human Computer Interaction Implemented in a Smartphone. Eddie Galarza, Fabrico Egas, Franklin Silva, Paola Velasco:The main reason for motor vehicular accidents is the driver drowsiness. This work shows a surveillance system developed to detect and alert the vehicle driver about the presence of drowsiness. It is used a smartphone like small computer with a mobile application using Android operating system to implement the Human Computer Interaction System. For the detection of drowsiness, the most relevant visual indicators that reflect the driver's condition are the behavior of the eyes, the lateral and frontal assent of the head and the yawn. The system works adequately under natural lighting conditions and no matter the use of driver accessories like glasses, hearing aids or a cap. Due to a large number of traffic accidents when driver has fallen asleep this proposal was developed in order to prevent them by providing a non-invasive system, easy to use and without the necessity of purchasing specialized devices. The method gets 93.37% of drowsiness detections.

Real-Time Driver Drowsiness Detection System Using Eye Aspect Ratio and Eye Closure Ratio. Sukrit Mehta, Sharad Dadhich, Sahil Gumber, Arpita Jadhav Bhatt: Every year many people lose their lives due to fatal road accidents around the world and drowsy driving is one of the primary causes of road accidents and death. Fatigue and micro sleep at the driving controls are often the root cause of serious accidents. However, initial signs of fatigue can be detected before a critical situation arises and therefore, detection of driver's fatigue and its indication is ongoing research topic. Most of the traditional methods to detect drowsiness are based on behavioral aspects while some are intrusive and may distract drivers, while some require expensive sensors. Therefore, in this paper, a light-weight, real time driver's drowsiness detection system is developed and implemented on Android application. The system records the videos and detects driver's face in every frame by employing image processing techniques. The system is capable of detecting facial landmarks, computes Eye Aspect Ratio (EAR) and Eye Closure Ratio (ECR) to detect driver's drowsiness based on adaptive thresholding. Machine learning algorithms have been employed to test the efficacy of the proposed approach. Empirical results demonstrate that the proposed model is able to achieve accuracy of 84% using random forest classifier

III.PROPOSED WORK

Developing predictive system for weariness prediction and object detection. Tools which are conventionally used for developing model are PyCharm. Python programming language and different algorithms like YOLO and Dlib Various steps involved are:

A. Proposed System

The proposed system has basically divided into two modules:

1) Weariness Detection Module
2) Object Detection with Distance Estimation with Warning

The objective of this Python project is to build a weariness and object/obstacle detection system . Weariness detection System will detect that a person's eyes are closed for a few seconds. This system will alert the driver when drowsiness is detected. In this Python project, we will be using "OpenCV" for gathering the images from webcam and feed them into a Deep Learning model which will classify whether the person's eyes are 'Open' or 'Closed'. The approach we will be using for this Python project is as follows: *Step 1 – Take image as input from a camera. Step 2 – Detect the face in the image and create a Region of Interest (ROI). Step 3 – Detect the eyes from ROI and feed it to the classifier. Step 4 – Classifier will categorize whether eyes are open or closed. Step 5 – Calculate score to check whether the person is drowsy*

a) *Facial Expression:* The video camera in front of driver was set to record parts of the participant's face. The subjective evaluation of their drowsiness levels was then processed offline by two evaluators in intervals of 10 s in accordance with predetermined criteria. The evaluation of drowsiness levels based on the features of facial expressions has been defined in several different methods. In this study, the scale for drowsiness levels was based on the Zilberg's criteria, which uses whole integer numbers ranging from 0 (alert state) to 4 (extremely drowsy state). Facial Action Coding System (FACS):(This is the face recog. Software) It is one of the most popular expression coding systems, used to code facial expressions. The facial expressions are decomposed to 46 component movements, this number corresponds to the number of an individual's facial muscles. Head motions can be detected through an automatic eye tracking and an accelerometer. FACS is also capable of discovering new patterns based on emotional states.
b) *Support Vector Machines (SVM):* For face detection, the Haar feature algorithm is used. Each feature is classified by a Haar feature classifier. It takes the captured face as input, detected face as output. Detects eyes image from this detected face and this detected eye is sent to the ML algorithm for further ML processes. SVM is used to identify whether the eyes are closed or open. An SVM can be trained to detect the face and see whether the eyes are shut to open and then decide to trigger the alarm or not. The training set has a set of images that have eyes shut and some set of images that have eyes open. When the model is built, it will be used to classify any new pre-processed eye-image. An ML classifier is built using this algorithm to classify the pre-processed eye image. SVMs can efficiently solve linear or non-linear classification problems. It maximizes the margin around the separating hyperplane and then can find an optimal classifier.
c) *The Model Architecture:* The model we used is built with Keras using Convolutional Neural Networks (CNN). A convolutional neural network is a special type of deep neural network which performs extremely well for image classification purposes. A CNN basically consists of an input layer, an output layer and a hidden layer which can have multiple numbers of layers. A convolution operation is performed on these layers using a filter that performs 2D matrix multiplication on the layer and filter. The CNN model architecture consists of the following layers:

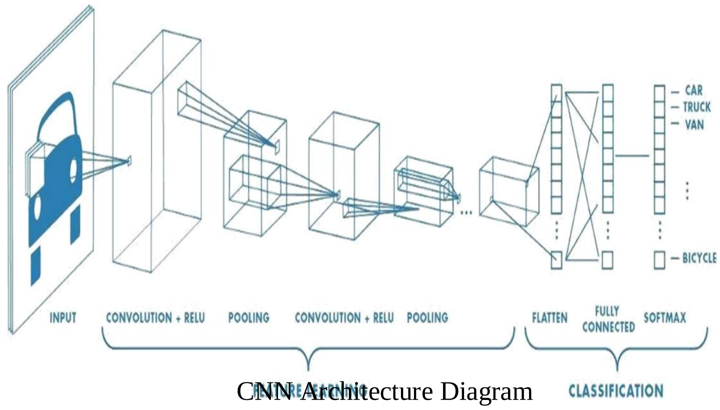
- Convolutional layer; 32 nodes, kernel size 3
- Convolutional layer; 32 nodes, kernel size 3
- Convolutional layer; 64 nodes, kernel size 3
- Fully connected layer; 128 nodes

The final layer is also a fully connected layer with 2 nodes. In all the layers, a Relu activation function is used except the output layer in which we used Softmax.30

B. CNN

CNN could be a style of deep learning model for handling information that contains a matrix design, similar to pictures, which is motivated by the association of creature cortical locale and intended to naturally and adaptively learn spatial orders of highlights, from low-to undeniable level examples.
CNN could be a numerical develop that is commonly made out of three types of layers (or building blocks): convolution, pooling, and completely associated layers.
CNN Architecture comprises of 3 BLOCK LAYER :

1) Fully Connected Layer yield highlight guides of a definitive convolution or pooling layer is generally straightened.
2) Pooling Layer gives a commonplace down examining activity which decreases the in-plane dimensionality.
3) Convolution Layer is a key segment of the CNN design that performs highlight extraction



CNN Architecture Diagram

C. Yolo Algorithm

YOLO algorithm is an algorithm based on regression, instead of selecting the interesting part of an Image, it predicts classes and bounding boxes for the whole image in one run of the Algorithm.

To understand the YOLO algorithm, first we need to understand what is actually being predicted. Ultimately, we aim to predict a class of an object and the bounding box specifying object location. Each bounding box can be described using four descriptors:

1) Center of the box (bx, by)
2) Width (bw)
3) Height (bh)
4) Value c corresponding to the class of an object

Along with that we predict a real number pc, which is the probability that there is an object in the bounding box.With YOLO, a single CNN simultaneously predicts multiple bounding boxes and class probabilities for those boxes. YOLO trains on full images and directly optimizes detection performance. This model has a number of benefits over other object detection methods:

a) YOLO is extremely fast.
b) YOLO sees the entire image during training and test time so it implicitly encodes contextual information about classes as well as their appearance.
c) YOLO learns generalizable representations of objects so that when trained on natural images and tested on artwork, the algorithm outperforms other top detection methods.

YOLO (You Only Look Once Object Detection Algorithm): *"It is defined as a state of Art algorithm for real- time object detection system."*

Object detection is a field of Computer Vision and Image Processing that deals with detecting instances of various classes of objects (like a person, book, chair, car, bus, etc.) in a digitally captured Image or Video. This domain is further divided into sub-domains like Face detection, Activity recognition, Image annotation, and many more. Object Detection has found its applications in various important areas like Self-Driving cars, robots, Video Surveillance, Object Tracking, etc. Other old algorithms used for object detection before YOLO:

1.Fast R-CNN, 2.Single Shot MultiBox Detector (SSD),

3.Retina-Net.But these algorithms were difficult to implement and they had some disadvantages. So we choose best algorithm i.e. YOLO algorithm.

D. Weariness Detection Algorithm

Step 1 – Take image as input from a camera, Step 2 – Detect the face in the image and create a Region of Interest (ROI).

Step 3 – Detect the eyes from ROI and feed it to the classifier, Step 4 – Classifier will categorize whether eyes are open or closed,

Step 5 – Calculate score to check whether the person is drowsy.

1) *Algorithm step by Step*

a) *Step 1* – Take Image as Input from a Camera With a webcam, we will take images 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.

b) *Step 2* – Detect Face in the Image and Create a Region of Interest (ROI) To detect the face in the image, 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. For (x,y,w,h)in faces:

cv2.rectangle(frame, (x,y), (x+w, y+h), (100,100,100), 1)

c) *Step 3* –Detect the eyes from ROI and feed it to the classifier

l_eye = frame[y : y+h, x : x+w]

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.

d) *Step 4* – Classifier will Categorize whether Eyes are Open or Closed 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.

First, we convert the color image into grayscale using

r_eye = cv2.cvtColor(r_eye, cv2.COLOR_BGR2GRAY).

Then, we resize the image to 24*24 pixels as our model was trained on 24*24 pixel images cv2.resize(r_eye, (24,24)).

We normalize our data for better convergence r_eye = r_eye/255 (All values will be between 0-1). Expand the dimensions to feed into our classifier.

We loaded our model using

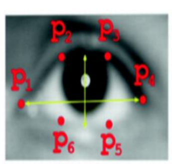
model = load_model('models/cnnCat2.h5') . Now we predict each eye with our model lpred = model.predict_classes(l_eye). If the value of lpred[0] = 1, it states that eyes are open, if value of lpred[0] = 0 then, it states that eyes are closed.

e) *Step 5* –Calculate Score to Check whether Person is Drowsy the score is basically a value we will use to determine how long the person has closed his eyes. 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.

Cv2.putText(frame, "Open", (10, height-20), font, 1, (255,255,255), 1, cv2.LINE_AA)

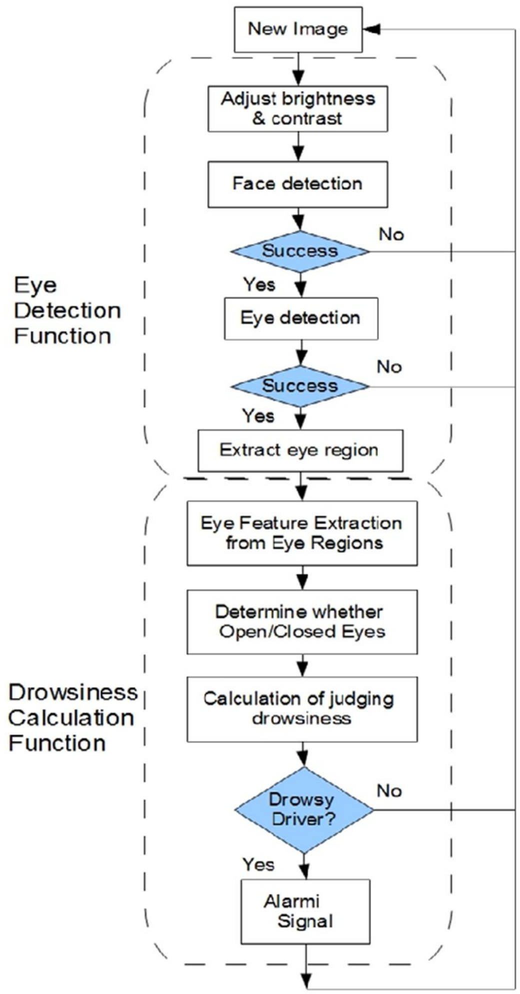
The same procedure to detect faces is used to detect eyes. First, we set the cascade classifier for eyes in leye and reye respectively then detect the eyes using left_eye = leye.detectMultiScale(gray). Now we need to extract only the eyes data from the full image. This can be achieved by extracting the boundary box of the eye and then we can pull out the eye image from the frame with this code



$$\text{EAR} = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|}$$

\

IV. ARCHITECTURE DIAGRAM FLOW



V. WORKING

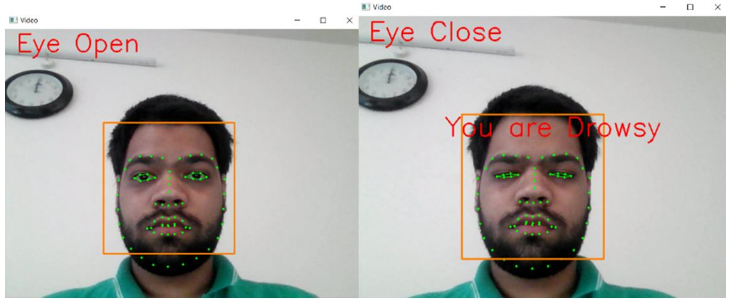
With a webcam, we will take images as input. So to access the webcam, we made an infinite loop that will capture each frame. We use the method provided by OpenCV to access the camera and set the capture object we will read each frame and we store the image in a frame variable. To detect the face in the image, 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 Then we perform the detection using faces 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. The same procedure to detect faces is used to detect eyes. First, we set the cascade classifier for eyes in leye and reye respectively then detect the eyes Now we need to extract only the eyes data from the full image. This can be achieved by extracting the boundary box of the eye and then we can pull out the eye image from the frame with this code. 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.

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. First, we convert the color image into grayscale Then, we resize the image to pixels as our model was trained on pixels images We normalize our data for better convergence then, we states that eyes are closed.The score is basically a value we will use to determine how long the person has closed his eyes. 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 a function which will display real time status of the person. 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. This is when we beep the alarm using sound.play().

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI Jun 2021- Available at www.ijraset.com

VI. OUTPUT

A. Weariness Detection



1) Warning Email Sent Status

```
Limit Cross Now I am sending an alert message to your family members
Mail Successfully send to Ids ['sayyadsahil7@gmail.com', 'sayyadsahil162000@gmail.com', 'belandigvijay@gmail.com']
Mail Successfully send to Ids ['sayyadsahil7@gmail.com', 'sayyadsahil162000@gmail.com', 'belandigvijay@gmail.com']
Mail Successfully send to Ids ['sayyadsahil7@gmail.com', 'sayyadsahil162000@gmail.com', 'belandigvijay@gmail.com']
Mail Successfully send to Ids ['sayyadsahil7@gmail.com', 'sayyadsahil162000@gmail.com', 'belandigvijay@gmail.com']
```

B. Object Detection, Distance & Warning

1) Alert

2) High Alert

VII. CONCLUSION

Thus, we can conclude that A Vision-based Driver Night Time Assistance and Surveillance System yields efficient and significant results and can be used in the real world to avoid and prevent deadly road accidents. Further this system can be expanded as per requirement. The result of this system depends on the quality of video captured by camera. Hence High-Definition good quality videos will definitely yield best and most accurate results.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI Jun 2021- Available at www.ijraset.com

REFERENCES

[1] “Road Accidents in India 2016,” 2016. S. Sangle, B. Rathore, R. Rathod, A. Yadav, and A. Yadav
[2] “Real Time Drowsiness Detection System,” pp. 87–92, 2018. V. Varghese, A. Shenoy, S. Ks, and K. P. Remya
[3] “Ear Based Driver Drowsiness Detection System,” vol. 2018, pp. 93–96, 2018. A. Kumar
[4] and R. Patra
[5] “Driver drowsiness monitoring system using visual behaviour and machine learning,” ISCAIE 2018 – 2018 IEEE Symposium on Computer Applications and Industrial Electronics, pp. 339–344, 2018. T. Hwang, M. Kim, S. Hong, and K. S. Park
[6] “Driver drowsiness detection using the in-ear EEG,” Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, vol. 2016–Octob, pp. 4646–4649, 2016. S. Junawane, S. Jagtap, P. Deshpande, and L. Soni
[7] “Driver Drowsiness Detection Techniques : A Survey,” vol. 6, no. 11, pp. 2015–2017, 2017. R. Jabbar, K. Al-Khalifa, M. Kharbeche, W. Alhajyaseen, M. Jafari, and S. Jiang
[8] “Real-time Driver Drowsiness Detection for Android Application Using Deep Neural Networks Techniques,” Procedia Computer Science, vol. 130, pp. 400–407, 2018. T. Soukupova and J. Cech
[9] “Real-time eye blink detection using facial landmarks,” Computer Vision Winter Workshop (CVWW), 2016. I. García, S. Bronte, L. M. Bergasa, J. Almazán, and J. Yebes
[10] “Vision-based drowsiness detector for real driving conditions,” IEEE Intelligent Vehicles Symposium, Proceedings, pp. 618–623, 2012. S. S. Nagargoje and D. S. Shilvant
[11] “Drowsiness Detection System for Car Assisted Driver Using Image Processing,” International Journal of Electrical and Electronics Research ISSN, vol. 3, no. 4, pp. 175–179, 2015. S. Podder and S. Roy
[12] “Driver’s drowsiness detection using eye status to improve the road safety,” International Journal of Innovation Research in Computer and Communication Engineering, vol. 1, no. 7, 2013. F. Omidi and G. Nasleseraji.
[13] Drowsiness Detection with Machine Learning | by Grant Zhong | Towards Data Science
[14] https://machinelearningmastery.com/how-to-perform-object-detection-with-yolov3-in-keras/
[15] YOLO Algorithm and YOLO Object Detection: An Introduction – Appsilon | End¬ to¬ End Data Science Solutions
[16] https://towardsdatascience.com/yolo-you-only-look-once-real-time-object-detection-explained-492dc9230006
[17] https://pypi.org/project/dlib/
[18] https://en.wikipedia.org/wiki/Machine_learning
[19] “Road Accidents in India 2016,” 2016.
[20] S. Sangle, B. Rathore, R. Rathod, A. Yadav, and A. Yadav, “Real Time Drowsiness Detection System,” pp. 87–92, 2018.
[21] V. Varghese, A. Shenoy, S. Ks, and K. P. Remya, “Ear Based Driver Drowsiness Detection System,” vol. 2018, pp. 93–96, 2018.
[22] A. Kumar and R. Patra, “Driver drowsiness monitoring system using visual behaviour and machine learning,” ISCAIE 2018 - 2018 IEEE Symposium
[23] on Computer Applications and Industrial Electronics, pp. 339–344, 2018.
[24] T. Hwang, M. Kim, S. Hong, and K. S. Park, “Driver drowsiness detection using the in-ear EEG,” Proceedings of the Annual International
[25] Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, vol. 2016–Octob, pp. 4646–4649, 2016.
[26] S. Junawane, S. Jagtap, P. Deshpande, and L. Soni, “Driver Drowsiness Detection Techniques : A Survey,” vol. 6, no. 11, pp. 2015–2017, 2017.
[27] R. Jabbar, K. Al-Khalifa, M. Kharbeche, W. Alhajyaseen, M. Jafari, and S. Jiang, “Real-time Driver Drowsiness Detection for Android Application
[28] Using Deep Neural Networks Techniques,” Procedia Computer Science, vol. 130, pp. 400–407, 2018.
[29] T. Soukupova and J. Cech, “Real-time eye blink detection using facial landmarks,” Computer Vision Winter Workshop (CVWW), 2016.
[30] I. García, S. Bronte, L. M. Bergasa, J. Almazán, and J. Yebes, “Vision-based drowsiness detector for real driving conditions,” IEEE Intelligent
[31] Vehicles Symposium, Proceedings, pp. 618–623, 2012.
[32] S. S. Nagargoje and D. S. Shilvant, “Drowsiness Detection System for Car Assisted Driver Using Image Processing,” International Journal of
[33] Electrical and Electronics Research ISSN, vol. 3, no. 4, pp. 175–179, 2015.
[34] S. Podder and S. Roy, “Driver’s drowsiness detection using eye status to improve the road safety,” International Journal of Innovation Research
[35] in Computer and Communication Engineering, vol. 1, no. 7, 2013.
[36] F. Omidi and G. Nasleseraji, “Non-intrusive Methods used to Determine the Driver Drowsiness: Narrative Review Articles,” International Journal
[37] of Occupational Hygiene, vol. 8, no. 3, pp. 186–191, 2016.
[38] A. Gopal and V. . Vineeth, “Driver Drowsiness Detection System,” International Journal of Advances in Electronics and Computer Science, vol.
[39] 4, no. 3, pp. 101–104, 2017.
[40] K. Srijayathi and M. Vedachary, “Implementation of the Driver Drowsiness Detection System,” International Journal of Science, Engineering and
[41] Technology Research (IJSETR), vol. 2, no. 9, pp. 1751–1754, 2013.
[42] A. Chellappa, M. S. Reddy, R. Ezhilarasie, S. Kanimozhi Suguna, and A. Umamakeswari, “Fatigue detection using Raspberry Pi 3,” International
[43] Journal of Engineering and Technology(UAE), vol. 7, no. 2, pp. 29–32, 2018.
[44] H. Singh, J. S. Bhatia, and J. Kaur, “Eye tracking based driver fatigue monitoring and warning system,” India International Conference on Power
[45] Electronics, IICPE 2010, 2011.

AUTHORS PROFILE

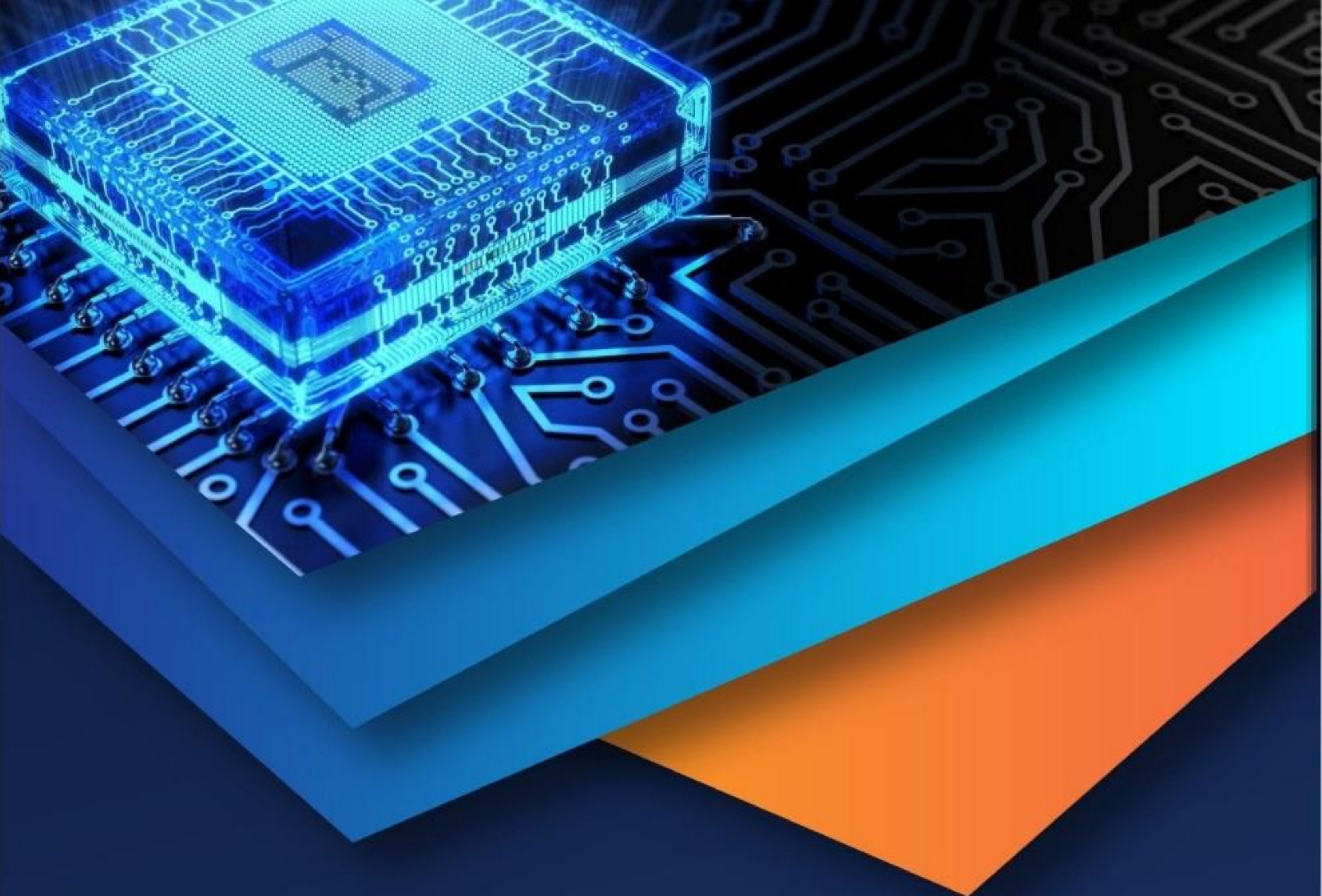
Sahil Sayyad, an undergraduate, is pursuing Bachelor of Engineering, in the branch of Information Technology from Department Of Information Technology, JSPM's Jaywantrao Sawant College of Engineering, and Pune and currently is in her final year. Area of Interest is Machine Learning, Deep Learning.

Kaustubh Patil, an undergraduate, is pursuing Bachelor of Engineering, in the branch of Information Technology from Department Of Information Technology, JSPM's Jaywantrao Sawant College of Engineering, and Pune and currently is in her final year. Area of Interest is Deep Learning.

Digvijay Belan, an undergraduate, is pursuing Bachelor of Engineering, in the branch of Information Technology from Department Of Technology, JSPM's Jaywantrao Sawant College of Engineering, Pune and currently is in his final year. Area of Interest is IoT, Machine Learning.

Jayesh Gosavi, an undergraduate is pursuing Bachelor of Engineering, in the branch of Information Technology from Department Of Information Technology, JSPM's Jaywantrao Sawant College of Engineering, and Pune and currently is in his final year. Area of interest is GUI and frontend.

Prof. Arati Kale, Currently working as Assistant Professor Department of Information Technology JSPM's Jayawantrao Sawant college of Engineering Hadapsar ,Pune,India. Qualification: Masters in Engineering (M.E Computer Science), she has total teaching experience of 9 years Area of Interest is Machine Learning, Data Mining, Deep Learning.



10.22214/IJRASET

INDEX COPERNICUS 45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY