International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 05 | May - 2023 SJIF 2023: 8.176 ISSN: 2582-3930

INTELLIGENT ACCIDENT AVOIDING SYSTEM IN AUTOMOBILE USING RASPBERRY PI.

Janhvi Dhande[1], Yash Funde[2], Akshay Chavan[3]

[1]*Electronics and Telecommunication, Sinhgad college of Engineering*
[2]*Electronics and Telecommunication, Sinhgad college of Engineering*
[3]*Electronics and Telecommunication, Sinhgad college of Engineering*

---***---

Abstract - This system aims to make driving vehicles safer and protect against drowsiness. Alcohol detection prevents accidents and collisions between vehicles while driving and reduces road accidents. This product will come to prepare a combination of face detection and face contours, as well as an additional feature of the driver's alcohol consumption, and in accordance, the vehicle acceleration is maintained. This product is made up of deep learning algorithms that are used in conjunction with a microcontroller and interface with a microprocessor. The face will be detected using computer vision and contours will be formed around it. A set of devices is used to check the person for drowsiness and then for alcohol. The set of devices examines the person's alcohol parameters. To show and notify alertness, the device in this paper employs a display interface. It sends a message to the person in charge to pick up the alcoholic. When the ignition lock is reset and the quality of alcohol consumed is checked, the ignition lock is removed. It will determine where the alcohol consumption device will exit. The device will detect alcohol consumption and stop the vehicle accordingly. The Open CV library is being used to aid in the detection of face drowsiness. To detect collisions, the product includes a distance tracker on each side of the vehicle that detects the distance between the vehicle with the model and the vehicles around it.

***Key Words*:** Internet of Things, Haar-cascade, Open CV, Euclidian distance, Deep Learning.

1.INTRODUCTION

The number of vehicles is rapidly increasing, and this trend has led to an increased number of vehicle accidents. The National Highway Traffic Safety Administration (NHTSA) Fatality Analysis Reporting System Encyclopedia found that approximately 55,926 vehicles were involved in vehicular accidents in the United Statesian 2007. Moreover, 9,979 vehicle collision accidents were due to Driver blinking and inattention in 2007. The NHTSA's National Centre for Statistics and Analysis also provided a brief statistical summary report indicating that 416,000 accidents were caused by eye blinking driving between 2005 and 2009. Traffic vehicle accidents cause many injuries and deaths every day around the world. For instance, the NHTSA stated that approximately 1,500 people were killed in traffic accidents in 2002. Furthermore, there were 71,000 injuries resulting from the 100,000 collisions between 1989 and 1992. Driver blinking is when a driver's ability to drive safely is reduced as a result of being physically or mentally tired or sleepy. Driver blinking or is a significant safety hazard for the road transport industry. The main causes of 'eye blinking driving' are too little sleep, driving at times when you would normally be asleep and working or being awake for very long hours. To detect driver drowsiness can be classified into three categories:

1. Vehicle-based approaches,
2. Behavior-based approaches, and
3. Physiological-signal based approaches.

Fatigue, drowsiness and sleepiness are often used synonymously in driving state description. Involving multiple human factors, it is multidimensional in nature that researchers have found difficult to define over past decades. Despite the ambiguity surrounding fatigue, it is a critical factor for driving safety. Studies have shown that fatigue is one of the leading contributing factors in traffic accidents worldwide. It will be also uses alcohol pulse detection to check out the person is normal or abnormal. It is particularly critical for occupational drivers, such as drivers of buses and heavy trucks, due to the fact that they may have to work over a prolonged duration of the driving task, during the peak drowsiness periods.

2.1 PROBLEM STATEMENT

•This system treats the automatic detection of driver drowsiness based on visual information and artificial intelligence. We locate, track and analyze both the driver face and eyes to measure PERCLOS (percentage of eye closure) with Softmax for neural transfer function.

• To avoid the accident, we are using image processing for human action recolonization.

• When it comes to knowledge of the risk of falling asleep, the drivers were confronted with several statements concerning characteristics of drivers who fall asleep (age, sex, physical condition, sleeping problems) in addition to a statement that falling asleep can happen to anyone. Based on the driver's evaluation of these statements, it seems to be a general agreement among them, both private and professionals, that falling asleep can happen to anyone. In addition, they seem to have good knowledge of the actual risk of falling asleep while driving. The private drivers and the professional drivers respectively assume that an average of 40 and 36 drivers out of a hundred drivers have experienced falling asleep while driving. Calculated in percent these numbers make up shares that are close to the actual proportions found in this study. it will be also uses alcohol pulse detection to check out the person is normal or abnormal.

2.2 OBJECTIVE

• The motivation of this project is to develop a system to keep the vehicle secure and protect it by the occupation of the intruders. And also, to avoid the accidental conditions.

• Behind a driver decision will be based on their awareness (knowledge) and their skills to respond to various traffic situations. This involves internalizing the driving process and owning the decisions to drive appropriately.

• The skill of looking up ahead and scanning for hazards depends on the driver's motivation to choose to do so and this will be linked to their awareness of the possible risks or hazards that looking ahead may detect. The motivation behind a driver's decisions will be based on their awareness (knowledge) and their skills to respond to various traffic situations. This involves internalizing the driving process and owning the decisions to drive appropriately.

• This contrasts with the simpler example of motivation where a driver slows down if a police officer is in the vicinity. There is little internalizing occurring.

• It will be also uses alcohol pulse detection to check out the person is normal or abnormal.

• In the case of a person driving for their employment, he or she may be motivated to override the need for safe driving behavior in order to meet performance requirements in their work especially if there is an incentive for doing so. The issue for those attending driver training is around their motivation to change their driving behavior. This needs to be driven internally rather than through factors such as enforcement.

2.3 PROPOSED METHOLOGY

A. Tools Used

• OpenCV python OpenCV-Python is a library of Python bindings designed to solve computer vision problems. OpenCV (Open-Source Computer Vision Library) is an open-source BSD-licensed library that includes several hundreds of computer vision algorithms. OpenCV-Python is the Python API for OpenCV, combining the best qualities of the OpenCV C++ API and the Python language. OpenCV-Python makes use of NumPy, which is a highly optimized library for numerical operations with a Python-style syntax. All the OpenCV array structures are converted to and from NumPy arrays. This also makes it easier to integrate with other libraries that use NumPy such as SciPy and Matplotlib. Equations.

• TensorFlow: TensorFlow is an open-source software library for machine learning across a range of tasks. It is a symbolic math library, and also used as a system for building and training neural networks to detect and decipher patterns and correlations, analogous to human learning and reasoning. TensorFlow provides a Python API as well as C++, Haskell, Java, Go and Rust API

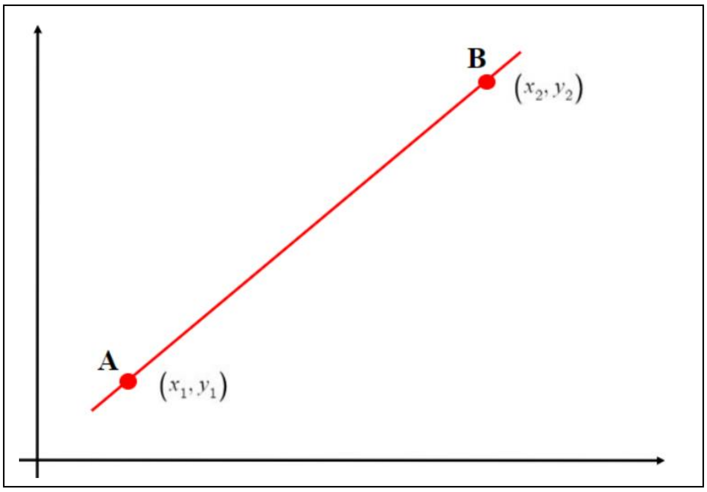
• Anaconda Navigator: Anaconda Navigator is a desktop graphical user interface (GUI) included in Anaconda distribution that allows users to launch applications and manage condo packages, environments and channels without using command- line commands. Navigator can search for packages on Anaconda Cloud or in a local Anaconda Repository, install them in an environment, run the packages and update them. It is available for Windows, macOS and Linux.

B. Equations

Euclidian Distance: - Euclidean Distance Metric is one of the most used distance metrics in the machine learning algorithm. It gives the shortest distance between two points. The distance between two points in either the plane or 3-D space measures the length of the segment between two points. The distance between two points in either the plane or 3-D space measures the length of the segment between two points. They are generally used to calculate the distance between two rows of data that have numerical values (Integer or Decimal Values).

- KNN- classifier uses a Euclidean metric to classify the unknown instances by calculating the distance between the points in the training set.
- The value of the Euclidean distance will be greater than or equal to zero.
- If the value of the Euclidean Distance is Equal to zero, it implies that both the points are equivalent; else, they are different from each other.

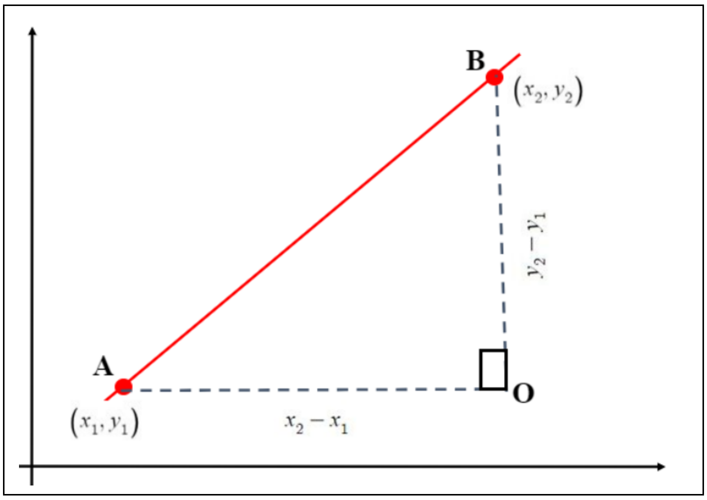
The Euclidean distance formula can be easily derived using the Pythagoras theorem. Pythagoras' Theorem states: "In a right-angle triangle, the sum of the squares of the base and perpendicular is equal to the square of the hypotenuse."



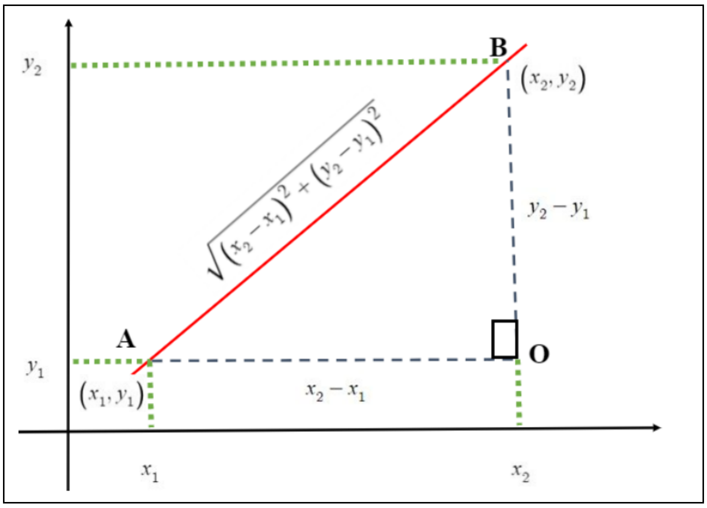
Now, we will calculate the Euclidean distance using the Pythagoras Formula:
Let's take any two points (A, B) on a line segment that have coordinates A (x1, y1), and B (x2, y2), where (x1, x2) are the points on the x-axis and (y1, y2) are the points on y-axis.

Now, using these two points, draw a right-angle triangle, having the right angle at O, So the distance between AO = (x2 – x1)
BO = (y2 – y1)

$$(AB)^2 = (AO)^2 + (OB)^2$$
$$\Rightarrow AB = \sqrt{(AO)^\wedge 2 + (OB)^\wedge 2}$$
$$\Rightarrow AB = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$



Now, using the Pythagoras Theorem, we will get the Euclidean distance between two points (here AB), i.e.,



Hence, the Euclidean distance between two points is:

```
Import cv2
# colored Image
Img = cv2.imread (&ldquo;Penguins.jpg&rdquo;,1)
# Black and White (gray scale)
Img_1 = cv2.imread (&ldquo;Penguins.jpg&rdquo;,0)
```

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

The general formula of Euclidean Distance metric in *n*-dimension space is given by:

$$d(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + (p_3 - q_3)^2 + \ldots\ldots + (p_n - q_n)^2}$$
or,
$$d(p, q) = \left(\sum_{i=1}^{n} (p_i - q_i)^2\right)^{\frac{1}{2}}$$

Where,

n: number of dimensions

(pi, qi): data points

C. Open CV

1. Loading an image using OpenCV:

As seen in the above piece of code, the first requirement is to import the OpenCV module.

Later we can read the image using **imread** module. The 1 in the parameters denotes that it is a color image. If the parameter was 0 instead of 1, it would mean that the image being imported is a black and white image. The name of the image here is 'Penguins'.

2. Image Shape/Resolution:

We can make use of the shape sub-function to print out

```
Import cv2

# Black and White (gray scale)

Img=cv2.imread (&ldquo;Penguins.jpg&rdquo;,0)

Print(img.shape)
```

the shape of the image. Check out the below image:

By shape of the image, we mean the shape of the NumPy array. As you see from executing the code, the matrix consists of 768 rows and 1024 columns.

3. Displaying the image:

Displaying an image using OpenCV is pretty simple and straightforward. Consider the below image:

```
import cv2

# Black and White (gray scale)

Img = cv2.imread
(&ldquo;Penguins.jpg&rdquo;,0)

cv2.imshow(&ldquo;Penguins&rdquo;, img)

cv2.waitKey(0)

# cv2.waitKey(2000)

cv2.destroyAllWindows()
```

As you can see, we first import the image using **imread.** We require a window output to display the images, right?

We use the **imshow** function to display the image by opening a window. There are 2 parameters to the **imshow** function which is the name of the window and the image object to be displayed.

Later, we wait for a user event. **waitKey** makes the window static until the user presses a key. The parameter passed to it is the time in milliseconds.

And lastly, we use **destroyAllWindows** to close the window based on the waitForKey parameter.

4. Face Detection Using OpenCV

This seems complex at first but it is very easy. Let me walk you through the entire process and you will feel the same.

Step 1: Considering our prerequisites, we will require an image, to begin with. Later we need to create a cascade classifier which will eventually give us the features of the face.

Step 2: This step involves making use of OpenCV which will read the image and the features file. So at this point, there are NumPy arrays at the primary data points.

All we need to do is to search for the row and column values of the face NumPy ndarray. This is the array with the face rectangle coordinates.

Step 3: This final step involves displaying the image with the rectangular face box.

2.4 BLOCK DIAGRAM

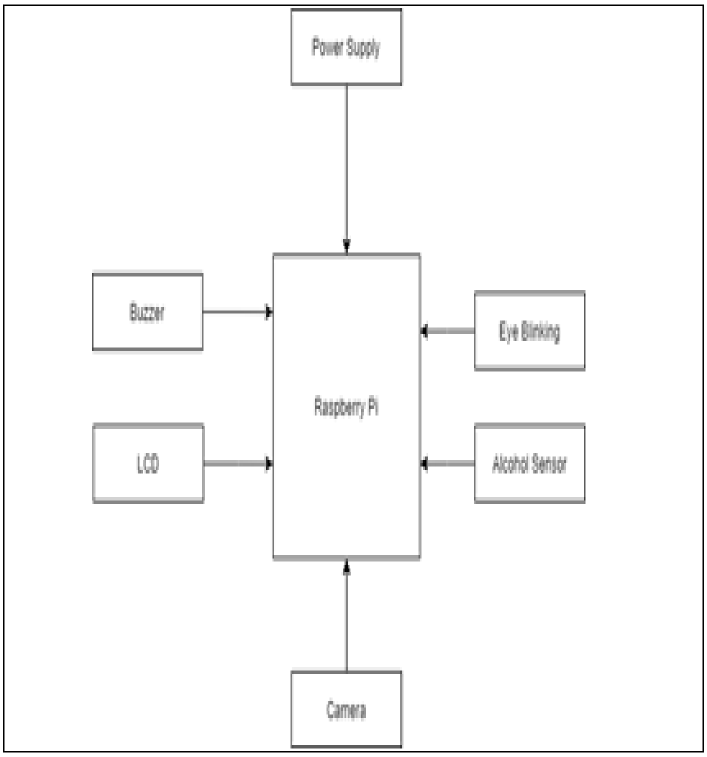


Fig 1. Block Diagram of System

2.5 ALGORITHM

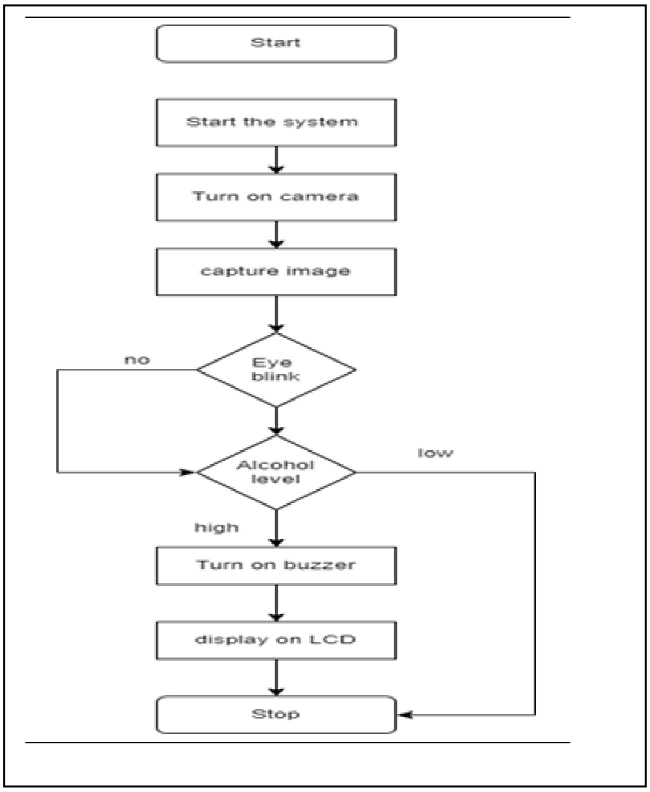


Fig 2. Algoritham

2.6 RESULT AND COMPONENTS

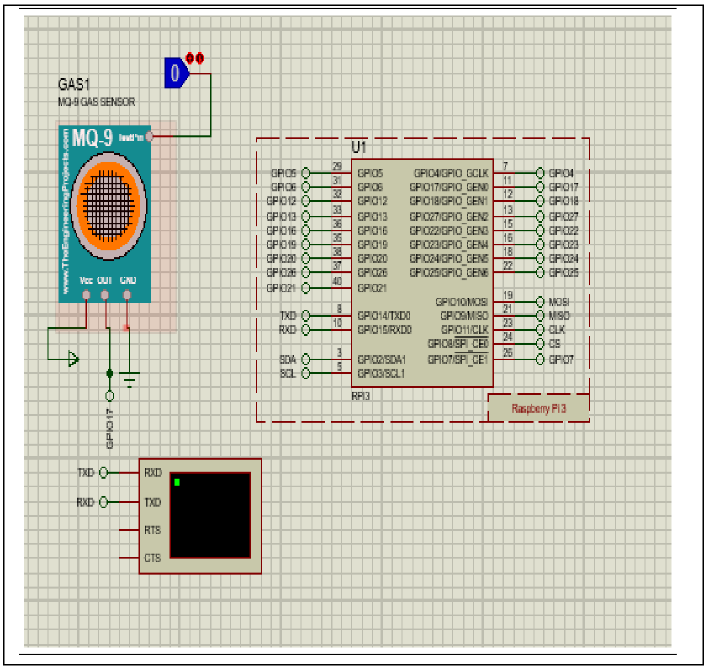


Fig 3. Proetus Simulation

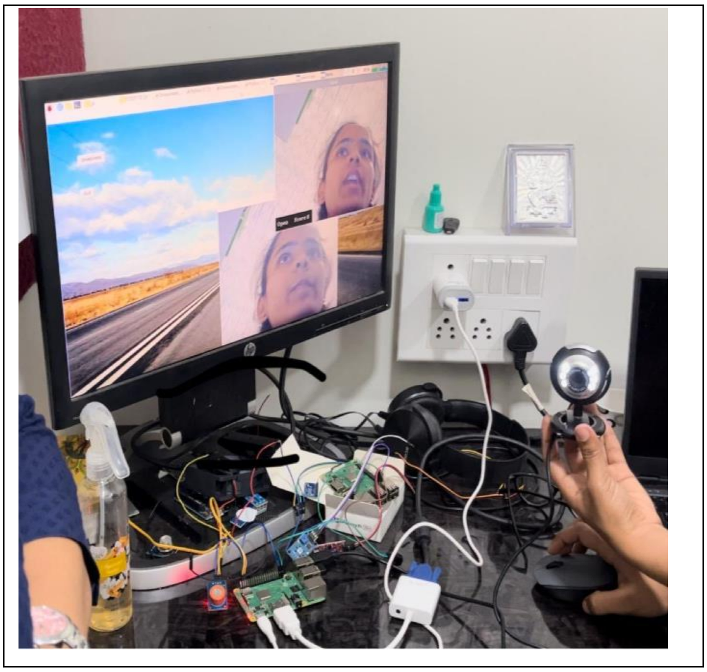


Fig 4. Image Capturing and Processing

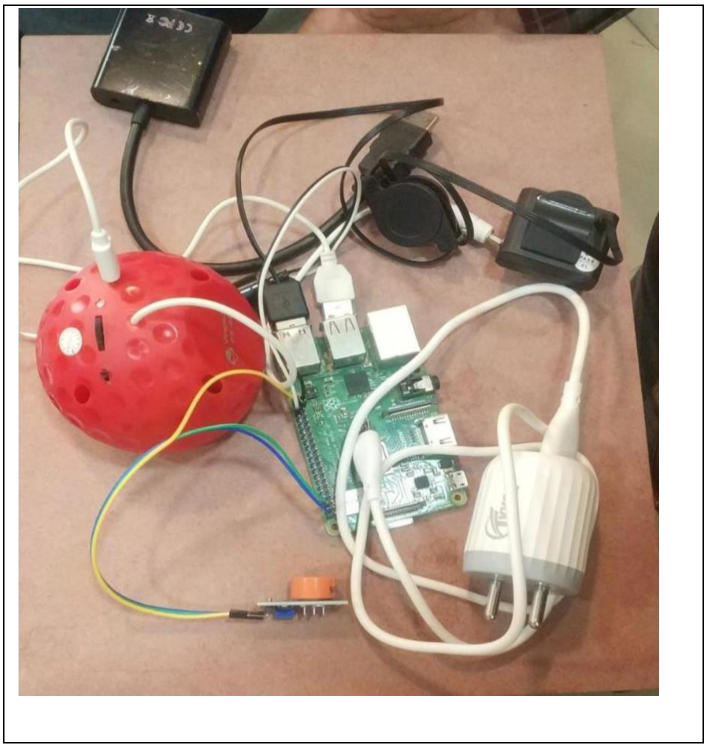


Fig 5. Hardware Components.

3. CONCLUSIONS

In this system, we proposed driver assistance System is presented in order to reduce the number of accidents caused by driver fatigue and thus improve road safety. This system treats the automatic detection of driver drowsiness based on visual information and artificial intelligence. We propose an algorithm to locate, track and analyze both the driver face and eyes to measure PERCLOS (percentage of eye closure) with Soft max for neural transfer function. We presented a vision-based method and system towards bus driver fatigue detection using existing dome Cameras in buses.

ACKNOWLEDGEMENT

We are highly indebted to our guide Prof. (Mrs) S.H.Barshikar for her guidance and constant supervision as well as for providing necessary information regarding the project report and also for her support in completing the project report. We would like to express our special gratitude and thanks to all the staff members of Department of Electronics and Telecommunication for giving us such attention and time. This acknowledgment would be incomplete without expressing our thanks to Head of the Department, Dr. M. B. Mali for his support during the work. We would like to extend our heartfelt gratitude to our Principal, Dr. S. D. Lokhande who provided a lot of valuable support, mostly being behind the veils of college bureaucracy.

We would also like to express our gratitude towards our parents and friends for their kind co-operation and encouragement which help us in the completion of this report. Our thanks and appreciations also go to our colleagues in developing this Project Report and people who have willingly helped us out with their abilities.

REFERENCES

[1] N Padhy, R Panigrahi, SC Satapathy," Identifying the Reusable Components from Component-Based System: Proposed Metrics and Model",- Information Systems Design and Intelligent application, pp. 89-99,2019.

[2] Padhy, N., Panigrahi, R. Neeraja, K. Threshold estimation from software metrics by using evolutionary techniques and its proposed algorithms, models, Evol. Intel. (2019). https://doi.org/10.1007/s12065-019-00201- 0.Charniya, N. N., Nair, V. R. (2017, June). Drunk driving and drowsiness detection. In 2017 International Conference on Intelligent Computing and Control (I2C2) (pp.1-6). IEEE.

[3] Ramesh, L., Monisha, M., Shirley Pradeeksha, A., Sowmiyaa, P., Vedhashree, S.K. (2018). Driver Drowsiness Detection and Alerting System. International Journal of Pure and Applied Mathematics, 118(20), 2247-2252.

[4] de Naurois, C. J., Bourdin, C., Stratulat, A., Diaz, E., Vercher, J. L. (2019). Detection and prediction of driver drowsiness using artificial neural network models. Accident Analysis Prevention, 126, 95-104.

[5] McDonald, A. D., Lee, J. D., Schwarz, C., Brown, T. L. (2018). A contextual and temporal algorithm for driver drowsiness detection. Accident Analysis Prevention, 113, 25-37.

[6] J. May and C. Baldwin, "Driver fatigue: The importance of identifying causal factors of fatigue when considering detection and countermeasure technologies," Transp. Res. F, Traffic Psychol. Behav., vol. 12, no. 3, pp. 218–224, 2009.

[7] E. Hitchcock and G. Matthews, "Multidimensional assessment of fatigue: A review and recommendations," in Proc. Int. Conf. Fatigue Manage. Transp. Oper., Seattle, WA, USA, Sep. 2005.