International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-6, April 2020

Driver Attention Monitoring System using IBM Cloud

Sathya T, Reshi Krish T, Monish Kumar MS, KS Preetha

Abstract**: *With online shopping and many logistic companies on the rise, a single accident can incur heavy loss to the supply chain department and not only disrupts the flow of the supply chain, but also causes injury to life and damage to property. These accidents occur primarily due to driving while feeling distracted or drowsy and it is paramount to monitor such behavior to avoid drastic outcomes in case of driving heavy duty vehicles. Therefore, it is natural for logistic companies to invest in securing their goods and ensuring that there is safe transportation of goods.***

The objective of our paper is to provide a novel solution to handle the aforementioned problems by monitoring the driver's performance by analysing the facial features of the driver in real-time while storing the event-triggered data in the cloud and using the cloud services to send mobile alerts when the driver is drowsy or distracted via a mobile application in a cost effective and in an efficient manner.

Keywords**: *Facial feature recognition, DLib, SolvePnP, IBM Watson IoT Platform (WIoTP), NodeRED, IFTTT (If This Then That).***

I. INTRODUCTION

Driver when losing his focus while driving by feeling drowsy, yawning or taking his eyes off the road, is said to be distracted and it can prove to be chaotic to any passengers, pedestrians and the goods on the vehicle. As a result, one of the reasons behind vehicle accidents is due to the negligence. Amongst which one of them is caused by the drowsiness of the driver while driving a vehicle [1].

A study conducted in the year 2013 states that drowsy driving alone is responsible for causing 72000 crashes in total, 44000 injuries, and 800 deaths by the NHTSA [2]. Also, these numbers are not accurate since many injuries and crashes are not completely reported to the authorities.

Revised Manuscript Received on April 30, 2020.
***** Correspondence Author

Sathya T*, School of Electronics Engineering (SENSE), B. Tech, Electronice and Communication Engineering with specialization in Internet of Things and Sensors, Vellore Institute of Technology, Vellore, India. Email: tsathya98@gmail.com

Reshi Krish T, School of Electronics Engineering (SENSE), B. Tech, ECE with specialization in Internet of Things and Sensors, Vellore Institute of Technology, Vellore, India. Email: reshi18krish@gmail.com

Monish Kumar MS, School of Electronics Engineering (SENSE), B. Tech, ECE with specialization in Internet of Things and Sensors, Vellore Institute of Technology, Vellore, India. Email: monish.ms45@gmail.com

Dr. KS Preetha, Department of Communication Engineering, School of Electronics Engineering (SENSE), Senior Assistant Professor, Vellore Institute of Technology, Vellore, India. Email: kspreetha@vit.ac.in

In India alone, out of every 100,000 inhabitants, 22.6 fatalities occur on road every year.

Another study reported an estimated of 328,000 cases of vehicle accidents occur annually due to drowsy driving among which 109,000 of the total cases involving drowsy driving have resulted in injury of which 6400 were fatal by AAA foundation. The study concludes that the accidents involving drowsy driving is 350% greater than what is reported [3].

According to a report given by the Ministry of Transport, India, a table which gives a trend of how road lengths, vehicles change the total number of road accidents and the total number who face death solely because of road accidents from the year 2005-2016. The road accidents report has been released by the State Police Authorities from which a comparison has been made between the deaths and the road accidents.

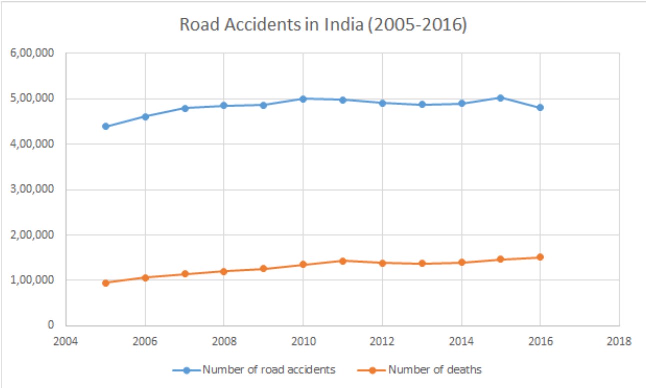

Fig. 1. Road Accidents in India

In addition to the statistics related to accidents occurring due to drowsy and distracted driving, the land transportation of cargo is a significant part of logistics expenditure. Looking into transportation economically, it is one of the defining elements of the production process. Transportation involves accounting for the price of the fuel, the management of the trucks including cargo handling, packaging, risk insurance, customs procedures, to monitor if the cargo is intact and has no damage whatsoever to meet the customer requirements and ensure the delivery time of the cargo is not delayed at any cost. A driver, due to carelessness might be the root cause for accidental impacts, i.e., goods knocking against each other or against the container wall. There is more chance for the goods to be experiencing shocks and vibration under drastic conditions. Depending upon the driving performance, this can prove to be quite severe since the distracted driving might lead to improper braking and acceleration which can damage the cargo which the truck is carrying.

This can prove to be a tremendous loss for the logistics department since drowsy driving as a reason plays poorly during freight claims for damaged goods.

For long distance land transportation, the major problem which drivers face is feeling drowsy, so to bring light to these problems stated, a system has been designed which focuses primarily on the driving patterns of the driver to monitor the distractions faced and record it simultaneously in real-time with help of a hardware prototype that is mounted on the vehicle to observe the facial movements by the driver with the help of a camera using OpenCV for facial detection and the data collected by the camera is processed on the edge level using a processor which is then sent to IBM WIoTP cloud to visualize the triggered events due to driver distractions.

II. LITERATURE REVIEW

To understand the concepts and the work that has already been carried out, it is important to do a literature survey on the existing work in this field. Several research has been done in this field or related to this field that supports as a base to this paper.

Sharma S, Shanmugasundaram K. and Ramasamy S.K., 2016, [4] proposed a CNN based system for face recognition using Dlib library. This paper proposes a system which involves deep learning technique i.e., CNN with Dlib face alignment for recognition of facial features. From the proposed technique they could find out the False Acceptance Rate as well as the accuracy. It is an effective method of facial recognition of a person using computer vision and programmed in C++. The model has been divided into 4 main parts: Face Detection, Face Cropping and Feature Extraction. Face detection is the process of recognition of the frontal face of a human from digital images or videos. Facial cropping is to cut only the relevant features from a facial image for the neural network to learn from that. It comes with an advantage of cropping the face with variable resolution of the image based on the distance of the face. Feature extraction is done using CNN to extract facial features like nose and upper lips using computer vision where the nose is considered as the center point of the facial landmarks. Dlib algorithm used for facial recognition which poses an effective way of minimizing the analytical functions.

Fabio Tango and Botta M [5] proposed a system which detects and monitors driver distractions in real-time using machine learning techniques in real-time. The paper emphasises on detecting cognitive and visual distractions by analysing the performance of the models namely, SVM, FFNN, LRNN and ANFIS. All of which have proven to be a means of detecting driver distraction. The data used for training the model have been collected from a driving simulator where hunans performed SURT which stands for surrogate visual research task which basically means performing tasks that can be considered as a distraction while driving. The authors have also pointed out the personalization aspect, with one specific model for each subject. The paper concluded that SVM classifiers outperformed others by giving results of high accuracy compared to the others, in result providing the highest classification rate for most subjects that had been used for this experiment.

Kyungwon Chang, Byung-Hun Oh, and Kwang-Seok Hong [6] proposed a novel method that uses a smartphone-based driving assistance system that uses image recognition This system is able to find the distance between the vehicle in the front and the driver so that it can warn in case the vehicle gets far too close. It also provides an option of controlling the engine checkup and switching between front and rear camera with the user's voice using sound recognition. The Adaboost algorithm has been used for training the classifier and Samsung Galaxy S2 is used as the front and rear camera where the image captured is then resized to 120x160 pixels. The region of interest(ROI) on the face was focused mainly on the driver's eyes and mouth. These two facial features were used for detection if the driver is experiencing drowsiness.

Nataliya Boyko, Oleg Basystiuk, Nataliya Shakhovska [7] compares two major computer vision libraries, i.e., OpenCV's Haar Cascade and DLib. Analysis has been made based on the comparison of the efficiency of both OpenCV and DLib libraries with regards to the execution time to the number of iterations of the applied algorithms. Both the models of different algorithms are made to process the same images in order to analyze the advantages and disadvantages of the features in each one of them. The system is divided into 4 main parts: finding faces, face positioning, to define unique facial features, identification of a person. OpenCV algorithm proves to be a far better application for this particular use-case since heavy processing was not involved for a long duration. Based on the analysis, it has been concluded that there exists no single method for searching, positioning and organizing the recognition process. Depending upon the use-case that is given, specific technologies shall be used. Also, it is advisable to go with Haar Cascade if ample amounts of photos are provided for the model to train and work accordingly. Since the training model is done on numerous images, the accuracy is very high and is also fast in terms of processing speed.

Igor Lashkov, Alexey Kashevnik, Nikolay Shilov, Vladimir Parfenov, Anton Shabaev [8] proposed a system which is able to detect dangerous driving of the driver based on OpenCV and Dlib libraries on smartphone. In this paper, the image of the driver which is captured by the video cameras can describe its facial features like head rotation, state of the eye to see if its open or closed, lips state and accordingly identify a level of fatigue state which is processed on the smartphone that can continuously monitor the drowsiness level experienced by the driver and other distractions. Using OpenCV and Dlib, due to the processing being done on the smartphone, the efficiency and performance in online mode is very high. On offline mode, the results are based on statistical analysis provided by the cloud service by making use of statistics which were accumulated in real time and the statistics which were previously collected, stored and produced by machine learning tools.

Miguel García-García, Alice Caplier, Michèle Rombaut [9] proposed a system which utilizes Haar Cascade classifier (OpenCV) for distinguishing between two states, i.e., sleep deprived and rested.

International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-6, April 2020

The processing is done in such a way that the background subtractor is used to crop out only the necessary part of the image, i.e., the person's face and then normalizing the cropped image. The dataset used for the same is DROZY which comes with 14 people who were made to undergo psychomotor vigilance test for 10 minutes and which was used to monitor the accelerated sleep deprivation. The entire system processing was done on a laptop and to monitor in real-time, a smartphone was used for the same. MobileNet, which is a non-linear model is used in case of smartphones and the output for camera frames is probability of the frame to belong to the "sleep deprived" class. In case the probability of this class is more than 0.5, the driver in the frame is classified as "sleep deprived".

R. Manoharan and S.Chandrakala [10] proposed an OpenCV technique based system to monitor the distraction and fatigue experienced by the driver using the camera of the smartphone. The classes that were used for fatigue and distractions were namely: rubbing of eyes, drowsy blinking and yawning. The smartphone used for this experiment is Xiaomi Redmi 1s and an application has been developed for the same. The authors of the study conducted the experiments and evaluated the CPU load and the battery consumption of the developed system. They concluded their system consumed 12% of battery of continuous use for one hour. The paper highlights that the proposed approach might not prove to be a viable option under low/no light conditions.

III. METHODOLOGY

This paper gives sole emphasis on the topic of driver distraction detection using computer vision, cloudant storage and visualisation via a mobile application. The on-board driver monitoring will be used to check the state of the driver to see if he is feeling drowsy or is experiencing other forms of distractions while driving. This is done through Computer Vision using Deep Learning libraries of OpenCV namely, Dlib and SolvePnP which is based on Convolutional Neural Network that is used in majority of the research nowadays due to high accuracy as well as its efficiency.

The OpenCV libraries that are used for facial recognition and processing of the captured image from Raspberry Pi (RPi) cameras are DLib and SolvePnP. The RPi camera which captures the image is also processed on the Raspberry Pi which acts as an edge device.

The processed data is sent to the IBM Watson IoT cloud which comes under WIoTP and is stored in a cloudant database with help of NodeRED which is a programming tool by IBM. The triggered event, which is sent to the cloud, is then sent as a notification via a messaging application named Telegram using NodeRED bot. The RPi does the following steps once it captures the frame:

1. Cropping ROI
2. Facial Landmark detection
3. State Identification

Once the states are identified and events are triggered based on the driver's performance, the events are sent to IBM WIoT cloud. The entire flow process of this paper has been given in Fig. 2.

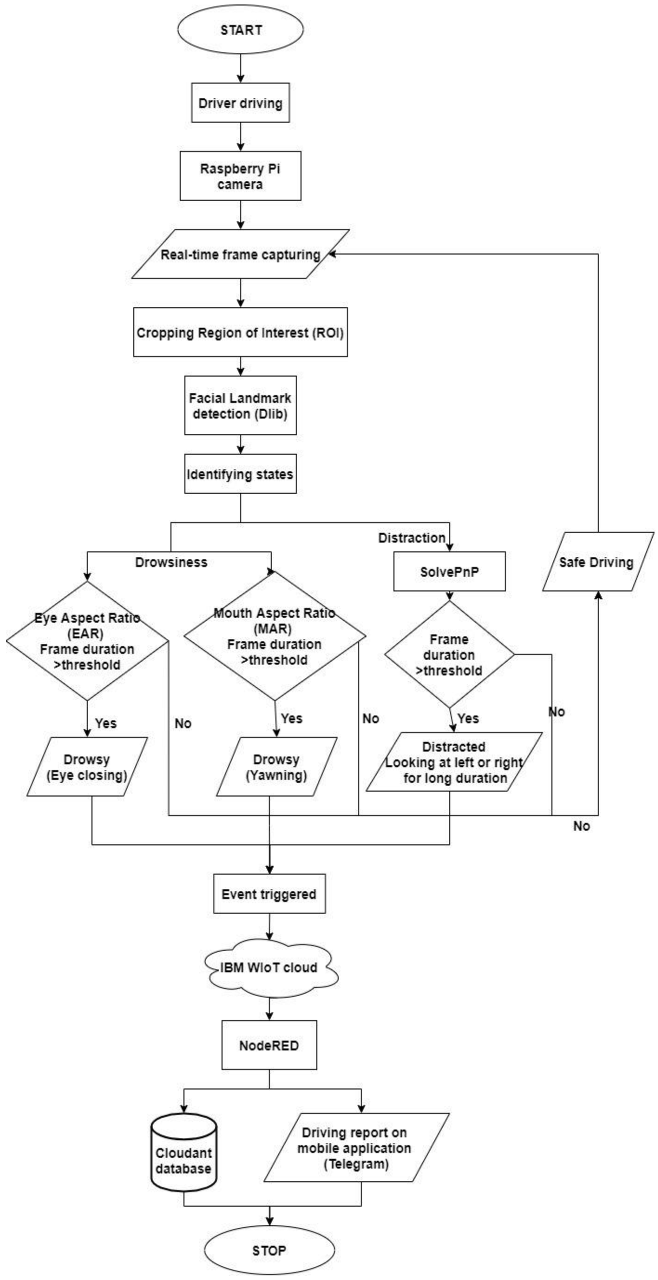

Fig. 2. Flowchart of the system

A. Cropping ROI

The ROI which we focus for this particular use-case of this paper is face since analysis is made on facial features. In the case of this paper, DLib HOG detector is used for cropping out ROI since it is more accurate and efficient for this particular use-case even though it is comparatively slower than other algorithms like Haar Cascade algorithm. The Dlib detector is used for cropping out the borders of the face which was initially trained from thousands of images.

B. Facial Landmarks Detection

This process is done in every facial recognition program. This step is done to identify the face and mark the facial features based on the requirement presented to the user. Hence, facial landmarking is done with help of shape_predictor_68_model, or in short SP68.

The SP68 model is created with the help of iBUG 300-W dataset which contains a vast number of images of faces that comes with pre-trained data consisting of 7764 images from which 68 different points on the face are extracted manually by researchers to specify each facial feature of the face as given in figure 4.3. The center of the face is considered to be the tip of the nose, which is the reference point for marking other facial features in the face. The figure given below is the feature points by the SP68 model.

Fig. 3. Facial feature points of pre-trained SP68 model

As the model is pre-trained, once the frame is captured using a Pi camera, the face alone is cropped out from the entire image for further processing using algorithms of the choice by the user depending upon the use-case.

The Dlib library itself comes with a pre-trained model on iBUG-300W which can identify based on 68 points (x, y pairs) on the face, and therefore be able to localize the features on the face. This library is based on ERT which stands for Ensemble Regression Trees which offers a simple and a fast technique to estimate the position of the facial feature landmarks. The landmark positions that are estimated are iteratively refined which is performed by regressors. The regressors give out new sets of points after every iteration which can bring down the error caused due to alignment of the estimated points.

Dlib algorithm has been the choice of usage for this paper because the facial landmarking options provided by this algorithm to localize and identify the features in the face are: eyes, nose and mouth. The detection and identification process can be divided into subparts:

1. Facial feature detection using DLib

One of the problems with facial recognition methods which needs to be accounted for is when the face tracking is not accurate. Typically, if the program is not able to recognize a face, the facial detection part is repeated all over again and then the facial feature detection and identification is done. But in the case of this paper, since we are using Dlib and SolvePnP algorithms for implementation of facial landmark detection and 3D tracking of the driver's face, the condition of face tracking getting lost is very minimal.

2. Facial features identification –

This step is performed to identify the features of the face and what you do with it accordingly. In the case of this paper, we take 3 classes of facial feature identification, they are Drowsiness which includes closing of eyes as well as yawning detection and distraction which includes looking in the left or right direction for a significantly long duration of time. In case of drowsiness, Dlib library alone is used. Whereas, in case of distraction, SolvePnP library also has been used to monitor the rotation of the head and not lose track of facial recognition.

In this paper, the DLib library function is used for returning the real time positions of different feature points in a face. With help of DLib and SolvePnP library, classification of the driver's behaviour have been done and the classes used are:

1. Safe Driving
2. Drowsiness
    a. Lack of sleep (Closing eyes)
    b. Yawning
3. Distraction
    a. Looking left for a long duration
    b. Looking right for a long duration

The drowsiness detection of the driver is done by the following steps:

1. Initially, setting a threshold for closing of the eye and opening of mouth
2. Setting counter for duration of blink and drowsy so as to distinguish between blinking and feeling drowsy i.e., having eyes closed for a considerably long duration. In case of yawning, counter for the duration of yawn so that we can distinguish if the driver is yawning or just opening his mouth for a short duration.
3. Monitoring the counter value- The duration for drowsy condition is set in such a way that the eye must be closed or mouth must be open for a longer duration and cross the threshold value set by the developer.

C. Eye Aspect Ratio (Drowsiness)

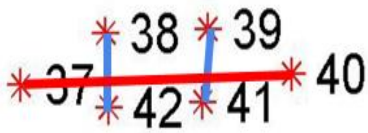

Fig. 4. Eye points marking

EAR or Eye Aspect Ratio is a part of facial landmarking that is used to track the movement of the eye to see if the eye is opened or closed. The points given on the figure are a part of the 68 facial feature points which were pre-trained in the SP68 model. Determination of EAR value of one eye is given below:

$$EAR = \frac{1}{2} * \frac{\text{Euclidean Distance between blue lines}}{\text{Euclidean Distance between red lines}} \quad (1)$$

$$EAR = \frac{1}{2} * \frac{ED(38,42) + ED(39,41)}{ED(37,40)} \quad (2)$$

ED mentioned on the formula stands for Euclidean Distance. The points of the eye are from 37 to 48 where 37-42 is for the right eye and 43-48 is for the left eye.

International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-6, April 2020

Now these points are put in an array from index 0 and Euclidean distance is calculated between vertical points and those are the blue lines. The Euclidean distance between horizontal points (37-40,43-46) are the red lines. The threshold for EAR is set by averaging the EAR of both the eyes.

D. Mouth Aspect Ratio (Yawning)

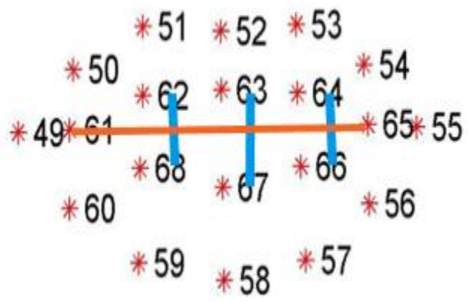

Fig. 5. Mouth points marking

MAR or Mouth Aspect Ratio is a part of facial landmarking which is used to track the movement of the mouth whether its opening and closing.

$$MAR = \frac{1}{3} * \frac{\text{Euclidean Distance between blue lines}}{\text{Euclidean Distance between red lines}} \quad (3)$$

$$MAR = \frac{1}{3} * \frac{ED(62,68) + ED(63,67) + ED(64,66)}{ED(61,65)} \quad (4)$$

The points of the mouth are from 49 to 65. The Euclidean distance between points 61 and 65 represents horizontal red lines and the pairs of points (62,68),(63,67),(64,66) represent vertical blue lines.

E. Facial rotation for long duration (SolvePnP)

If the driver has his head turned on either left side or right side for an uncomfortably long duration, then the system will consider it as a distraction and send the respective feed to the cloud and to the user via a mobile application.

The rotation of the head is monitored with help of an OpenCV algorithm named SolvePnP which complements Dlib for the sole purpose for facial tracking at different directions. In this paper, SolvePnP is especially used for solving the angle of rotation in 3D axis for facial landmarks.
The Solve PnP algorithm is used for giving projection of 3D points in a 2D image, i.e., by marking the points on edges of the face and finding the relative distance between each other to predict the rotation of the face from a 2D image. Hence, SolvePnP library is used for facial tracking so that the facial recognition is not lost even when the driver turns his head at some angle. Since it projects the image to 2D, it is possible to identify if the driver is rotating his head left or right side.
With help of PiCam, we are able to calculate the rotation of 6 Degrees of freedom with respect to the camera used. Hence, this library helps us in detecting if the person is turning left or right while driving. A counter is kept to monitor the duration of frames in which the person stays at left rotation or right rotation. Once the rotation has been detected after monitoring for a certain period of time, an event is triggered and is sent to the IBM cloud and the event is stored in the database.

The table given below shows the parameters which were used to execute this prototype.

TABLE 1: Parameters Used in hardware

PARAMETER	DESCRIPTION	VALUE
EAR_THRESH	Eye Aspect Ratio threshold	0.3
MAR_THRESH	Mouth Aspect Ratio threshold	0.5
MAR_CONSEC_FRAMES	MAR consecutive frames threshold	30
EAR_CONSEC_FRAMES	EAR consecutive frames threshold	45
yCOUNTER	MAR counter to count the consecutive frames iteratively	0
eCOUNTER	EAR counter to count the consecutive frames iteratively	0

F. Cloud storage and mobile application

The cloud plays a pivotal role as it is used to store all the events which are triggered by the hardware on the cloud and monitor in real-time from anywhere across the world. The cloud was interfaced with the hardware that we used in this paper is IBM Watson IoT Platform (IBM WIoTP) and cloud service named NodeRED.

1. IBM WIoTP

The WIoTP cloud is used in this paper since it has several benefits for this particular use-case and provides great efficiency in terms of runtime speed, latency and expenditure. The data which is accumulated once the event is triggered is monitored on the cloud and later is used to store it in a cloudant database and is sent to an application.
The events are maintained as a log and the respective states are also included in the log as well.
The cloud comes with a lite version which perfectly satisfies our use-case for this paper since it comes with great benefits at free of cost. If we want to scale up this prototype to a greater extent, then monthly plans can be purchased for a certain amount to the company to save our data in their servers with almost no risk of our data getting lost.

2. NodeRED, IFTTT and Telegram bot

NodeRED is one of the cloud services by IBM which provides a multitude of features that can be implemented in any system of desired use-cases. NodeRED is a cloud service by IBM that can be used to maintain a flow and perform several actions based on the use-case presented to the developer and it can also be used in real-time with the IBM cloud simultaneously. Integration of the NodeRED with the Telegram application has been done with help of the IFTTT.

The IFTTT platform's sole purpose is to do actions when the events are triggered. The NodeRED flow used in this paper has been given in fig. 6.

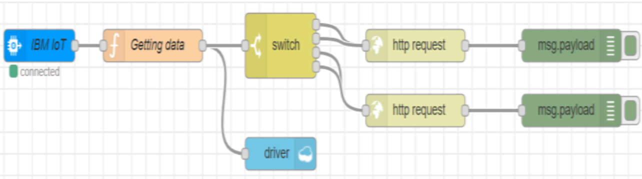

Fig. 6. NodeRED flow

The database stored can then be viewed by the user to keep track of the driving behaviour and note down whenever the driver is feeling drowsy or is distracted. It can then be used for giving feedback to the driver regarding his performance.

IV. RESULT AND DISCUSSION

The system has been successfully tested on our laptops and also deployed in a truck which is then monitored in real-time. This was done in two parts, that is, test phase and deployment phase. In the test phase, more emphasis was given for the literature survey on this particular use-case of this paper, designing the system, selection of algorithms which can be used for facial detection and tracking, deciding on the software used that provides faster and efficient results, code finalisation, integration and monitoring of data sent to the IBM WIoT cloud. The results were observed on laptops during the test phase.

In the deployment phase, the focus point was to deploy the hardware prototype on a truck and monitor the behaviour of the driver in real-time. Also more emphasis was given to the integration of software with the hardware components, cloud monitoring, data accumulation in the database, sending the triggered event message to Telegram application via IFTTT platform.

A. Test phase

The test phase is done on a laptop to test if our prototype is functioning without errors before being deployed on a truck for monitoring purposes. This phase can prove to be beneficial since most of the debugging and improvements are done in this phase alone.

Given below (Fig 7-Fig. 9) is the model which was used to test multiple classes on the laptop.

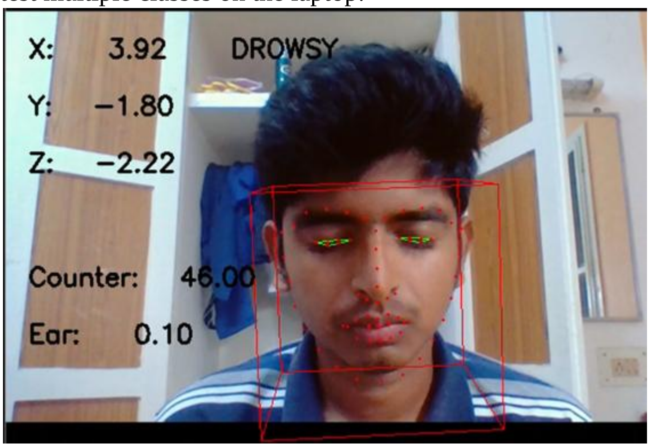

Fig. 7. State: Drowsy (Eyes closed)

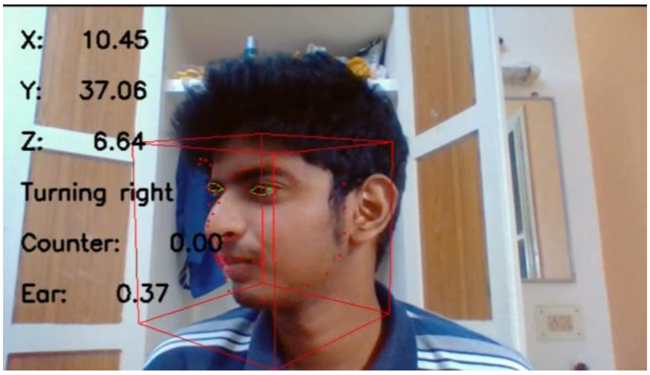

Fig. 8. State: Looking right for a long duration

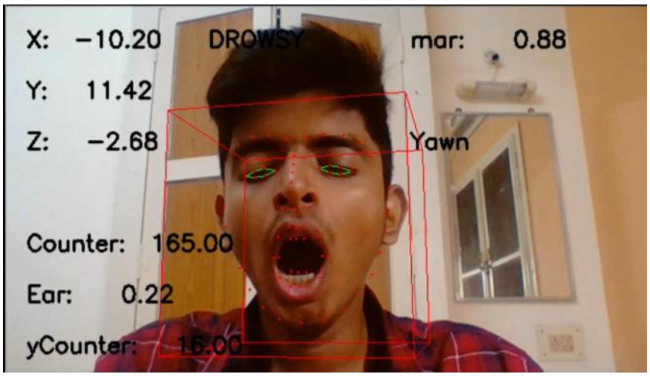

Fig. 9. State: Drowsy (Yawning)

Fig. 10 shows the event log that is maintained on the cloud end after every trigger of event. The event which was shown on the given below figure is the yawning state which is a subset of drowsiness state.

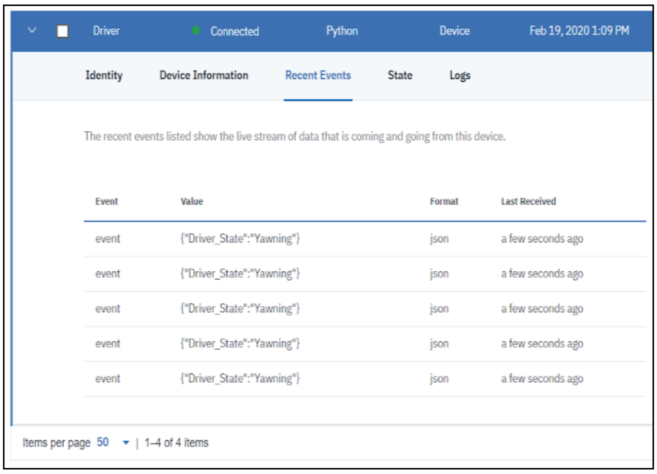

Fig. 10. State: Event log on IBM cloud end

B. Deployment Phase

This phase is done once the test phase is completed and is successfully executed. The deployment phase is when the prototype is used on a truck in real-time to monitor the driver behaviour and is then checked if the prototype works without any errors. Major constraints on the deployment phase is:

1. Positioning of the RPi camera to monitor the driver:

The camera positioning plays a pivotal role since the objective t is to monitor the driver's face to detect and track the movement precisely. If the camera is not oriented properly, the results may go haywire as the algorithm might not be able to detect or track the movement of the head and facial features.

2. Internet connectivity:

The internet connection also plays an important role in this system as the data needs to be sent to the cloud with minimum latency to provide good results. Minimal upload speed of the internet is enough to send the data to the cloud since each event triggered is in terms of few bytes. Without internet connectivity, real-time monitoring by the administrator of the logistics company is impossible since there is no way for the triggered events to be communicated with the company without cloud and database storage services. Fig. 11-14 shows us the model which was deployed in a truck with the event logs maintained simultaneously.

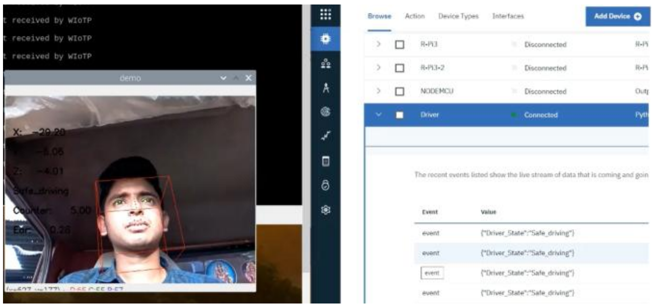

Fig. 11. State: Safe Driving

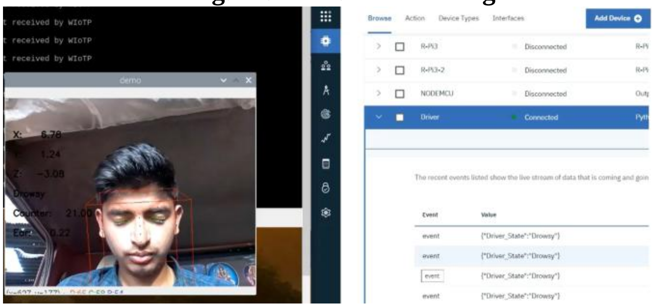

Fig. 12. State: Drowsy (Closed eyes)

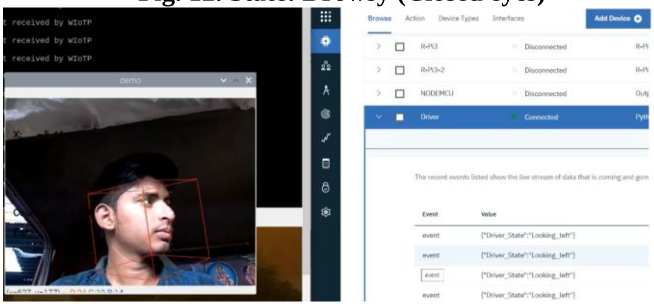

Fig. 13. State: Looking left for a long duration

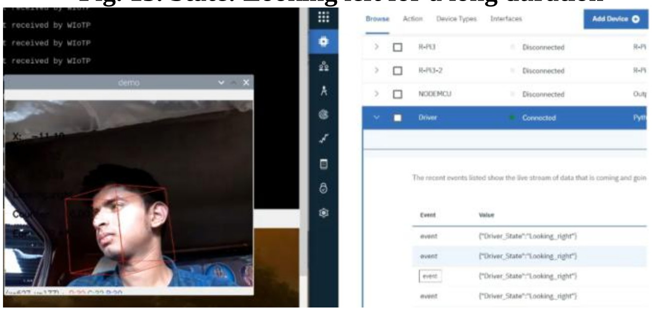

Fig. 14. State: Looking right for a long duration

These classes of drowsiness and distractions when triggered are considered as events. The event triggered data that is collected with the help of Raspberry Pi camera which captures the driver's performance is then processed on the edge device i.e., Raspberry Pi. By making use of cloud storage and analysis, we ensure that the processed data which is accumulated is then sent to the IBM WIoT cloud where the triggered events are maintained as logs.

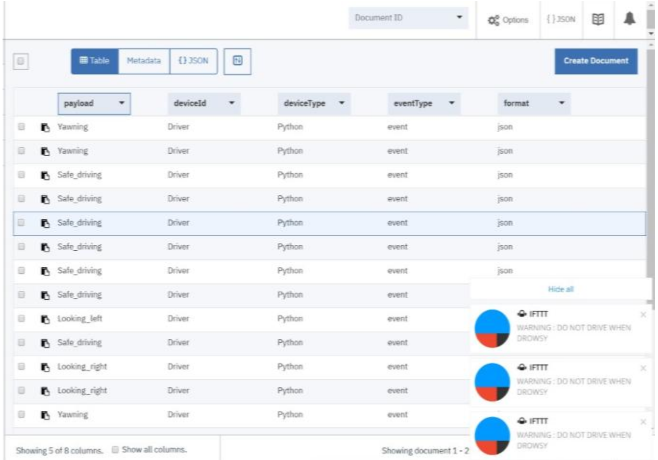

Fig. 15. Cloudant Database

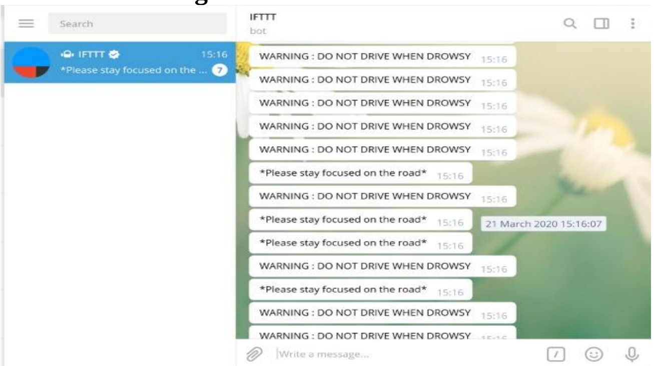

Fig. 16. Telegram IFTTT bot

V. CONCLUSION

In this paper, a system for monitoring driver distraction as well as the drowsiness is proposed using Raspberry Pi and camera for hardware, IBM WIoT cloud, NodeRED, IFTTT and Telegram application on software end. The target audience is logistic companies where the use-case is to monitor the driver's behavioural patterns that can be used to identify drowsiness or distraction. The drowsiness is detected based on classes like: closing of eyes, yawning. The distraction level is detected by monitoring the driver's head rotation and keeping a track of the duration of the posture in that direction. If the driver looks in the right or left direction for a long duration, it is considered a distraction. The accumulation of the events and sending the event triggered data on the IBM WIoT cloud and the storage of data is simultaneously stored on the database with help of NodeRED. This processed data is visualized via a mobile application named Telegram with help of NodeRED and IFTTT bot.

FUTURE SCOPE

The data which is collected from the driver can then be analysed further by the company to generate a feedback mechanism in form of a performance report as to their driving and the performance bonus on the salary could be given to the driver. In order to monitor a large number of trucks, the location of each truck can be monitored with help of GPS tracker and can be integrated to the IBM WIoT cloud with help of NodeRED which comes with an inbuilt feature of GPS plugin that can be used for the use-case.

Better and more sophisticated methods of monitoring can be implemented provided the processor has immense computational capacity.

REFERENCES

1. Johns, M.W., 2000. A sleep physiologist's view of the drowsy driver. Transportation research part F: traffic psychology and behaviour, 3(4), pp.241-249.
2. Barr, L., Howarth, H., Popkin, S. and Carroll, R.J., 2005. A review and evaluation of emerging driver fatigue detection measures and technologies. National Transportation Systems Center, Cambridge. US Department of Transportation, Washington. Disponível em< http://www. ecse. rpi. edu/~ qji/Fatigue/fatigue_report_dot. pdf.
3. Tefft, B.C., 2012. Prevalence of motor vehicle crashes involving drowsy drivers, United States, 1999–2008. Accident Analysis & Prevention, 45, pp.180-186.
4. Sharma, S., Shanmugasundaram, K. and Ramasamy, S.K., 2016, May. FAREC—CNN based efficient face recognition technique using Dlib. In 2016 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT) (pp. 192-195). IEEE.
5. Tango, F. and Botta, M., 2013. Real-time detection system of driver distraction using machine learning. IEEE Transactions on Intelligent Transportation Systems, 14(2), pp.894-905.
6. Chang, K., Oh, B.H. and Hong, K.S., 2014, January. An implementation of smartphone-based driver assistance system using front and rear camera. In 2014 IEEE International Conference on Consumer Electronics (ICCE) (pp. 280-281). IEEE.
7. Boyko, N., Basystiuk, O. and Shakhovska, N., 2018, August. Performance evaluation and comparison of software for face recognition, based on dlib and opencv library. In 2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP) (pp. 478-482). IEEE.
8. Lashkov, I., Kashevnik, A., Shilov, N., Parfenov, V. and Shabaev, A., 2019, August. Driver Dangerous State Detection Based on OpenCV & Dlib Libraries Using Mobile Video Processing. In 2019 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC) (pp. 74-79). IEEE.
9. García-García, M., Caplier, A. and Rombaut, M., 2018, June. Sleep deprivation detection for real-time driver monitoring using deep learning. In International Conference Image Analysis and Recognition (pp. 435-442). Springer, Cham.
10. Manoharan, R. and Chandrakala, S., 2015, February. Android OpenCV based effective driver fatigue and distraction monitoring system. In 2015 International Conference on Computing and Communications Technologies (ICCCT) (pp. 262-266). IEEE.

AUTHORS PROFILE

Sathya T, is currently pursuing his final year in B. Tech in Electronics and Communication Engineering with specialization in Internet of Things and Sensors from Vellore Institute of Technology, Vellore, India. An aspiring engineer who is interested in IoT, data analytics, cloud computing and edge computing. Email: tsathya98@gmail.com

Reshi Krish T, is currently pursuing his final year B. Tech in Electronics and Communication Engineering with specialization in Internet of Things and Sensors from Vellore Institute of Technology, Vellore, India. His area of interest is Data Mining, Sensors and Actuation, Robotics and ML. Email: reshi18krish@gmail.com

Monish Kumar MS, is currently pursuing his final year B. Tech in Electronics and Communication Engineering with specialization in Internet of Things and Sensors from Vellore Institute of Technology, Vellore, India. His area of interest is IoT, Cloud Computing and M2M communication ML. Email: monish.ms45@gmail.com

Dr. KS Preetha, received a B.E in Electronics and Communication Engineering in 2004 and obtained Master's in Communication and Networking from Madras Institute of Technology, Anna University, Chennai India in 2008. She completed her doctoral degree in Cognitive Radio Networks, 2019 from Vellore Institute of Technology, Vellore. She is currently serving as a Faculty in School of Electronics Engineering, Vellore Institute of Technology, Vellore, India. Her research interest includes Cognitive Radio Networks, 5G networks and IoT. Email: kspreetha@vit.ac.in