# FINAL YEAR PROJECT
# Driver's Fatigue Level Detection


**Syed Salman Haider (salman.haider@khi.iba.edu.pk)**
**Hiba Haider (hhaider@khi.iba.edu.pk)**
**Fatima Zain Allarakha (fallarakha@khi.iba.edu.pk)**

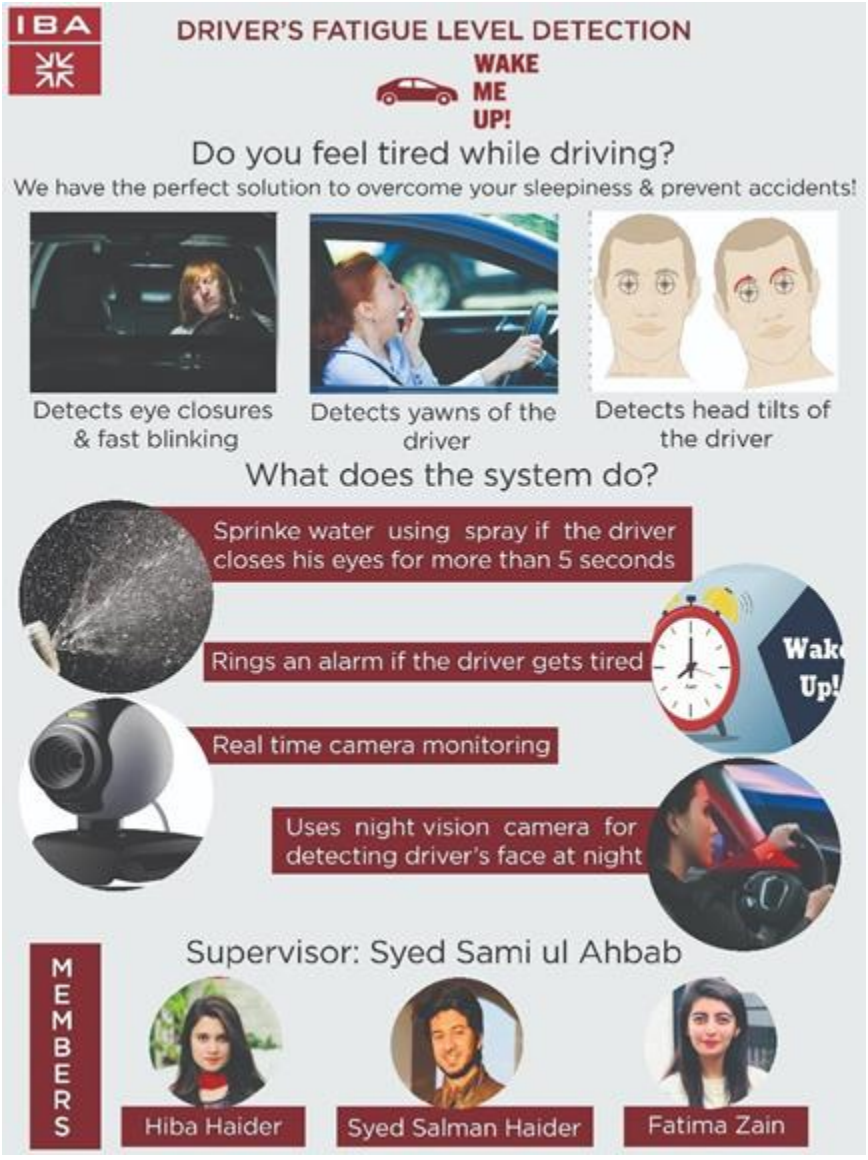
## Introduction:

Road accidents are suspected to be a primary concern nowadays and the key cause is the fatigue level of the driver which is a subject of intense research today. However, effective driver fatigue monitoring in environments with little or no light has been thought-provoking since the state of the driver is hard to identify. The automobile business also has tried to build several systems to predict driver drowsiness but there are only a few commercial products available today. The systems oversee driver capability and features but do not focus at driver's performance.

For this our project emphases on building a facial recognition system for driver sleepiness which can be due to long hours, shift work, lack of sleep or sleep apnoea that may exhaust the driver and cause him/her to sleep while driving. It will include a dashboard mounted camera that will be able to detect tiredness level of the driver in real time video stream and then play a sound with vibration using warning. The system will track driver's facial features like fast blinking, eye closures, head poses and yawning. We will try to achieve accuracy during the day, at night and under illuminated conditions through night vision camera.

Classifiers like Haar Cascade Classifiers or HOG + linear SVM will be chosen depending on their accuracy and shortest time of computation. EAR (eye aspect ratio) and facial expressions will be used to sense the tiredness of the car driver which will stimulate the system to ring alarms and send warnings consequently. The technical facets of the project will rely on OpenCV

and python for facial recognition with Raspberry pi for hardware port connections.

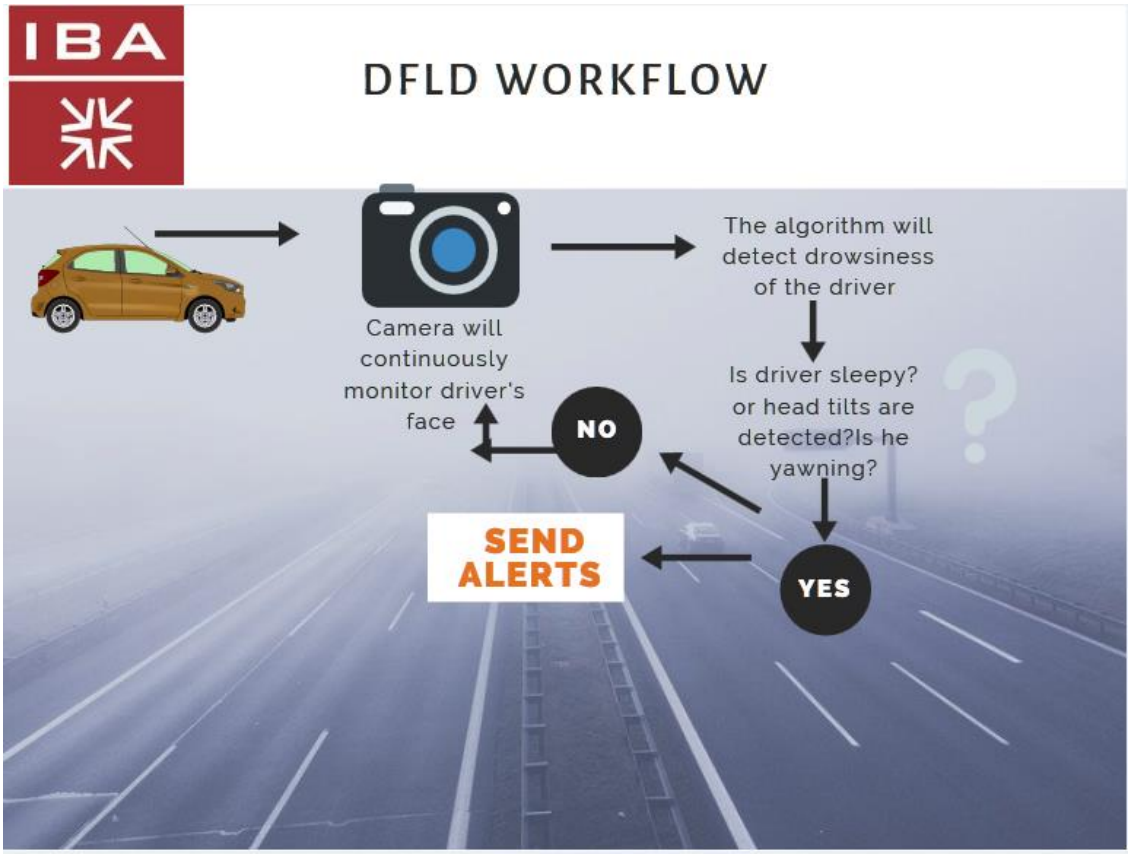


## High Level Design:

1. **Rationale and sources of your project idea:**

The project “Driver’s Fatigue Level Detection System (DFLD)” was designed and developed to easily facilitate, monitor and help detect when a driver may be exhibiting diminished. Our system keeps an eye on the driver’s behaviour to identify any signs that the driver is losing concentration to the point where safety is being compromised. Our rationale was to create a system in a way to avoid accidents risks which are caused due to long hours, shift work, lack of sleep or sleep apnoea that may exhaust the driver and cause him/her to sleep while driving. Our aim was to make our system efficient and less costly.

## Logical structure:



## Background:

The rise in road accidents has caused frequent number of disastrous expiries not only in Pakistan but a lot more around the world which is a serious matter of study currently. According to a research paper (December- 2014) in www.researchgate.net, According to World Health Organization (W.H.O) more than 1.3 million people die every year and many more are injured in Road Traffic Crashes (RTC). Fatality rate due to RTC in Pakistan is significantly higher as compared to other countries. It has been assessed that about 90 percent accidents in Pakistan occur due to the mistake of drivers. Which is why it is extremely necessary for the driver to take all possible safety mechanisms to avoid such incidents to happen as the death of one person can have an emotional impact on the lives of the entire family. Hence, to overcome this issue our project focuses on making a "Driver's fatigue level detection system" which is a safety technology that can help the driver awakened in uncertain conditions. Our system will be able to detect drowsy driving patterns like eye closures, blinking, yawning and head postures and alert drivers to not to sleep or take a break as prevention is better than cure. A lot of well-known automobiles companies have tried making this system but there are very less commercial systems available today. The major challenge we have here is to make a system that can work in little or no light which we will try to deal with through our project.

## Hardware / Software Trade-Offs:

- **HOG + Linear SVM:**

  HOG which stands for histogram of oriented gradients and linear support vector machines can be used in conjunction as a classifier. HOG will use a positive sample P from the training data of the objects that needed to be detect and then it will extract HOG descriptors from these samples. Later it will sample N negative samples and extract HOG descriptors from these samples as well. Finally it will train Linear support vector machines on the positive and negative samples to detect facial features. HOG along with facial landmark predictor will be used to detect facial landmarks from that we can extract the eye region then calculate EAR (eye aspect ratio) using the equation:
  EAR =||p2-p6 ||+ ||p3-p5 || / 2||p1-p4 ||

- **HAAR CASCADES:**

  Haar cascade is a machine learning algorithm to identify the objects in images. It is another approach as opposed to another approach of using HOG+linear SVM. It is a faster but usually slight less accurate than HOG. Haar cascade will be applied to detect face in an image which will help to detect the coordinates surrounding the face in an image. Later facial landmark predictor will be used to

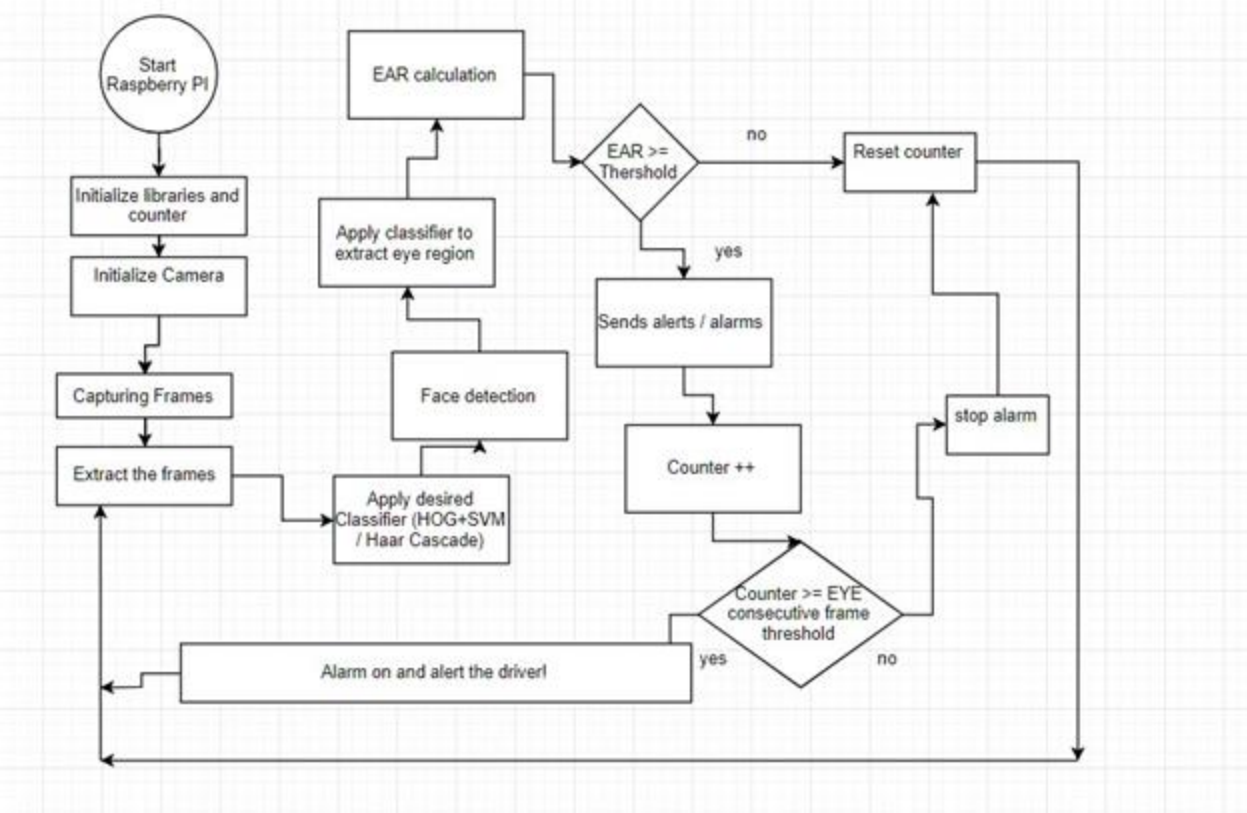
extract facial landmarks after that EAR will be calculated with the equation as discussed above. Both techniques are there to be followed. Haar cascades are usually faster but less accurate than HOG+ linear svm classifier, so we'll perform tests on both of algorithm then select the best classifier.



## Software / Hardware Design:

1. **Overview:**

   Our program will be running on Raspberry Pi. A python script along with OpenCV will be used. Different set of libraries would be used, like numpy, OpenCV, play sound, argparse, dlib, distance, timer, client, ApiClient, and picamera. A camera along with the buzzer for alarm and water sprinkler will be connected to raspberry Pi for continuous monitoring of driver's fatigue level detection.


Start Raspberry PI
Initialize libraries and counter
Initialize Camera
Capturing Frames
Extract the frames
Apply desired Classifier (HOG+SVM / Haar Cascade)
Face detection
Apply classifier to extract eye region
EAR calculation
EAR >= Thershold
no
Reset counter
yes
Sends alerts / alarms
Counter ++
Counter >= EYE consecutive frame threshold
yes
no
Alarm on and alert the driver!
stop alarm

## User interface:


- This is how it indicates with a sound when a person closes his/her eyes.
- It also sends alerts when person is yawning as shown above.

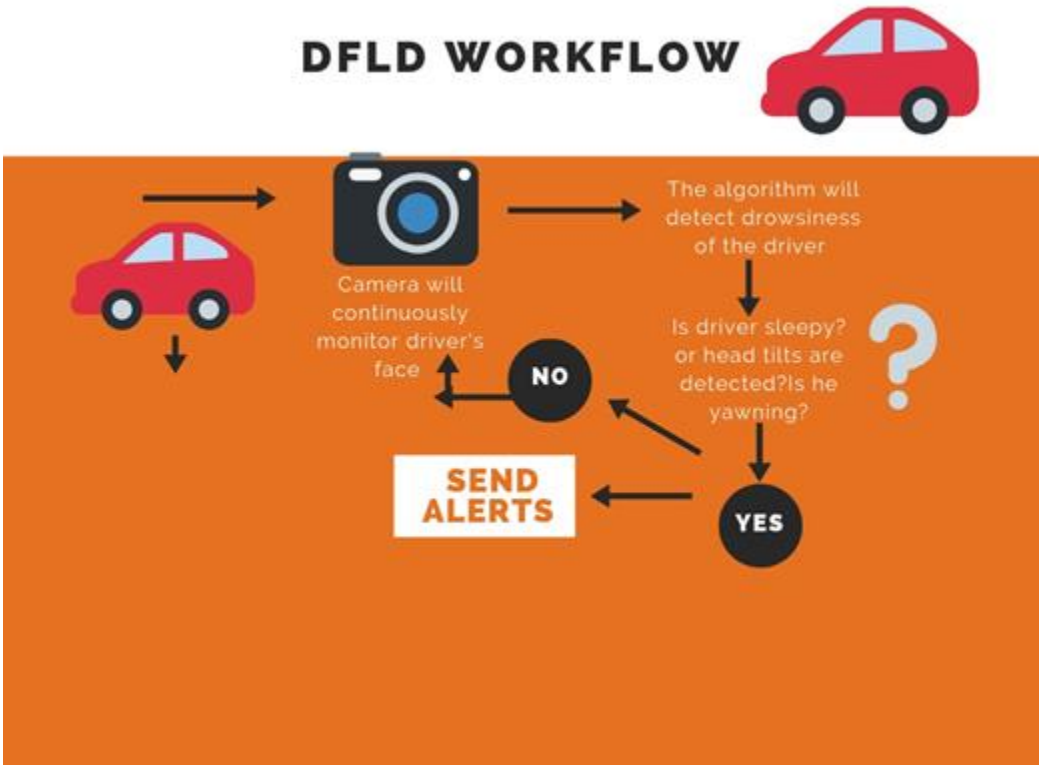

- It also detects head tilts and sends alerts accordingly.
- It works in night as well as shown above by using night vision configurations.

### Errors:

No errors found.

## Trials and tests:

1. Checking if the face detection algorithms are accurate and fast
2. Checking functionality of raspberry pi
3. To check whether camera is working fine
4. Checking whether night vision camera is working fine and detecting faces at night
5. Checking alarm buzzer and water sprinkler conditions.




**Things that did not work:**

-The idea of water spray did not work.

**Trials and tests:**

1. Checking if the face detection algorithms are accurate and fast

2. Checking functionality of raspberry pi

3. To check whether camera is working fine

4. Checking whether night vision camera is working fine and detecting faces at night

5. Checking alarm buzzer and water sprinkler conditions.

## Results:

1. Speed of execution: 5 seconds
2. Usability: real-time
3. Accuracy: 85%
4. Safety: 90%

## Conclusions:

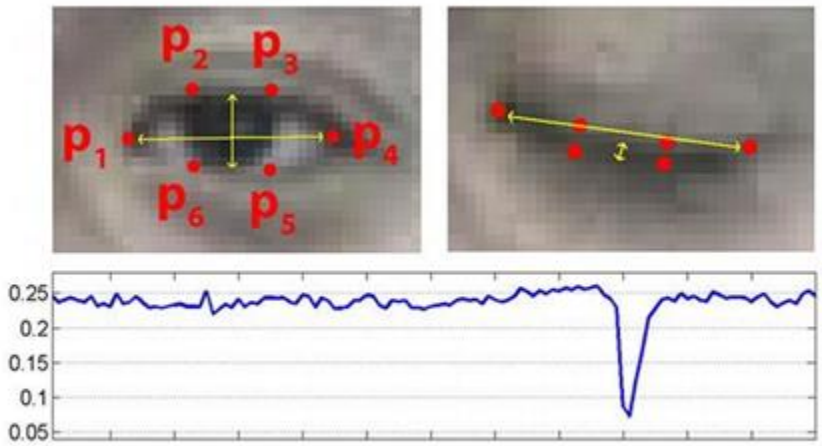
The expectation from our project was to reduce road accidents caused by fatigueness, , especially the ones at night when the drivers are very sleepy and have to go for long deliveries. We believed that it would help save people's lives. With this system , drivers could get a buzzer alarm as a warning to wake up incase they're about to fall asleep.

As a conclusion we could say that we were able to implement all of the aspects which include head tilts, yawns and eye closures. Also we were able to integrate a night vision camera. We had also planned on introducing a water spray to sprinkle water if the system detects eye closures, however after taking feedback from FYP-1 , we planned not to do that because it would have caused an accident if the driver spent time rubbing his eyes which had water, especially the ones who had to clean their glasses if they wore spectacles.

## Appendix:

### Appendix 1: Equations

EAR (eye aspect ratio) using the equation:
EAR =||p2-p6 ||+ ||p3-p5 || / 2||p1-p4 ||



**Appendix 2: Code**

Github repository : https://github.com/fcsiba/DFLD.git

**Appendix 3: Schematic of your hardware**

1.RaspberryPi
2. Camera

**Appendix 4: Software/parts list**

1. Python
2. OpenCv Library
3. Dlib library
4.Pycharm IDE

**Appendix 5:**

**Work-distribution**

**Syed Salman Haider** :

- Research and Development work on project
- Created Video presentation
- Wrote code for integration of machine learning algorithm
- Wrote code for Ear calculation
- Installed and configured necessary packages for the project including openCv library and dlib library
- Testing and debugging the code

**Hiba Haider**:

- Research and Development work on project
- Report and documentation
- Created Presentations
- Wrote code for detecting yawns
- Wrote code of sending necessary alerts and alarms for the driver
- Testing and debugging the code

**Fatima Zain Allarakha:**

- Research and Development work on project
- Testing and debugging the code
- Designed the workflow for the project
- Wrote code for detecting head tilts
- Wrote code of detecting facial landmarks
- Integrated effective sounds for head tilts and yawns in the code

**Appendix 6: Project timeline:**

# January 2019 (Week 1-4)

-Read up on research articles and started formulation of timeline.

-Gather user reviews of our fyp-1 by conducting surveys and cognitive walkthroughs. Requirement Gathering for the project.

-Research and Development for the project.

-Watched tutorials and did experiments.

-Reading up on OpenCV and it's libraries.

# February 2019 (Week 5-8)

- ERD and Wireframes
- Face Detection using machine learning algorithm
- Found out about Machine learning algorithms and software's used for analysis and accurate results.

# March 2019 (Week 9-12)

- Yawn Detection
- Head tilts detection
- Eye Closures detection

# April 2019 (Week 13-16)

- Configuration of night vision camera
- Added a sound effects for alert signals with adjustable volume
- Used Laptops camera to capture Facial features such as Eye closures, Blinking speed, head poses and yawning.

# May 2019 (Week 17-20)

- Debugging the final code.
- Resolving any glitches that arise.
- Making of posters, panaflex and video.

## References:

Inspirations for code and designs: https://www.pyimagesearch.com/
Papers: https://www.researchgate.net/

## Acknowledgements:

We would like to thank IBA for giving us this opportunity to show cast our skills and expertise in the symposium in front of a lot of guests.

We are also extremely privileged to have such motivating teachers who taught us throughout this 4 year journey and would really like to thank them all for teaching us and enhancing our skills, due to which we were able to execute this project.

A special thanks to our supervisor, Syed Sami Ul Ahbab, and our coordinator Sir Waseem Arain for their continuous guidance and support without which we would have been unable to execute this project.