PROJECT REPORT

PILE-UP Shielding

GROUP VII

Date:- 13-Dec-2020

Team Members:-

Disha Gupta (LIT2018026)

Saakshi Jain (LIT2018075)

Submitted To:- Dr. Vishal Krishna Singh.

Introduction:

Driving a car is a complex task, and it requires complete attention. Distracted driving is any activity that takes away the driver's attention from the road. Several studies have identified three main types of distraction: visual distractions (driver's eyes off the road), manual distractions (driver's hands off the wheel) and cognitive distractions (driver's mind off the driving task).

The National Highway Traffic Safety Administration (NHTSA) reported that 36,750 people died in motor vehicle crashes on an average, and 12% of it was due to distracted driving.

Many states now have laws against texting, talking on a cell phone, and other distractions while driving. We believe that computer vision can augment the efforts of the governments to prevent accidents caused by distracted driving (as shown in figure). Our algorithm automatically detects the distracted activity of the drivers and alerts them. We envision this type of product being embedded in cars to prevent accidents due to distracted driving.This project aims to recognize unsafe behaviour and send real time feedback to the driver using short sound alert etc.The system can be used to classify the behaviour of a driver via live video feed.The system uses various python libraries for extracting features in the training phase of the warning system

---

Problem Description:

Distracted Driving, a term which is coined for doing any kind of activity that can divert your attention away from the task of safe driving, including talking or texting on your phone, eating and drinking, talking to people in your vehicle, fiddling with the stereo, entertainment or navigation system. In this report, a module for Advanced Driver Assistance System (ADAS) is presented to reduce the number of accidents due to drivers' distractions and fatigue and hence increase the transportation safety. This system deals with automatic driver drowsiness detection based on visual information and Artificial Intelligence. We propose an algorithm to locate, track, and analyze real time video, both the drivers face and eyes using computer vision. This project will help us to create a better place where safe driving can happen and no more deaths are there.

Proposed Solution

The proposed algorithm works in seven stages as described by Figure :

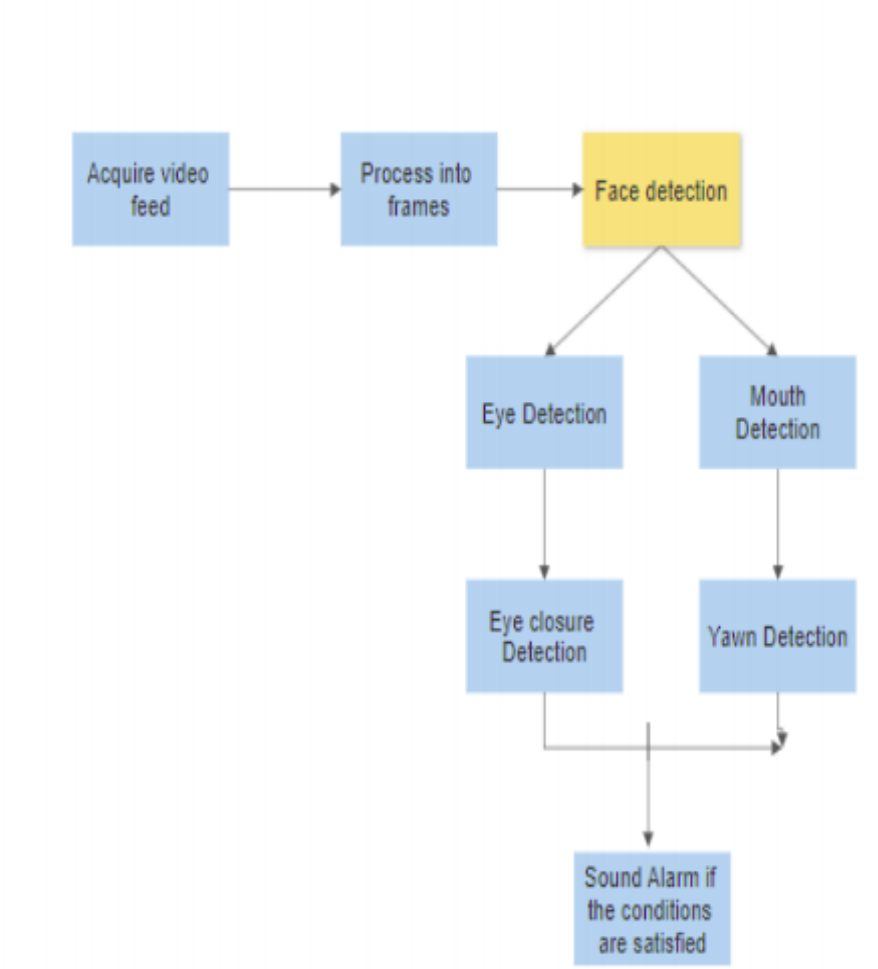


Fig 1 **Process flow of the proposed algorithm.**

To overcome the problem we came up with the solution implemented using OpenCV, DLib and other open source libraries . Python is used as a language to implement the idea.

A web camera is used to continuously track the facial landmark and movement of eyes and lips of the driver. This project mainly targets the landmarks of lips and eyes of the driver. For detection of drowsiness, landmarks of eyes are tracked continuously. If the driver is found to be distracted then a voice (audio) alert is provided and a message is displayed on the screen. Following use cases are covered in this project

1. If the eyes of drivers are closed for a threshold period of time then it is considered that the driver is feeling sleepy and a corresponding audio alarm is used to make the driver aware.

fig2 Driver is feeling sleepy (Yawning)

2. If the mouth of the driver remains open for a certain period of time then it is considered that the driver is yawning and corresponding suggestions are provided to the driver to overcome drowsiness.

3. If the driver doesn't keep eyes on the road then it is observed using facial landmarks and the corresponding alarm is used to make the driver aware.

Fig3 Driver gets distracted

All this functionality is then implemented with the help of python libraries, an espeak compact open source software is used to produce alert sound.

Requirement Specification:

The code of this system has been written in python mainly. For the smooth functioning of this system, some libraries and software must be there in the user Operating system priorly.

SOFTWARE REQUIREMENTS

Operating system - Windows is used as the operating system as it is stable and supports more features and is more user friendly

Dependencies

- Python 3
- opencv (tested with 3.4)
- dlib (tested with 19.18.0)
- imutils (tested with 0.5.3)
- scipy
- numpy
- Argparse
- Espeak (compact open source software)

HARDWARE REQUIREMENTS :

Intel core i5 2nd generation is used as a processor because it is faster than other processors and provides reliability and stability and we can run our pc for a long time.

By using this processor we can keep on developing our project without any worries.

Ram 1 gb is used as it will provide fast reading and writing capabilities and will in turn support in processing

Run

Python3 drowsiness_yawn.py -- webcam

//For external webcam, use the webcam number accordingly

Setups

Change the threshold values according to your need

EYE_AR_THRESH = 0.3

EYE_AR_CONSEC_FRAMES = 30
YAWN_THRESH = 10` //change this according to the distance from the camera

Feasibility Study:

Distracted driving is a leading cause of road traffic crashes the world over. One of the most common causes is the use of mobile phones while driving, whether that's talking on a phone, texting, checking your social media profiles or simply browsing, the risks of crashing rise dramatically. Our objective is to solve real-life problems and are working to reduce driver distraction, which is one of the leading causes of motor vehicle crashes.

Technical Feasibility

This assessment focuses on the technical resources available to the organization. It helps organizations determine whether the technical resources meet capacity and whether the technical team is capable of converting the ideas into working systems. Technical feasibility also involves the evaluation of the hardware, software, and other technical requirements of the proposed system.

This technology provides a new platform to detect operator drowsiness and has the potential to reduce distraction-related crashes in driving and aviation. The proposed distracted detector system developed and expected to reduce the road accident from doing any activity that takes your attention from driving. Focusing on truck transportation providers -risk groups for distraction.

SYSTEM BENEFITS FOR DRIVERS

- Warns the driver of distracted driving and the risk of a microsleep
- Compliance with driver warnings helps to avoid crashes caused by fatigue

Economic Feasibility

This assessment typically involves a cost/ benefits analysis of the project, helping organizations determine the viability, cost, and benefits associated with a project before financial resources are allocated. It also serves as an independent project assessment and enhances project credibility—helping decision-makers determine the positive economic benefits to the organization that the proposed project will provide.

However, existing distraction detection systems (e.g. eye-tracking and EEG) are often limited by the price, intrusiveness,and practicality. Several of the current distraction detection methods require purchasing special equipment and placing sensors on the driver.

The model presented in this paper will be a low budget system and all the hardware technology you require is a built-in camera which can be placed in front of the driver. The user needs no extra knowledge or specialisation in any form to use this system.

Legal Feasibility

This assessment investigates whether any aspect of the proposed project conflicts with legal requirements like zoning laws, data protection acts or social media laws. A feasibility study might reveal the organization's ideal location isn't zoned for that type of business.

Operational Feasibility

This assessment involves undertaking a study to analyze and determine whether—and how well—the organization's needs can be met by completing the project. Operational feasibility studies also examine how a project plan satisfies the requirements identified in the requirements analysis phase of system development.

A sensitive and user friendly distracted detection system can reduce the risk by issuing early warnings to alarm the drivers and make them aware that they are putting themselves in danger. This is particularly important for professional drivers (e.g.truck, bus and taxi drivers) who are quite prone to falling asleep behind the wheel, due to long driving hours and overconfidence about their driving skills.

Software Process Model Followed for the Development

In the implementation and development of our system, we are following both a plan-driven and incremental or iterative approach (mix of both the approaches). **In a plan-driven incremental approach, the system increments are identified in advance**, and so for the designing of this detection system, **both the basic structure and the sequential set of activities are well-defined**. We have already defined the basic skeleton and the design solution for the given problem but later on more features could be added. The model preparation section is decomposed into many small parts that can be incrementally developed and each incremental part is developed over an iteration. At a time one iteration is planned, developed and deployed to the customers.

Incremental development is based on the idea of developing an initial implementation, exposing this to user feedback, and evolving it through several versions until an acceptable system has been developed.

The activities of the process are not separated but interleaved with feedback involved across those activities.

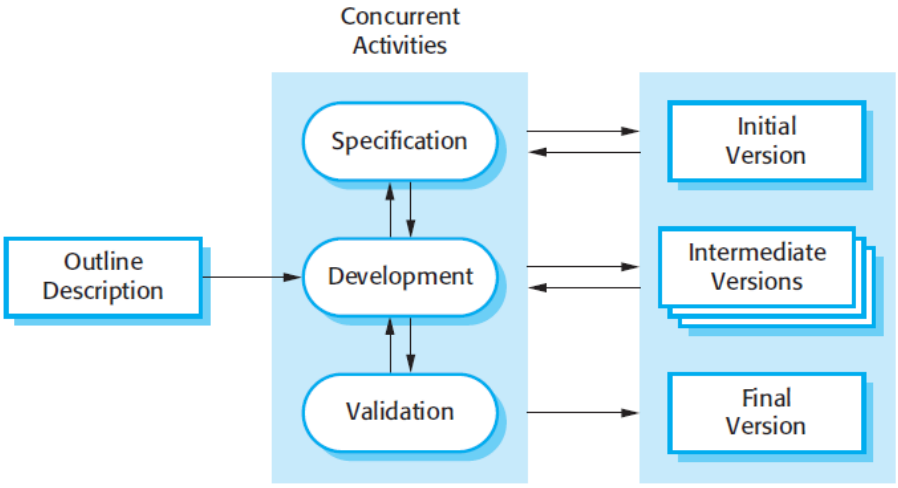


Fig-The Incremental Development Model

Detailed Description of the methods followed for different components of the model

Each system increment reflects a piece of the functionality that is needed by the customer. Generally, the early increments of the system should include the most important or most urgently required functionality.

This means that the customer can evaluate the system at an early stage in the development to see if it delivers what's required. If not, then only the current increment has to be changed and, possibly, new functionality defined for later increments.

Stage 1-

Outline Description- This project aims to recognize unsafe behaviour and send real time feedback to the driver using short sound alert etc.The system can be used to classify the behaviour of a driver via live video feed.

The rest of things like motivation and scope of this project is included above in this report.

Stage 2-

Software Specification- This defines the main functionalities of the software and the constraints around them.

Scipy-

SciPy stands for Scientific Python. SciPy is a scientific computation library that uses NumPy underneath. It provides more utility functions for optimization, stats and signal processing.

Imutils-

Series of convenience functions to make basic image processing functions such as translation, rotation, resizing, skeletonization, and displaying Matplotlib images easier with OpenCV and Python.

OpenCV

Open Source Computer Vision Library is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in commercial products.

Using OpenCV, we can read an image, display it and after we are done with our operations for the day, save it.

Facial Landmarks-

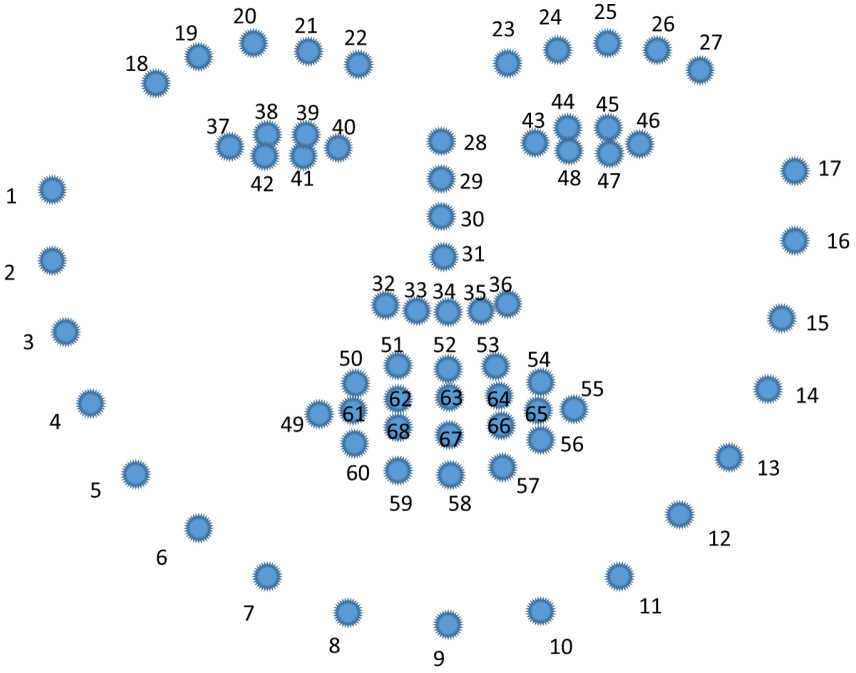
Facial landmarks are used for localizing and representing salient regions or facial parts of the person's face, such as:

Facial Landmarks Detection has 2 steps:

1. It involves localizing the face in the image.
2. To detect the key facial structures on the person's face.

DLIB-

The facial landmark detector which is pre-trained inside the dlib library of python for detecting landmarks, is used to estimate the location of 68 points or (x, y) coordinates which map to the facial structures. These indexes of 68 coordinates or points can be easily visualized on the image below:



Examining the image, we can see that facial regions can be accessed via simple Python indexing

- The mouth can be accessed through points [48, 68].
- The right eyebrow through points [17, 22].
- The left eyebrow through points [22, 27].
- The right eye using [36, 42].
- The left eye with [42, 48].
- The nose using [27, 35].
- And the jaw via [0, 17].

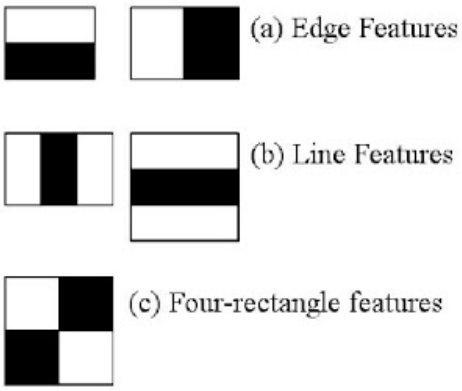
These mappings are encoded inside the ***FACIAL_LANDMARKS_IDXS*** dictionary inside ***face_utils of the imutils library.***

Stage 3-

Software Design and Development- This stage deals with the software that is to be designed and programmed.

Haar Cascade

Object and Face Detection using Haar feature-based cascade classifiers is an effective object detection method proposed by Paul Viola and Michael Jones in their paper, "Rapid Object Detection using a Boosted Cascade of Simple Features" in 2001.



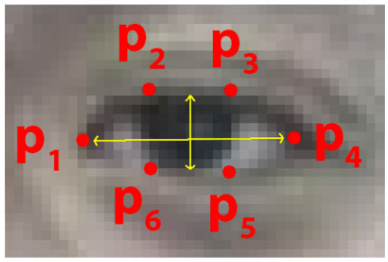
It is a machine learning based approach where a cascade function is trained from a lot of positive and negative images.

The algorithm has four stages:

1. Haar Feature Selection
2. Creating Integral Images
3. Adaboost Training
4. Cascading Classifiers

Eye-Closure Detection-

Each eye is represented by 6 (x, y)-coordinates, starting at the left-corner of the eye (as if you were looking at the person), and then working clockwise around the eye:



Eye Aspect Ratio-

For eye blinks we need to pay attention to facial landmarks points 37-46, the points that describe the eyes.

In *Real Time Eye Blinking Using Facial Landmarks*[2], Soukupová and Čech derive an equation that represents the Eye Aspect Ratio. The Eye Aspect Ratio is an estimate of the eye opening state.

The Eye Aspect Ratio is a constant value when the eye is open, but rapidly falls to 0 when the eye is closed.

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

Stage 4-

Software Validation-The software must conform to it's specification and meets the customer needs.

We have tested our software on multiple factors- like eye closure, yawing, fatigue, distraction while talking and distraction caused by cellphones and it gives accurate results in 90% of cases. The validation part can be found in the result section of this report.

Incremental development has three important benefits, compared to the other model:

1. The cost of accommodating changing customer requirements is reduced. The amount of analysis and documentation that has to be redone is much less than is required with the waterfall model.

2. It is easier to get customer feedback on the development work that has been done. Customers can comment on demonstrations of the software and see how much has been implemented. Customers find it difficult to judge progress from software design documents.

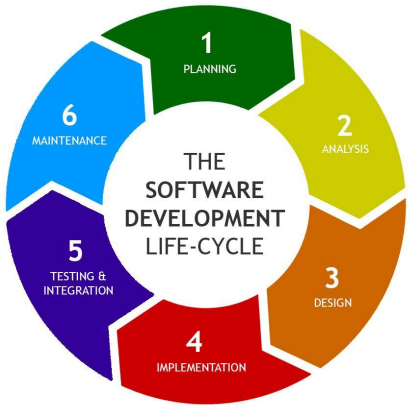
3. More rapid delivery and deployment of useful software to the customer is possible, even if all of the functionality has not been included. Customers are able to use and gain value from the software earlier than is possible with a waterfall process.

Some other benefits associated with plan driven approach software are mentioned below-

- A closer match to users' real needs.
- Improved design quality.
- Improved maintainability.
- Reduced development effort.

Complete Software Development Lifecycle Followed for the Development

SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.



A typical Software Development Life Cycle consists of the following stages −

Stage 1: Planning : We were planning to recognize unsafe behaviour and send real time feedback to the driver using short sound alert etc.The system can be used to classify the behaviour of a driver via live video feed.The system uses various python libraries for extracting features in the training phase of the warning system. We proposed an algorithm to locate, track, and analyze real time video, both the drivers face and eyes using computer vision. This project will help us to create a better place where safe driving can happen and no more deaths are there.

Stage 2: Requirement Analysis:-

Already described above in Requirement Specification(Point 4).

Stage 3: Design:-

We use OpenCV library for the software GUI. Open Source Computer Vision Library is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of

machine perception in the commercial products.
Using OpenCV, we can read an image, display it and after we are done with our operations for the day, save it.

Stage 4:- Implementation or Coding:-

The subject is placed in front of the camera so that the face is in focus. The window displays the plotted landmarks, yawn count and a message of the subject's state. It is expected that An alarm will be sounded when the yawn count exceeds the set boundary parameter. Alarm is also sounded if the eyes are found to be closed. When the subject found distracted then message along with an alarm being activated.

Step1: Firstly The video is obtained from a camera focused on the driver's face. The processing rate of the acquired video is 30 frames per second.
Step 2: These frames are then flipped and converted to grayscale.
Step 3: Then it detects face , if face is detected then it detects mouth and eyes of the driver,
Step 4: If everything go well and any distracted activity is distracted then sound or alert is produced

Stage 5: Testing:-

We have tested the software on different types of distraction and different different situations and it gives accurate results in 90% of cases.

Stage 6: Maintenance:-

We need to update the libraries which are used in this system in a definite interval of time.For the smooth functioning of this system, some libraries and software must be always there in the user Operating system with their proper version .

Results and Conclusion:

The subject is placed in front of the camera so that the face is in focus. The window displays the plotted landmarks, yawn count and a message of the subject's state. It is expected that An alarm will be sounded when the yawn count exceeds the set boundary parameter. Alarm is also sounded if the eyes are found to be closed.

When the subject is detected to have closed eyes or using phone while driving the message "Distracted ALERT" along with an alarm being activated. Fig. 2. depicts a subject yawning. The detection is acknowledged by messages "Yawn ALERT" and Yawn count value.

These pictures show the result of the software:

Distracted driving is a major problem leading to a striking number of accidents worldwide. Its detection is an important system component in semi autonomous cars. The proposed algorithm is expected to detect drowsiness in a driver using eye closure and yawning as markers. At last The system will be tested on a variety of subjects with a limited number of test cases.