The Seybold Report
ISSN 1533-9211
DOI 10.5281/zenodo.6686505

DRIVER DROWSINESS DETECTION USING IMAGE PROCESSING

[1]T.NITHYA.M.E, [2]CHARLIN.D, [3]MANOJ.T, [4]MEENAKSHI.R and [5]SUGADEV.B

[1] Assistant Professor, Department of Computer Science and Engineering, RVS Technical Campus Coimbatore.
[2,3,4,5] Students, Department of Computer Science and Engineering, RVS Technical Campus Coimbatore.

ABSTRACT

Drowsy driving has played a significant role in a number of traffic incidents throughout the years. Implementing a system with an alarm output to inform tired drivers to focus on the road can help prevent car accidents and other undesired situations. As one strategy to minimize accidents, save money, and reduce losses and sufferings, an intelligent system is being developed to detect driver drowsiness and trigger an alarm to notify drivers. However, present approaches have several drawbacks due to the considerable fluctuation of surrounding conditions. Bad lighting can make it difficult for the camera to precisely measure the driver's face and eye. Due to late detection or no detection, this will have an impact on image processing analysis. Reduce the technique's precision and efficiency. Several strategies have been explored and analyses in order to determine the best technique for detecting driver tiredness with the highest accuracy. In this paper, we propose a real-time system that uses a computerized camera to follow and process the driver's eye using Python, d lip, and Open CV. The driver's eye region is continuously measured and calculated to assess drowsiness before an output alarm is triggered to alert the driver.

Keywords: Drowsiness detection, face detection and tracking, eye detection and tracking, eye aspect ratio, yawning, eye closure, computer vision, real time detection.

1. INTRODUCTION

One of the most common causes of accidents is driver drowsiness and weariness. Each year, the number of people killed in such accidents rises around the world. This project attempts to reduce the amount of accidents caused by driver weariness and drowsiness. As a result, transportation safety will improve. Driver drowsiness detection is a car technology that can help prevent accidents and save lives by detecting when a driver is becoming drowsy. This study employs computer vision to detect tiredness in drivers. Transportation modalities have advanced as a result of the ongoing progress and novelty in technology. There are a rising number of incidents on the roads nowadays, and driver tiredness is a significant contributing element, which has been widely recognized. The exact number of accidents caused by drowsy driving is difficult to determine because it is frequently underestimated. The change from being tired to nodding asleep is subtle, and most drivers are unaware of it. This explains why additional research is needed in this area in order to reduce the number of tiredness-related accidents and motivates us to build a driver drowsiness detection system.

The Seybold REPORT
ISSN 1533-9211
DOI 10.5281/zenodo.6686505

Problem statement

The sleepiness detecting system's goal is to help reduce accidents in both passenger and commercial vehicles. Before the driver has completely lost all attention, the system will detect the early signs of drowsiness and notify them that they are no longer capable of properly operating the car.

Purpose of study

The sleepiness detecting system's goal is to help reduce accidents in both passenger and commercial vehicles. Before the driver has completely lost all attention, the system will detect the early signs of drowsiness and notify them that they are no longer capable of properly operating the car. Drowsiness detection is a safety feature that can help prevent accidents caused by drivers who fall asleep behind the wheel. This intermediate Python project's goal is to create a sleepiness detection system that detects when a person's eyes are closed for a few seconds.

2. LITERATURE REVIEW

Author name: Belal alshaqaqi and Abdullah salem (2019)

Depiction: Drowsiness and Fatigue of drivers are amongst the significant causes of road accidents. Every year, they increase the amounts of deaths and fatalities injuries globally. In this paper, a module for Advanced Driver Assistance System (ADAS) is presented to reduce the number of accidents due to drivers fatigue and hence increase the transportation safety; this system deals with automatic driver drowsiness detection based on visual information and Artificial Intelligence. We proposed an algorithm to locate, track, and analyze both the drivers face and eyes to measure percentage of eye dosure, a scientifically supported measure of drowsiness associated with slow eye closure. © 2016, International Institute for General Systems Studies.

Author name: U.Ceerthi Bala And T.Sarath (2020)

Depiction: The rate of road accidents in today's world is very high. One of the most critical factors causing these accidents is human error. One such significant error is Drowsiness. Hence, Timely warning and precaution can aid in protecting lives. Many systems try to address this problem, by alerting the Driver; but fails to address the Traffic department that needs this information the most. Here, the proposed system aims to alert the Driver when in drowsy state and also the Traffic department, to control and evade any plausible accident. This is a non-intrusive system that is divided into 3 subsystems. The first system is a drowsiness detection unit that detects drowsiness. Once the drowsiness is detected, it sends the message via Controller Area Network (CAN) to the second subsystem; the Dashboard unit. This system cautions and alerts the driver. The system then sends a message indicating Drowsiness and vehicle's Position to the IoT platform using the third subsystem; the communication unit. This information is then sent to Road Transport Office (RTO) informing the position of the vehicle. This information can be used by the department to do the needful.

ISSN 1533-9211

DOI 10.5281/zenodo.6686505

Author name: G.Sikander and S.Anwarn (2019)

Depiction: Driver fatigue has been attributed to traffic accidents; therefore, fatigue-related traffic accidents have a higher fatality rate and cause more damage to the surroundings compared with accidents where the drivers are alert. Recently, many automobile companies have installed driver assistance technologies in vehicles for driver assistance. Third party companies are also manufacturing fatigue detection devices; however, much research is still required for improvement. In the field of driver fatigue detection, continuous research is being performed and several articles propose promising results in constrained environments, still much progress is required. This paper presents state-of-the-art review of recent advancement in the field of driver fatigue detection. Methods are categorized into five groups, i.e., subjective reporting, driver biological features, driver physical features, vehicular features while driving, and hybrid features depending on the features used for driver fatigue detection. Various approaches have been compared for fatigue detection, and areas open for improvements are deduced.

Author name: Chris Schwarz, john gaspar, Thomas miller &Reza yousefian (2019)

Depiction: Drowsiness is a major cause of driver impairment leading to crashes and fatalities. Research has established the ability to detect drowsiness with various kinds of sensors. We studied drowsy driving in a high-fidelity driving simulator and evaluated the ability of an automotive production-ready driver monitoring system (DMS) to detect drowsy driving. Additionally, this feature was compared to and combined with signals from vehicle-based sensors. Methods: The National Advanced Driving Simulator was used to expose drivers to long, monotonous drives. Twenty participants drove for about 4 h in the simulator between 10 p.m. and 2 a.m. They were allowed to use cruise control and traffic was sparse and semirandom, with both slower- and faster-moving vehicles. Observational ratings of drowsiness (ORDs) were used as the ground truth for drowsiness, and several dependent measures were calculated from vehicle and DMS signals. Drowsiness classification models were created that used only vehicle signals, only driver monitoring signals, and a combination of the 2 sources.

Author name: B.Mohana , C.M.Sheela Rani (2019)

Depiction: The number of major road accidents that occur per day is on a rise and most of them are attributed to being the driver's fault. According to the survey done in 2015, drivers are held responsible for approximately 78% of the accidents. To minimize the occurrence of these incidents a monitoring system that alerts the driver when he succumbs to sleep is proposed. This algorithm processes live video feed focused on the driver's face and tracks his eye and mouth movements to detect eye closure and yawning rates. An alarm sounds if the driver is drowsy or already asleep. Haar-cascade classifiers run parallelly on the extracted facial features to detect eye closure and yawning.

Author name: Aditya Ranjan , Karnan Vyas , Sujay Ghadge , Siddharth Patel , Suvarna Sanjay Pawar (2020)

ISSN 1533-9211
DOI 10.5281/zenodo.6686505

Depiction: A driver drowsiness detection system is proposed that involves detection of driver drowsiness by use of an algorithm. For detection of drowsiness, the most relevant visual indicators that reflect driver's condition is the eye behavior. The facial algorithm employed makes use of an eye aspect ratio and physical landmark measurements. Landmark detectors used in the algorithm demonstrate robustness against varied head orientations, facial expressions and lighting conditions. The proposed real time algorithm will estimate eye aspect ratio that measures eye open level in each video frame. It perceives eye blink pattern as EAR values. By doing so, potential drowsiness is detected. A large number of road mishaps occur due to drivers falling asleep due to exhaustion or long-haul driving and negligence. The proposed system under development can help prevent the same by providing non-invasive and easy to use specialized devices.

Author name: Rateb Jabbar, Mohammed Shinoy , Mohamed Kharbeche , Khalifa Al-Khalifa , Moez Krichen , Kamel Barkaoui.

Depiction: A sleepy driver is arguably much more dangerous on the road than the one who is speeding as he is a victim of microsleeps. Automotive researchers and manufacturers are trying to curb this problem with several technological solutions that will avert such a crisis. This article focuses on the detection of such micro sleep and drowsiness using neural networkbased methodologies. Our previous work in this field involved using machine learning with multi-layer perceptron to detect the same. In this paper, accuracy was increased by utilizing facial landmarks which are detected by the camera and that is passed to a Convolutional Neural Network (CNN) to classify drowsiness. The achievement with this work is the capability to provide a lightweight alternative to heavier classification models with more than 88% for the category without glasses, more than 85% for the category night without glasses. On average, more than 83% of accuracy was achieved in all categories. Moreover, as for model size, complexity and storage, there is a marked reduction in the new proposed model in comparison to the benchmark model where the maximum size is 75 KB. The proposed CNN based model can be used to build a real-time driver drowsiness detection system for embedded systems and Android devices with high accuracy and ease of use.

Author name: PA Harsha Vardhini, Venkat Kalyan Reddy Yasa, G Janardhana Raju

Depiction: Vehicles intentionally or unintentionally enter a restricted area and commonly termed as trespassing. Unauthorized entry offences have been common in most of the countries. In order to keep a check on this various techniques have been followed. This work illustrates the method to avoid trespassing through proper detection of vehicles. The gateway will be integrated with the camera which will detect the vehicle and detect the number plate. Using IoT the collected data sent to an online server is compared with the pre-existing data. If the number plate is an authorized one, gateway automatically opens for the vehicle to enter. If it is detected as unauthorized vehicle then the gate will remain closed and a buzzer alarm will be activated. Through the mobile app admin can add the number plate details to the device. The detection process is carried out by processing the images of the number plates captured by the image processing.

ISSN 1533-9211

DOI 10.5281/zenodo.6686505

Author name: Jasper S. Wijnands, Jason Thompson, Kerry A. Nice, Gideon D.P.A. Aschwanden, Mark Stevenson

Depiction: Driver drowsiness increases crash risk, leading to substantial road trauma each year. Drowsiness detection methods have received considerable attention, but few studies have investigated the implementation of a detection approach on a mobile phone. Phone applications reduce the need for specialised hardware and hence, enable a cost-effective roll-out of the technology across the driving population. While it has been shown that three-dimensional (3D) operations are more suitable for spatiotemporal feature learning, current methods for drowsiness detection commonly use frame-based, multi-step approaches. However, computationally expensive techniques that achieve superior results on action recognition benchmarks (e.g. 3D convolutions, optical flow extraction) create bottlenecks for real-time, safety-critical applications on mobile devices. Here, we show how depthwise separable 3D convolutions, combined with an early fusion of spatial and temporal information, can achieve a balance between high prediction accuracy and real-time inference requirements. In particular, increased accuracy is achieved when assessment requires motion information, for example, when sunglasses conceal the eyes. Further, a custom TensorFlow-based smartphone application shows the true impact of various approaches on inference times and demonstrates the effectiveness of real-time monitoring based on out-of-sample data to alert a drowsy driver. Our model is pre-trained on ImageNet and Kinetics and fine-tuned on a publicly available Driver Drowsiness Detection dataset. Fine-tuning on large naturalistic driving datasets could further improve accuracy to obtain robust in-vehicle performance. Overall, our research is a step towards practical deep learning applications, potentially preventing micro-sleeps and reducing road trauma.

3. EXISTING SYSTEM

The current technology uses an eye blink sensor to detect driver tiredness. We intend to make it mandatory to wear eye glasses by employing the Eye Blink Sensor. The eye-blink sensor works by illuminating the eye and/or eyelid area with infrared light, then utilising a phototransistor and differentiator circuit to monitor variations in the reflected light. The exact functioning is very dependent on how the emitter and detector are positioned and aimed in relation to the eye.

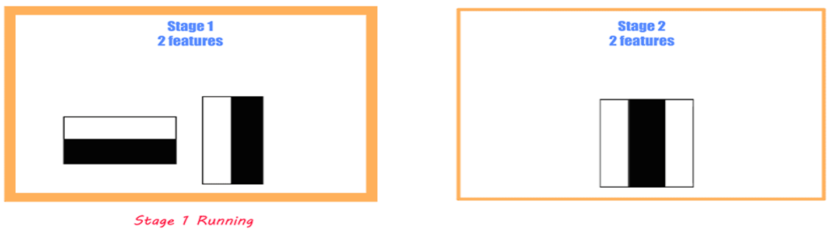
3.1 Disadvantages of existing system

- If we utilise an eye blink sensor to detect drowsiness, we must wear glasses.
- The EYE Blink Sensor is a physical component. As a result, we'll need to spend more money on gear.
- Work Efficiency of the Eye Blink Sensor is likewise low.
- EBS has a low accuracy rating.

ISSN 1533-9211
DOI 10.5281/zenodo.6686505

4. PROPOSED SYSTEM

Open CV is used to detect the face and eye in this project. One of the most effective technologies for detecting the face is Open CV. Yawn detected and alerted the driver with sound in our project. Python software was used to finish all of the projects. It's an Object Detection Algorithm that detects faces in images and real-time videos. Face detection is one of the first methods for locating a human face and returning a rectangle in x,y,w,h coordinates. Face landmark: Once we've found the location of a face in an image, we need to find points within that rectangle.



4.1 Open CV

Open Computer Vision is shortly denoted as Open CV. Officially launched in 1999 the Open CV project was initially an Intel Research initiative to advance CPU-intensive applications, part of a series of projects including real-time ray tracing and 3D display walls. The main contributors to the project included a number of optimization experts in Intel Russia, as well as Intel's Performance Library Team. Open CV is written in C++ and its primary interface is in C++, but it still retains a less comprehensive though extensive older C interface. All of the new developments and algorithms appear in the C++ interface. There are bindings in Python, Java and MATLAB. The API for these interfaces can be found in the online documentation. Wrappers in several programming languages have been developed to encourage adoption by a wider audience. In our project Open Computer Vision used to detect the face in our face recognition technique. Open CV is act like eye of the computer or any machines.

4.2 Cascade Classifier

It's a machine-learning approach in which a cascade function is learned using a large number of positive and negative photos. After then, it's utilised to find items in other photos. We'll be working on facial detection here. To train the classifier, the method requires a large number of positive images (images of faces) and negative images (images without faces). After that, we must extract features from it. The hear features illustrated in the graphic below are

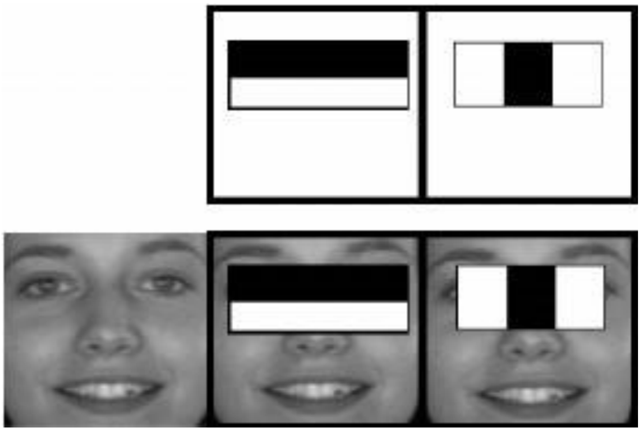
ISSN 1533-9211
DOI 10.5281/zenodo.6686505

employed for this. They're really similar to our convolution kernel. Each feature is a single value produced by subtracting the total of pixels in the white and black rectangles.

4.3 Features

- Many features are now calculated using all possible sizes and positions of each kernel. (Can you fathom how much computing power is required? A 24x24 window yields more than 160000 characteristics). We must find the total of pixels under white and black rectangles for each feature computation. They used integral pictures to address the problem. It makes calculating the total of pixels easier.
- To an action using only four pixels, no matter how great the number of pixels is. Isn't it lovely? It speeds up the process.
- However, the majority of the features we estimated are irrelevant.
- Take a look at the illustration below. Two good aspects are shown in the first row.
- The first feature chosen appears to be based on the fact that the area around the eyes is often darker than the area around the nose and cheeks; the second feature chosen is based on the fact that the eyes are darker than the bridge of the nose; however, the same windows could be used on the cheeks or anywhere else. So, how do we pick the best features from a list of 160000+ options? Adaboost is the one who accomplishes it.

4.4 Examples



We do this by applying each feature to all of the training photos. It calculates the appropriate threshold for each feature to identify the faces as positive or negative. However, there will undoubtedly be errors or misclassifications. We choose characteristics with the lowest error rate, which implies they're the ones that best distinguish between face and non-facial photos. (The procedure is not as straightforward as this.) At first, each image is assigned the same weight. The weights of misclassified photos are increased after each categorization. Then the

ISSN 1533-9211
DOI 10.5281/zenodo.6686505

process is repeated. Error rates are calculated at new levels. There are also new weights. The method is repeated until the requisite accuracy or error rate is met, or until the required number of features is discovered. The final classifier is a weighted sum of the classifiers. A weighted sum of these weak classifiers makes up the final classifier. It's named weak since it can't classify the image on its alone, but when combined with others, it creates a strong classifier. Even 200 characteristics, according to the report, yield 95 percent accuracy in detection. Around 6000 characteristics were included in their final arrangement. (Imagine a reduction in the number of features from 160000 to 6000.) That's a significant gain). So now you snap a photograph. Take a look at each 24x24 window. Apply 6000 characteristics to it. Check to see if it's a face or not. Isn't it a little inconvenient and time-consuming? Yes, it is correct. For this, authors have a decent solution. In an image, majority of the image region is non- face region. As a result, having a simple approach to check is a better notiona potential facial region They came up with the idea of a Cascade of Classifiers to do this. Instead of applying all 6000 characteristics to a single window, divide them into separate stages of classifiers and apply them one at a time.

(In most cases, the first few stages will include a small number of features.)

• If a window fails to pass the first stage, it should be discarded.

• We don't take into account any leftover features. Apply the second stage of features and continue the procedure if it passes.

• A face region is a window that passes through all stages. What's the status of the plan???

4.5 Facial landmarks

The Dlib package includes a free, pre-trained facial detector that can locate a total of 68 co-ordinates. A functional facial structure is mapped and developed using the 68 Co-Ordinates. By incorporating facial landmarks into the screenplay, the live camera can forecast and determine the position of the left and right eye, eye brows, nose, mouth, and jaw depending on the shape of the subject's head. In the case of sleepiness detection, eye location is critical and must be detected correctly in order for the script to appropriately identify the driver's drowsiness.

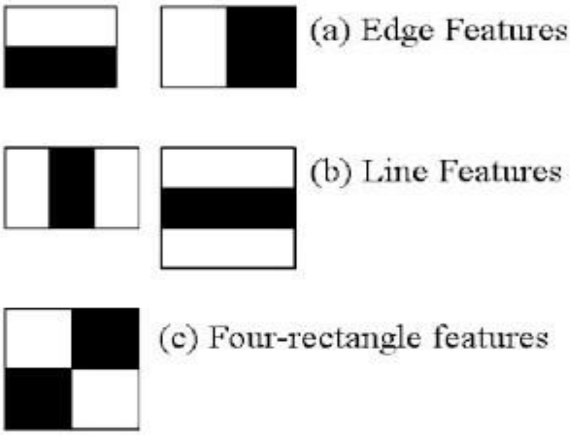
4.6 Advantages

- OPEN CV provides access to over 2,500 cutting-edge and traditional algorithms.
- Open CV provides an effective solution for detecting drowsiness.
- HAARCASCADE The accuracy of file detection is very high.
- TAKE LESS TIME IN TRAINING WITH AN OPEN CV

ISSN 1533-9211

DOI 10.5281/zenodo.6686505

4.7 Features



image

• Countless features are now calculated using all conceivable sizes and placements of each kernel. (Can you fathom how much computing power is required?

•A 24x24 window yields more than 160000 characteristics). We must find the total of pixels under white and black rectangles for each feature computation.

•They used integral pictures to address the problem. It makes calculating the total of pixels easier.

•To an action using only four pixels, no matter how great the number of pixels is. Isn't it lovely? It speeds up the process.

• However, the majority of the features we estimated are irrelevant.

•Take a look at the illustration below. Two good aspects are shown in the first row.

• The first attribute chosen appears to be the fact that the area around the eyes is generally darker than the area around the nose and cheekbones

• The second feature chosen is based on the fact that the eyes are darker than the nasal bridge. However, the application of the same windows to the cheeks or any other location is irrelevant. So, how do we pick the best features from a list of 160000+ options? Adaboost is the one who accomplishes it.

5. METHODOLOGY

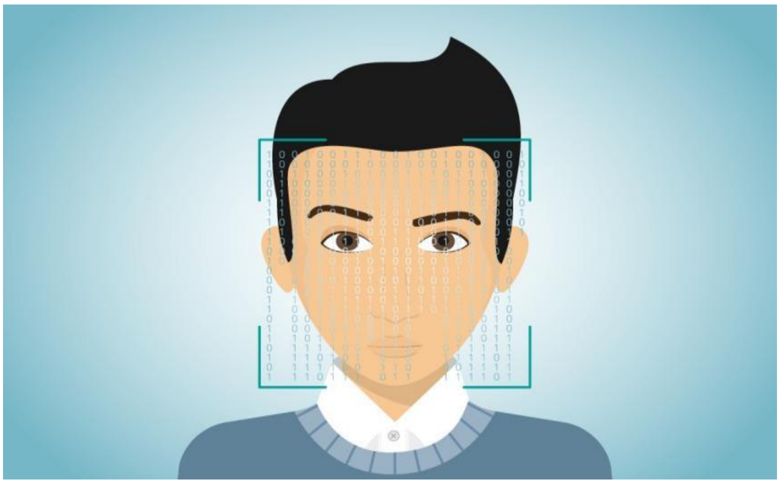
5.1 Face detection

Face detection is a computer technique that recognises human faces in digital images and is used in a range of applications. The psychological process through which humans locate and

ISSN 1533-9211

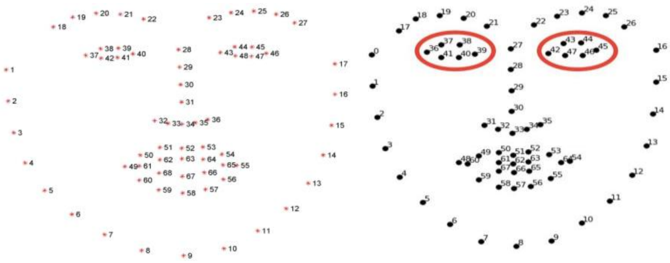
DOI 10.5281/zenodo.6686505

attend to faces in a visual context is known as face detection. Face detection is done using the Haar Cascade Classifier.



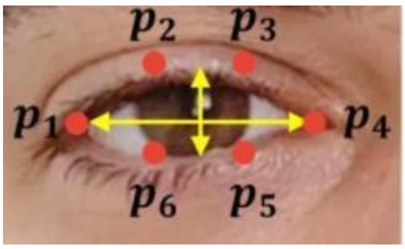
5.2 Eye detection

Face detection is one of the first methods for locating a human face and returning a rectangle in x,y,w,h coordinates. Face landmark: Once we've found the location of a face in an image, we need to find points within that rectangle. DLIB 68 was used to recognise face landmarks.



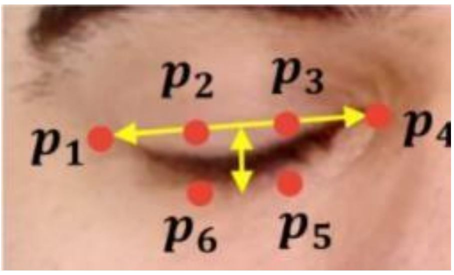
5.3 Ear (eye aspect ratio)

The driver's sleepiness level was calculated after facial landmarks were used to recognise the driver's face. Calculate the Eye Aspect Ratio and compare it to the Threshold Value. The driver is not notified if the EAR is higher than the threshold. Otherwise, it will sound an alarm to inform the motorist.



Open EYE will have more EAR

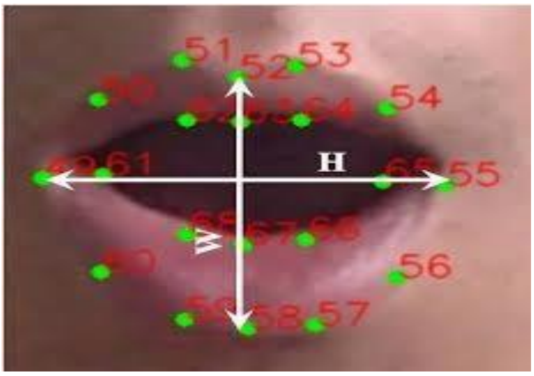
ISSN 1533-9211
DOI 10.5281/zenodo.6686505



Close EYE will have Less EAR

5.4 Yawn detection

The goal of this research is to use yawning detection to track driver fatigue levels. One of the most common causes of accidents is yawning. It detects the driver's top and lower lips in the yawn detection process. The distance between the upper and lower lips is then calculated. If the distance is less than our threshold value, no alarm will be triggered. Otherwise, an alarm will be sounded.



6. IMAGE PROCESSING

The process of identifying objects in a photograph. Image processing techniques such as noise removal would most likely be used first, followed by (low-level) feature extraction to detect lines, regions, and possibly areas with specific textures. Collections of these forms can be interpreted as single objects, such as cars on a road, boxes on a conveyor belt, or malignant cells on a microscope slide. Because an object might appear substantially different when viewed from multiple angles or under varied illumination, this is an AI problem. Another issue is determining which features belong to which object and which are backdrop, shadows, and so on. These functions are generally performed instinctively by the human visual system, but a computer requires skilled programming and a lot of processing capacity.

6.1 Binary image

A binary image is a computer image in which each pixel has just two possible values. A binary image is often made up of two colours: black and white, however any two colours can be used. The foreground colour is used for the object(s) in the image, while the background colour is used for the rest of the image.

ISSN 1533-9211
DOI 10.5281/zenodo.6686505

Binary images are sometimes known as two-level or bi-level images. This implies that each pixel is represented by a single bit (0 or 1). This idea is typically referred to as black and white, monochrome, or monochromatic, although it can also refer to images with only one sample per pixel, such as grayscale images.

6.2 Colour image

A (digital) colour image is a digital image in which each pixel contains colour information. Each pixel has a unique value that determines its colour appearance. This value is accompanied by three digits that represent the color's decomposition into the three primary hues of Red, Green, and Blue. This method can be used to represent any colour visible to the naked eye. A number between 0 and 255 represents the decomposition of a colour into its three primary colours. For instance, white will be R = 255, G = 255, and B = 255; black will be (R,G,B) = (0,0,0); and bright pink will be: (255,0,255).

6.3 Gray scale image

A grayscale image is a digital image in which each pixel's value is a single sample, meaning it only contains intensity information. Images of this type, often known as black-and-white, are made entirely of shades of gray(0-255), ranging from black(0) at the lowest intensity to white(255) at the highest intensity. Grayscale images are distinct from one-bit black-and-white images, which are images with only two colours, black and white, in the context of computer imaging (also called bi-level or binary images). There are various shades of grey in grayscale photographs. Grayscale images are sometimes known as monochromatic images since they lack any chromatic variation. Grayscale images are frequently the result of detecting the intensity of light at each pixel in a single band of the electromagnetic spectrum (e.g., infrared, visible light, ultraviolet, etc.), and thus are monochromatic proper when just one frequency is captured in such circumstances. They can also be created from a full-color image; read the section on converting to grayscale for further information.

7. SOFTWARE SPECIFICATION

7.1 Python:

Python is a high-level, interpreted programming language that may be used for a variety of tasks. Python was created by Guido van Rossum and initially released in 1991. Its design philosophy emphasises code readability, with a lot of whitespace. It has structures that allow for clear programming at both small and large sizes. Van Rossum was the language community's leader until July 2018, when he stepped down.Python has a dynamic type system and memory management that is automated. It features a big and extensive standard library and supports several programming paradigms, including object-oriented, imperative, functional, and procedural.For a wide range of operating systems, Python interpreters are available. CPython, the standard Python implementation, is open source software with a community-driven development strategy, as are practically all of Python's other implementations. The Python and CPython projects are overseen by the Python Foundation.

ISSN 1533-9211
DOI 10.5281/zenodo.6686505

7.2 Indentation:

To delimit blocks, Python employs whitespace indentation rather than curly brackets or keywords. Certain statements are followed by an increase in indentation, while the current block is ended by a drop in indentation. [58] As a result, the visual organisation of the programme accurately reflects the semantic structure of the programme. The off-side rule is another name for this feature.

7.3 Windows 10

To delimit blocks, Python employs whitespace indentation rather than curly brackets or keywords. Certain statements are followed by an increase in indentation, while the current block is ended by a drop in indentation. [58] As a result, the visual organisation of the programme accurately reflects the semantic structure of the programme. The off-side rule is another name for this feature.

8. SYSTEM IMPLEMENTATION

Python IDLE has a full-featured file editor that allows you to write and run Python programmes directly from within the software. The built-in file editor also has various features that will help you code faster, such as code completion and automatic indentation. An integrated development environment (IDE) for Python is IDLE (Integrated Development and Learning Environment). The IDLE module is included by default in the Python installer for Windows. IDLE can be used to execute a single statement in the same way as Python Shell does, as well as to create, modify, and run Python scripts.

9. CONCLUSION

To sum it up Driver Drowsiness Detection is a safety feature that aids in the prevention of accidents caused by drowsy drivers. It's critical to recognise and inform the driver as soon as possible, before any potentially fatal incidents occur. The suggested system uses an image processing technique to calculate and measure the Eye Aspect Ratio, or the size of the driver's eyes, in order to detect driver drowsiness. Data on EAR value was collected and compared to a threshold value that indicates whether a driver is drowsy. If the EAR is less than the threshold, the ALARM module will inform the driver.

10. FUTURE ENHANCEMENTS

For the Realtime Driver Drowsiness Detection mechanism in the proposed system, Open CV is used. Face detection was done using the Haar Cascade Classifier.

• Based on 68 facial landmarks, face landmark 68 was used for eye and mouth detection. Only under high lighting power does our proposed solution perform successfully.

The Seybold REPORT

ISSN 1533-9211

DOI 10.5281/zenodo.6686505

• In low-light situations, it may not function properly. A face is only detected in a straight direction by the Haarcascade classifier.

•In the future, we will use a high-resolution camera for real-time processing.

• Our faces are not always correctly detected.We will employ advanced methods or files in the future to recognise a face and tiredness in realtime.

REFERENCES

1. Aditya Ranjan, Karan Vyas, SujayGhadge, Siddharth Patel, Suvarna Sanjay Pawar, “Driver Drowsiness Detection System Using Computer Vision and Rpi.”, in International Research Journal of Engineering and Technology(IRJET), 2020
2. BelalAlshaqaqi and Abdullah Salem, "Vision Based System for Driver Distaction", IEEE International conference on Innovations, 2019.
3. U. CeerthiBala and T. Sarath, "Internet of things based Intelligent Drowsiness Alert System", Fifth International Conference on Communication and Electronics Systems (ICCES 2020), pp. 36-36, 2020.
4. Chris Schwarz, John Gaspar, Thomas Miller & Reza Yousefian, “The detection of drowsiness using a driver monitoring system” , in Journal of Traffic Injury Prevention (Taylor and Francis Online), 2019.
5. R. Jabbar, M. Shinoy, M. Kharbeche, K. Al-Khalifa, M. Krichen and K. Barkaoui, "Urban Traffic Monitoring and Modeling System: An IoT Solution for Enhancing Road Safety", 2019 International Conference on Internet of Things Embedded Systems and Communications IINTEC 2019 - Proceedings, pp. 13-18, 2019.
6. Jasper S. Wijnands, Jason Thompson, Kerry A. Nice, Gideon D. P, Aschwanden& Mark Stevenson, “Real-time monitoring of driver drowsiness on mobile platforms using 3D neural networks”, Neural Computing and Applications, 2019.
7. B.Mohana, C.M.Sheela Rani, “Drowsiness Detection Based on Eye Closure”, in International Research Journal of Engineering and Technology (IRJET), 2019.
8. Rahul AtulBhope, “Computer Vision based drowsiness detection for motorized vehicles with Web Push Notifications”, IEEE 4th International Conference on Internet of Things, IEEE, Ghaziabad, India, 2019.
9. G. Sikander and S. Anwar, "Driver Fatigue Detection Systems: A Re-view", IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 6, pp. 2339-2352, June 2019.
10. P. A. H. Vardhini, V. K. R. Yasa and G. J. Raju, "Raspberry Pi Vehicle Gateway System with Image Processing based Authorization Detection using IoT", 2021