International Conference - Innovation-2021-Innovation-2021
International Journal of Scientific Research in Computer Science, Engineering and Information Technology ISSN : 2456-3307 (www.ijsrcseit.com)
Volume 8, Issue 3, May-June-2021

Smart Passenger

Akash Kurund*, Manthan Khariwal, Prashant Gaikwad, Akshay Mali, Varsha Babar
Department of Computer Engineering, Dr. D. Y. Patil School of Engineering and Technology, Pune, Maharashtra, India

ABSTRACT

Smart Passenger is a smart device which is used to keep driver away from his drowsiness and makes his journey peaceful. Smart Passenger keeps interacting with the driver so that he stays awake during his entire journey. Smart passenger has features such as mist spraying of water, alarm system, alcohol detection, etc. Smart passenger interaction with the driver will be completely based upon driver's interest as well as his hobbies. A camera fixed with this device keeps track over the driver's movement of eyes and face using ML algorithms. Once the drowsiness is detected, stage of drowsiness will be checked & appropriate action is performed.
Keywords : Machine learning, IoT, NLP.

I. INTRODUCTION

Based on a study by AAA Foundation on Traffic Safety about 328,000 drowsy driving crashes occur annually. The same study found that 109,000 of those drowsy driving crashes resulted in an injury and about 6,400 were fatal. To reduce these accidents, we have come up with a smart device which acts as a companion with a driver during his entire journey and keeps him awake. This device has a built-in camera which keeps track on the eyes and mouth of the driver. It also has a microphone which is used by driver to interact with device. Based on these inputs, this device alerts the user in various ways such as mist spraying of water or raising an alarm. Its main functionality is to interact with the driver and asking him different set of questions based upon his/her interest. Driver has to fill few information in his account on the Smart Passenger website where he can upload his area of interest, hobbies, etc. all the things which he loves to talk about. Then this device uses this given data and ask question based on it. Suppose if the driver gives interest as "Cricket" then the device will ask questions like "Hey buddy, who is your favourite cricketer?" or "What do you like most about cricket?". Also, device will ask questions based on his personal information such as "Which is your favourite holiday destination?" and if the user answers wrong or answers late, then device performs appropriate actions. Apart from this, it also has a GPS inside which helps him send his location to the specified contact. Not only device asks question but also the driver can ask questions such as "Hey smarty, what's the weather condition in Pune?". Smart passenger gives real time information to the driver fetching data from the internet. Once the driver is found drowsy, first step is to keep interacting with him. If still he feels drowsy, then the next step is to mist spray of water over his face to make him feel fresh enough. This water will be in a very little amount and won't stumble. Still if he feels drowsy, then an alarm is raised with a high volume so that he wakes up as well as nearby passing vehicles also be alert from the driver. It contains an alcohol sensor which detects if the driver is drunk. If found drunk, it asks driver not to drive car and also sends a message with location to the given contact number.

II. LITERATURE REVIEW

[1]. Deep Learning for Natural Language Processing & Language Modelling by Mr. Piotr Klosowski. This paper presents deep learning methods for language processing and modelling. Development of statistical language models helps to predict a sequence of recognized words and phonemes, and can be used for improving speech processing and speech recognition

[2]. A Survey on State-of-the-Art Drowsiness Detection Techniques by Muhammad Ramzan, Hikmat Ullah Khan, Shahid Mahmood Awan, Amina Ismail, Mahwish Ilyas and Ahsan Mahmood. Drowsiness or fatigue is a major cause of road accidents and has significant implications for road safety. Several deadly accidents can be prevented if the drowsy drivers are warned in time.

[3]. Telematics: Artificial passenger & beyond by Mr. Dimitri Kanevsky. This paper describes human-machine interface for in-vehicle technology that are based on conversational interactivity.

[4]. Drunken driving detection and prevention models using Internet of Things by Suparna Sahabiswas and Sourav Saha. In this paper a model based on IoT is proposed with the aim to safeguard drunk and drowsy drivers especially at night. It also discusses several models which have already been proposed and attempts to assimilate the best ideas which are proposed there.

[5]. Artificial intelligence techniques for driving safety by Zahid Halim, Rizwana Kalsoom, Shariq Bashir and Ghulam Abbas. This paper presents a study on the existing approaches for the detection of unsafe driving patterns of a vehicle used to predict accidents.

[6]. Facial expression recognition using face-regions by Khadija Lekdioui, Yassine Ruichek, Rochdi Messoussi, Youness Chaabi and Raja Touahni. This paper proposes a facial expression recognition method based on a novel facial decomposition.

[7]. Research on Speech Recognition Technology and Its Application by Youhao Yu. The paper depicts the speech recognition system and the main techniques of speech recognition, and makes a preliminary exploration for its application in various fields.

[8]. Driver fatigue detection system by Yogesh Chellappa, Narendra Nath Joshi, and Vaishnavi Bharadwaj. The research aims to detect the onset of drowsiness in drivers, while the vehicle is in motion. Detection is done by continuously looking out for symptoms of drowsiness.

[9]. Real-Time Eye Blink Detection using Facial Landmarks by Tereza Soukupova and Jan Cech. This paper presents eye blink system which uses Facial points and a threshold value for differentiating between drowsy eye blink and normal eye blink.

III. FUNCTIONS OF SMART PASSENGER

A. Alcohol Detection

Alcohol detection is one of the features in this device which uses sensor to detect alcohol and raise and alarm for warning other passing cars.

MQ3 sensor is used in detecting alcohol.

MQ-3 gas sensor has high sensitivity to Alcohol, and has good resistance to disturb of gasoline, smoke and vapor. This sensor provides an analogue resistive output based on alcohol concentration. When the alcohol gas exists, the sensor's conductivity gets higher along with the gas concentration rising. Whenever the user enters the car, the stinky smell of alcohol is detected by this sensor and informs the microprocessor to raise an alarm.

Fig.2.1 MQ3 Sensor

B. Water Spray

Water spray is key functionality of this device. This device sprinkles a very low amount of sweet smell water over user face to make him feel fresh. This spray won't stumble driver as the amount of water is very less. This uses a small refillable bottle with a spray notch head over it. It is actuated using a 100 RPM centre shaft DC motor. With the help of this motor, the spray nozzle is actuated.

Fig. 2.2 DC Motor

C. Real time Location

If the driver is drunken, then his real time location is sent to the specified mobile number which he has added to his profile as an emergency contact. This will help his relatives to track him if he meets with an accident due to drunken drive. Also, in future scope this location can be forwarded to nearby police stations so that cops can trace him. It uses a REB 4216 GPS module to locate the driver. Based on the latitude and longitude, the driver's location can be found. This will get internet connectivity through a GSM module which consist of only internet scheme.

Fig.2.3 GPS Module

D. Drowsiness Detection

This device's main functionality is drowsiness detection. Using computer vision algorithms, we can automatically detect driver's drowsiness in real time video stream and then take appropriate action. It uses Raspberry Pi Camera module to detect drowsiness. To detect drowsiness, we first have to detect eye blinks in video stream. If the eye is closed for more than a specified threshold, then drowsiness is detected. This is done by finding Eye Aspect Ratio (EAR). It is represented by Σ(x, y) co-ordinates, starting from the left corner of an eye and then working clockwise around remainder of the region.

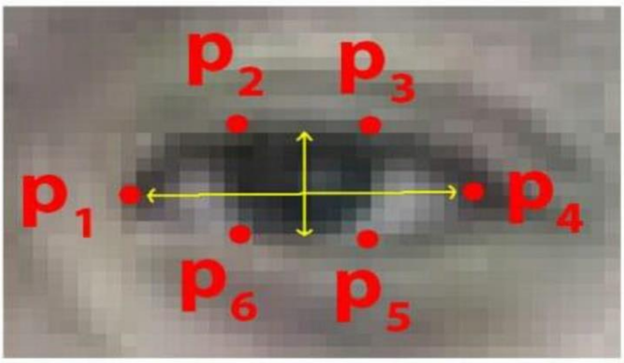

Fig.2.4 EAR

So, after getting these 6 co-ordinates of an eye, we can then derive an equation that reflects thin relation called the Eye Aspect Ratio.

$$\mathbf{EAR} = \frac{\|P2 - P6\| + \|P3 - P5\|}{2\|P1 - P4\|}$$

Where p1, p2…., p6 are facial landmark locations. The numerators of this equation compute the eye verticals landmarks, while the denominator computes the distance between horizontal eye landmarks. Since there is only one set of horizontal points but two sets of vertical points.

So, how EAR equation detects the blink of eye? Well, the EAR is approximately constant while the eye is open but will rapidly fall to zero when an eye blink occurs.

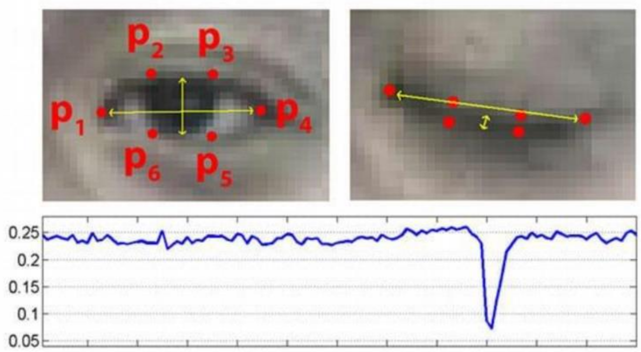

Fig.2.5 EAR Calculation

So, consider Fig.2.5, on the left side we have an eye which is fully open – the EAR is large and constant over time. However, once the person blinks the eye, EAR decreases to zero, then rapidly increases again, indicating a single blink has taken.

Using same technique, we compute eye aspect ratio for each eye, which gives us singular value. We than have declare the constant "Threshold" i.e. 0.25 & "Frame check" i.e. 20 in our algorithm.

So, if eye aspect ratio falls below this threshold, we will start counting number of frames the person has closed their eyes for. If the person's eye is closed for more than 20 frames, then it will be detected as drowsy and appropriate action will be taken.

E. Human Machine Interaction

By using NLP algorithms, human machine interaction is made possible. Data generated from conversation are unstructured data. Unstructured Data usually do not fit properly into the traditional row and column structure of databases, and represents the vast majority of data available. Similarly, whatever user speaks through microphone is unstructured and we need to handle this which is very messy and complex part. During this process we use tokenization technique of NLP where segments of the voice are broken into small piece of text called token where each token will be a word spoken by the user from microphone. Suppose if user says "Hey Smarty, what's the weather condition in Pune?" While driver is speaking this tokenization, technique breaks it into tokens like [Hey][Smarty][what's][the][weather][condition][in][Pune]. Once it is broken into tokens it becomes easy to understand what user said by accessing these tokens. Tokenization removes the commas, question mark, etc. so that processing time is reduced.

IV. WORKING OF SMART PASSENGER

Working of Smart Passenger is very simple, driver first interacts with smart passenger using an inbuilt microphone. At the same time, this device keeps monitoring driver's face to detect drowsiness. This thing continues in a loop for long time. If any alcohol gas is detected inside the car, MQ3 Sensor will detect the gas and makes a call to alert system.

For drowsiness detection it, performs several operations such as extract frame, find facial landmarks, calculate EAR, etc. Based on that decision is made whether driver is drowsy or not. If drowsy, then a call to alert system is made.

For Natural Language Processing, the input taken from microphone is given to NLP algorithm. Then the algorithm processes the sound with different operations such as morphological processing, syntax analysis, semantic analysis, etc. Based on that smart passenger interacts with the driver. Also, if the user fails to answer the question to in given time then a call to alert system is made.

Questions asked to the driver are based on the profile which driver filled while creating an account on smart passenger official website. This data is accessed over the internet into the smart passenger and based on this the entire conversation is planned.

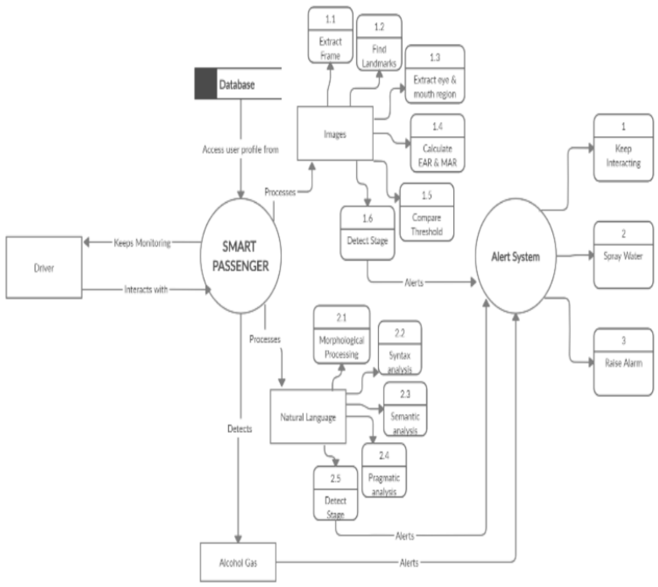

Fig. 3.1. Working of Smart Passenger

V. RESEARCH METHODOLOGY

To start, we will apply OpenCV's Haar cascades to detect the face in an image, which boils down to finding the bounding box *(x, y)*-coordinates of the face in the frame.

Given the bounding box the face we can apply dlib's facial landmark predictor to obtain **68 salient points** used to localize the eyes, eyebrows, nose, mouth, and jawline:

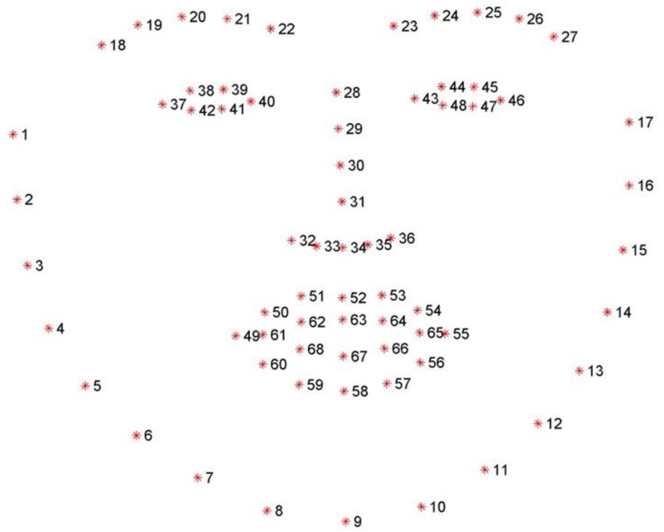

Fig 4.1: Visualizing the 68 facial landmark coordinates

Given the facial landmarks associated with an eye, we can apply the *Eye Aspect Ratio (EAR)* algorithm.

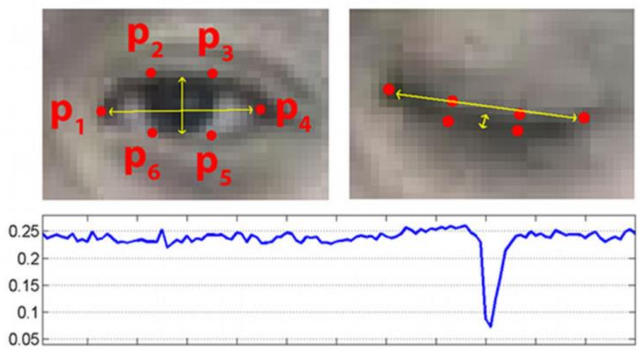

Fig 4.2: *Top-left:* A visualization of eye landmarks when then the eye is open. *Top-right:* Eye landmarks when the eye is closed. *Bottom:* Plotting the eye aspect ratio over time. The dip in the eye aspect ratio indicates a blink

On the *top-left* we have an eye that is fully open and the eye facial landmarks plotted. Then on the *top-right* we have an eye that is closed. The *bottom* then plots the eye aspect ratio over time. As we can see, the eye aspect ratio is constant (indicating that the eye is open), then rapidly drops to close to zero, then increases again, indicating a blink has taken place.

In our drowsiness detector case, we'll be monitoring the eye aspect ratio to see if the value *falls* but *does* *not increase again*, thus implying that the driver/user has closed their eyes.

Once implemented, our algorithm will start by localizing the facial landmarks on extracting the eye regions and then finally raising an alarm if the eye aspect ratio is below a pre-defined threshold for a sufficiently long amount of time (indicating that the driver is tired).

VI. SPEECH RECOGNITION

- speech recognition –" Library for performing speech recognition, with support for several engines and APIs, online and offline"
- pydub – "Manipulate audio with a simple and easy high-level interface"
- gTTS – "Python library and CLI tool to interface with Google Translates text-to-speech API"

The next thing to do and likely most importantly for a speech recognition feature is to recognize speech. To do so, we'll need to first capture incoming audio from the microphone, and then perform the speech recognition. This is all handled via the speech recognition library.

VII. CONCLUSION

In this paper, we have studied about facial expression detection as well as voice-based interaction by the machine with the driver using Machine Learning. From this paper, we conclude that Drowsiness detection can be handled with a very good way using Machine Learning.

VIII. REFERENCES

[1]. Deep Learning for Natural Language Processing & Language Modelling by Mr. Piotr Klosowski SAP 2018, September 19th -21st, 2018, Poznan, POLAND.

[2]. A Survey on State-of-the-Art Drowsiness Detection Techniques by Muhammad Ramzan, Hikmat Ullah Khan, Shahid Mahmood Awan, Amina Ismail, Mahwish Ilyas and Ahsan Mahmood in IEEE Access on 1st May 2019.

[3]. Telematics: Artificial passenger & beyond by Mr. Dimitri Kanevsky at IBM, T. J. Watson Research Center.

[4]. Drunken driving detection and prevention models using Internet of Things by Suparna Sahabiswas and Sourav Saha.

[5]. Artificial intelligence techniques for driving safety by Zahid Halim, Rizwana Kalsoom, Shariq Bashir and Ghulam Abbas at Artif Intell Rev DOI 10.1007/s10462-016-9467-9

[6]. Facial expression recognition using face-regions by Khadija Lekdioui, Yassine Ruichek, Rochdi Messoussi, Youness Chaabi and Raja Touahni in 3rd International Conference on Advanced Technologies for Signal and Image Processing - ATSIP'2017, May 22-24, 2017, Fez, Morroco.

[7]. Research on Speech Recognition Technology and Its Application by Youhao Yu in 2012 International Conference on Computer Science and Electronics Engineering.

[8]. Driver fatigue detection system by Yogesh Chellappa, Narendra Nath Joshi, and Vaishnavi Bharadwaj in 2016 IEEE International Conference on Signal and Image Processing.

[9]. Real-Time Eye Blink Detection using Facial Landmarks by Tereza Soukupova and Jan Cech in 21st Computer Vision Winter Workshop Luka Cehovin, Rok Mandeljc, Vitomir Struc (eds.) Rimske Toplice, Slovenia, February 3–5, 2016.