On-Board Diagnostics and Driver Profiling

Navneeth S[1], Prithvil K P[1], Sri Hari N R[1], Thushar R [1], Dr. M Rajeswari[2]

[1]Students, Dept. of Telecommunication Engineering, Bangalore Instituteof Technology, Bangalore, Karnataka, India
[2]Associate Professor, Dept. of Telecommunication Engineering, Bangalore Institute of Technology, Bangalore, Karnataka, India

---***---

Abstract - *Automobile manufacturing industry is an ever-developing industry driven to satisfy consumer's needs focusing on key factors like performance, comfort, luxury, eco-friendliness and most importantly driver and passenger safety. Driver profiling in modern times is a decisive and convenient method ensuing driver and passenger safety. This project offers a knowledge base outline to the consumer for On-Board Diagnostics (OBD) and Driver Profiling. On-Board Diagnostics assists the user in car maintenance by enlisting Diagnostic Trouble Codes (DTC) from the car's Engine Control Unit (ECU) using an Android application. The second aspect of this project, i.e. to profile the driver's behaviour is proposed in two different approaches, offering mathematical, visual and analytical analysis of consumer's driving behaviour. A simple, immediate and cost-effective approach to profile driver behaviours using GPS co-ordinates is accessible on the same Android application proposed for On-Board diagnostics. An alternate detailed, novel approach to profile driver behaviour is based on car engine parameters such as Engine speed/rpm, Vehicular Speed, Engine load & Throttle valve; is modeled using Data Analytics & unsupervised Machine Learning techniques.*

***Key Words*: On-Board Diagnostics (OBD), Diagnostic Trouble Codes (DTC), Driver behaviour Profiling, Risk Score, K-means clustering, Elbow method, Machine Learning, Data Analytics.**

1. INTRODUCTION

Car diagnostics is carried out by various leading car manufacturers like Toyota, Hyundai, Maruti Suzuki and Renault for their customers at their respective service centers. A survey conducted by our team to these manufacturers found that the industries use their own proprietary diagnosis tools. The diagnostics tool is used to obtain DTC or "fault codes" and information like rpm, fuel consumption, and air conditioning readings from the ECU at real-time. The fault codes help service technicians to understand the condition of the various parts of the Engine using their On-Board diagnostics tool. Our project provides a tool to the user to individually track and monitor the condition of the vehicle's ECU using an Android app.

Driver profiling is an essential modern commodity useful in categorizing an individual based on their driving behaviour. Driver safety is one of the most critical but under looked aspect in India. Indian roads are considered to be one of the most dangerous roads around the world and a lot of lives are lost due to negligent driving. Out of all the vehicle accidents, 78% of these accidents occur due to driver's negligence like distraction, rash driving and over speeding. Driver profiling measures an individual's attitude towards driving, it provides excellent insights into the risk posed by drivers. Driving behaviour data is based on on-board sensors installed by car manufacturers to monitor speeds, accelerators, harsh deceleration, abrupt lane change, travel time etc. Our project provides a tool to the user to interpret the risk in the user's driving behaviour resulting in an accident.

1.1 Problem Statement

- Users at service centers generally are unaware and not certain of the parts being replaced and/or serviced and end up paying up hefty bills for it. Mechanics may exploit naïve users and charge them for parts that might/ might not have been replaced.
- There were nine road accidents that killed three people for every 10 minutes in 2015, with an expected increase of 9% over four years. These stats are evolving and drastically worsening causing a steep increase in deaths.
- Cases of over-speeding are constantly causing accidents due to the driver's irresponsibility.
- Unnecessarily revving the engine not only harms the vehicle but also has a bad impact on fuel economy.

1.2 Project Objectives

The objectives of this project are as follows:
- User based car maintenance.
- Easy maintenance.
- Avoiding traffic violations.
- To know the problems existing in the engine before the user gives it to the service center.
- To ensure the driver's safety.
- Profile user's driving behaviour.

1.3 Application

- *Maintenance*: This application helps an individual to maintain their automobile on a regular basis and to be independent of a mechanic. The vehicle owner can understand minor glitches in the vehicle and can thus avoid frequent visits to a mechanic for petty issues saving both money and time. With basic knowledge to troubleshoot minor issues, the user can themselves patch it up preventing unnecessary expenditure.
- *Driver profiling*: This application helps in measuring the number of acceleration, braking and turning events conducted by a driver, as well as the severity of the events and the distance travelled, Results are achieved through the collection of data from the vehicle, which allows us to build a detailed picture of which drivers need to improve their performance.

2. LITERATURE SURVEY

Various research papers were accredited to provide a solution to the problem statement and are tabulated below.

Table-1: Literature survey

Sl. no	Reference Paper	Information Gathered
1	Android-Based Universal Vehicle and Tracing System [1].	-Vehicle Performance -On-Board Diagnostics using OBD-II
2	Vehicle Health Monitoring System [2].	-On-Board diagnostics using OBD-II -Vehicle Health Report
3	The Driver Behaviour profiles for road safety analysis [3].	-GPS based Driver behaviour profiling -Risk Score
4	Online Black Box System for Cars [4].	-System on Chip (SoC) for embedded Car Black Box -SD card storage -GPS and GSM
5	Driving Behviour Analysis Based on Vehicle OBD Information and AdaBoost Algorithms [5].	-Driver Profiling using AdaBoost ML technique -OBD parameters to profile driver

3. METHODOLOGY

The design flow of the project is represented in Fig 1 and each stage is briefed below:

Fig- 1: Design Flow of the Project

- **Data sensing:** The data from the car's engine such as vehicle speed, RPM, acceleration, etc. are sensed by the sensors generate DTC, used for car diagnosis and is stored in the ECU.
- **Data Acquisition:** The data from the ECU is acquired and transferred over Bluetooth to the Android device using an OBD-II adapter.
- **Data Processing:** The data collected from the OBD-II through the ECU's sensors are processed on the smartphone using the application.
- **Data Storage:** The smartphone also makes provision to store the processed data, on-board using an SD card which serves as a Black Box to study the vehicle's condition and driver's behaviour patterns in case of an accident.
- **User Interface:** The user is able to view the results of OBD using an Android application ensuring car maintenance. The Android application is also able to profile the driver's behaviour by providing a Driver Score. A detailed Driver Score is available using the ML & DA model.

The block-diagram of the project is visualized as in Fig 2. The ECU is the central unit which is linked to all the various sensors in a car's engine measure fuel tank, car body, air-conditioning, etc. and provides real-time data that can be processed. The OBD- II adapter transmits the data from the ECU of the car to the Android application via Bluetooth. The android app processes this hexadecimal data to determine the Diagnostics Trouble Codes or Fault codes and displays it to the user. The android application is also capable of profiling the driver based on GPS co-ordinates fetched using the GPS sensor of the smartphone. The app formulates a Driver Score mathematically depicting the probability of the driver meeting with an accident and displays it to the user. The smartphone can be accessed by the end user even in real

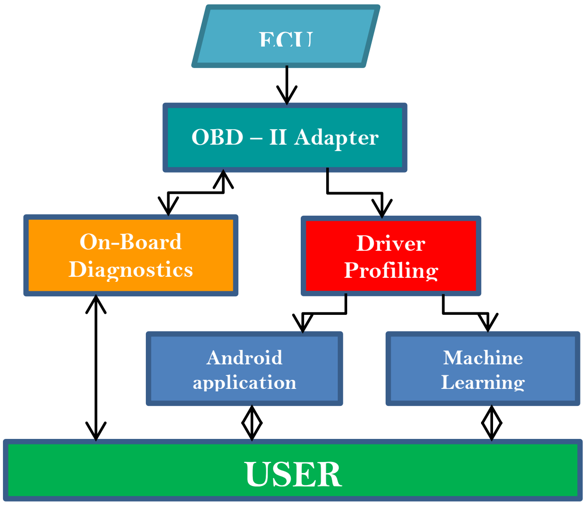

Fig- 2: Project Block Diagram

time, to know the car's condition and see how his/ her driving skills are rated and can improve on their skills. An alternate, more detailed Driver Profiling approach is modeled using ML & DA techniques. The driver score is formulated based on various OBD parameters obtained from a Comma Separated Values (CSV) file provided by a third-party android app "AndrOBD". This data is fed to Jupyter Notebook which applies the ML & DA techniques to interpret the driver's driving behaviour both visually and analytically.

4. SOFTWARE AND HARDARE DESCRIPTION

4.1 OBD- II Adaptor

On-Board Diagnostics (OBD) is a vehicle's built-in self-diagnostic system. It indicates error via the 'malfunction indicator light. OBD-II runs on CAN bus in majority of vehicles today. The OBD-II system can be accessed via an OBD-II 16-pin connector found within 0.61m of the steering wheel. There were multiple versions of OBD until they standardized it to OBD – II. Modern OBD implementations are all standardized and are all digital communication ports that offer a real-time data with advantage of standardized series of Diagnostic Trouble Codes (DTC), which helps to identify the issue right away and the remedy for the malfunction.

Fig- 3: OBD-II ADAPTOR

4.2 Android Studio

Android Studio provides the fastest tools for building apps on every type of Android device. World-class code editing, debugging, performance tooling, a flexible build system, and an instant build/deploy system all allows to focus on building unique and high-quality apps. The intelligent code editor built on intelliJ is capable of advance code completion, refactoring and code analysis. With Gradle, Android Studio offers high-performance build automation, robust dependency management, and customizable build configurations. Android Studio provides a unified environment to develop apps for Android phones, tablets, Android Wear, Android TV, and Android Auto.

4.3 Google APIs

Google APIs is a set of application programming interfaces (APIs) developed by Google which allow communication with Google Services and their integration to other services. Examples of these include the functionality of the existing services. The APIs provide functionality like analytics, machine learning as a service (the Prediction API) or access to user data (when Search, Gmail, Translate or Google Maps. Third-party apps can use these APIs to take advantage of or extend permission to read the data is given). Another important example is an embedded Google map on a website, which can be achieved using the Static maps API, Places API or Google Earth API.

4.4 Jupyter Notebook

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Its applications include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more. Jupyter Notebook can connect to many kernels to allow programming in many languages. By default, Jupyter Notebook ships with the IPython kernel. Project Jupyter's operating philosophy is to support interactive data science and scientific computing across all programming languages via the development of open source software.

5. PROJECT IMPLEMENTATION

An android app is fabricated for the two stages, i.e., On-Board diagnostics and Driver Profiling, A Machine Learning model is developed as an alternative approach to profile driver behaviour based on engine parameters collected from the car ECU.

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 06 |June 2020 www.irjet.net p-ISSN: 2395-0072

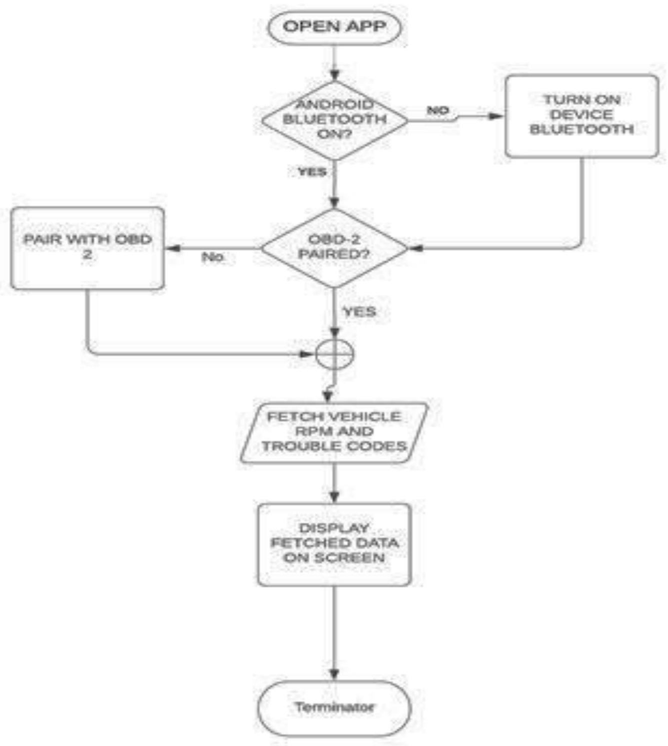

Fig- 4: On-Board Diagnostics Flow Chart

5.1 Android Application

5.1.1 On-Board Diagnostics

The workflow of the Android app for On-Board Diagnostics is represented diagrammatically, in Fig 4. To establish a connection with the vehicle's ECU, the OBD has certain set of protocols which are run in order to communicate with the ECU via bluetooth. These protocols are manufacture specific and only one or two may be applicable for the vehicle, and the protocols which best facilitates communication with the vehicle ECU is preferred. After connecting to the ECU via the OBD-2 through bluetooth the ECU is scanned by the Android App to search for any malfunctioning in heat, temperature, fuel and pressure sensors in the engine and generate respective fault codes or Diagnostic Trouble codes (DTC). These DTCs are fetched from the ECU, if present and this data is displayed on the screen of the user's android device.

5.1.2 Driver Profiling

The workflow of the Android app to achieve the objective to profile the driver based on risk factors such as speed, acceleration and retardation is shown diagrammatically in Fig 5. Once the user chooses to profile his/ her driving behaviour and starts the trip, the app fetches live GPS co-ordinates of the vehicle at periodic intervals of time. Using which, the distance travelled, its resulting speed, accelerati-

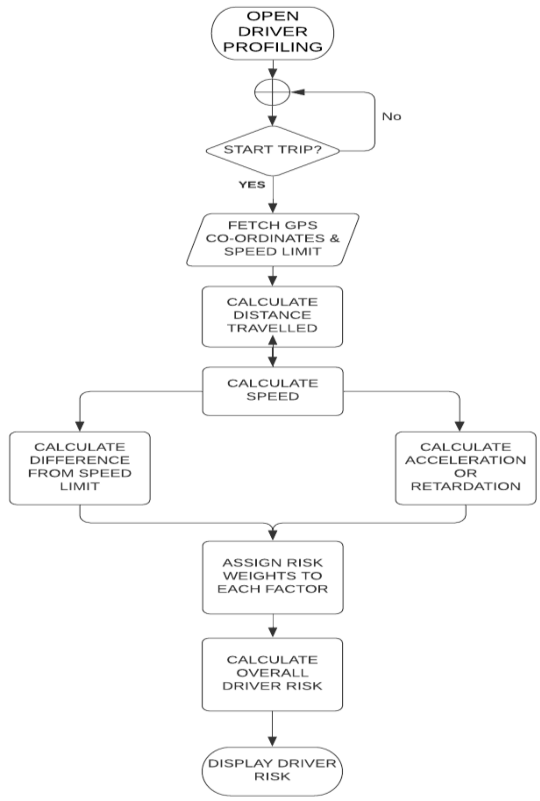

Fig- 5: Risk factor based Driver Profiling

-on/retardation is calculated at periodically. The speed limit set for the particular road is also fetched using the GPS co-ordinates of the vehicle. Based on the difference in speed of the vehicle from the speed limit risk weights are assigned based on the deviation of the value from ideality. A similar procedure is implemented to assign risk weights for both acceleration and retardation. The overall risk of the user's driving behaviour is calculated by considering how individual factors would contribute its severity leading to an accident over their ninetieth percentiles and hence a mathematically obtained Driver Score is displayed by the application.

5.2 Machine Learning & Data Analytics

An in depth, detailed, novel Driver profiling is implemented through Jupyter Notebook using ML & DA algorithms as in Fig 6. The implementation on Jupyter is made both visual and analytical considering real time vehicle engine parameters. The engine parameters considered to model Driver profiling are Engine RPM, Vehicle Speed, Throttle valve and Engine Load. All of these parameters correlate to each other positively, i.e., they are directly proportional to each other. Profiling based on vehicle speed and engine speed can only be interpreted visually using the K-means clustering plot due

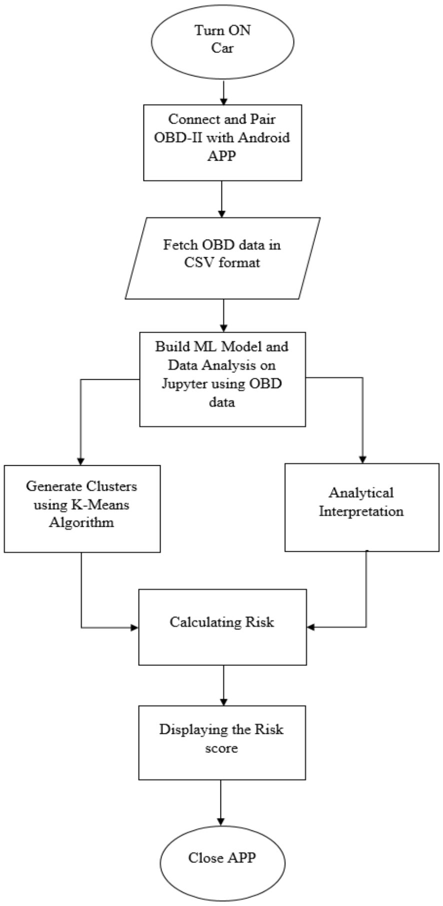

Fig- 6: Machine Learning & Data Analytics approach

to lack of data in gear ratios; whereas the relative ratio of the engine speeds & throttle valve and the engine load are realized analytically. A technique called Elbow method is used to find the optimal number of clusters having minimum within cluster error as in Fig 7.

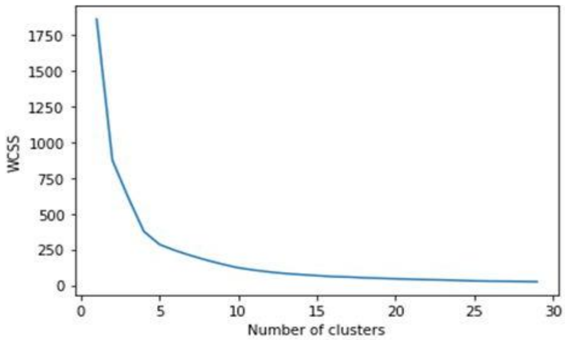

Fig- 7: Elbow Diagram

The Driver Score is evaluated by calculating the frequency of samples falling in low speed high rpm and high speed high rpm which are considered as bad behaviours. The engine parameters throttle valve, engine speed and the engine load are used to analytically interpret the driver behviour as either normal or bad. The thresholds used to interpret normal driving behaviour are: the relative ratio of the throttle valve and engine speed is maintained at between 0.9 and 1.3; the engine load is maintained between 20% and 50%. Likewise, for bad driving behaviour: the relative ratio of the throttle valve and engine speed is maintained more than 1.3 or less than 0.9; the engine load is maintained greater than 50% or less than 20% [5]. The Driver score is calculated in similar terms as to that of using the K-means by considering the instances of relative ratios which do not meet the criteria as bad behviour and its respective bad percentile are calculated. Fig 7 describes the ML & DA approach diagrammatically.

6. RESULTS

On Board Diagnostics: On-Board diagnostics is achieved through the Android app which scans the car ECU based on the data transmitted by the OBD-2 adapter. On scanning the signals which are in hexadecimal values the Android App generates a list of Diagnostic Trouble codes (DTC) or Fault codes (if problems exist with the car). The fault codes are displayed on the screen on the android app. Fig 8 shows a set of DTC displayed by the application.

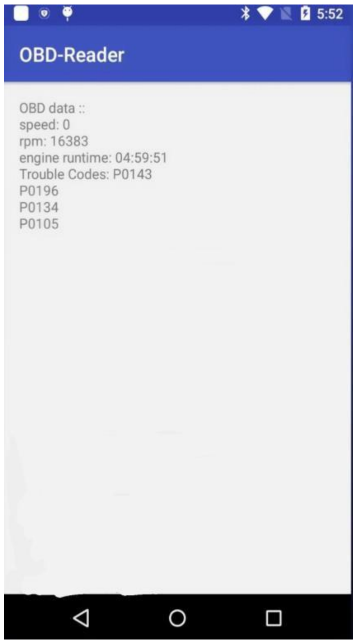

Fig- 8: Probable set of DTC

Driver Profiling: Driver behaviour is profiled in two implementations, one on the Android App along with On-Board diagnostics and the other using Machine Learning & Data Analytics on the Jupyter Notebook. By simulations, the ranges of risk scores on a scale of [0,1] describing the driver's behaviour is formulated and the driver can be classified or profiled as Safe, Modest, Moderate Risk, Risk, Reckless and Critical as in Table-2.

Table-2: Driver behaviour profiles

Driver Score	Driving Behaviour
< 0.10	Safe
0.1 – 0.19	Modest
0.19 – 0.38	Moderate Risk
0.38 – 0.56	Risk
0.56 - 0.75	Reckless
> 0.75	Critical

These ranges of risk score are also used to profile the driver's behaviour based on Driver Score obtained using ML & DA techniques.

i. **Android App**: The Android app profiles the driver based on GPS co-ordinates and a simulated driver result is shown below in Fig 9. Based on this result which is obtained mathematically and by referring to Table -2 we profile the driver's driving behaviour as moderately risky.

```
Result
$javac com/retroiceman36/Main.java

$java -Xmx128M -Xms16M com/retroiceman36/

speed risk 14.32
acceleration risk 8.26
retardation risk 9.57
Driver score = 0.308271
```

Fig- 9: Driver Score obtained through Mathematical Analysis

ii. **Jupyter Notebook**: The Machine Learning model profiles the driver both visually and analytically considering the engine parameters Engine speed, Vehicle Speed, Throttle valve and Engine Load. The scatter plot helps in visually interpreting the driving behaviour by considering engine speed and vehicular speed. The scatter plot classifies the data samples into various subgroups shown in Fig 10.

The samples grouped in High Engine speed & High Vehicle Speed and in High Engine & Low speed are considered to be bad driving behaviour and other groups are considered good. The analytical Driver Score obtained using Data Analytics is shown in Fig 11. As per the simulation the Driver Score for that dataset is 0.15 and profiles the driver's behaviour as a modest driver.

7. CONCLUSIONS

This project describes about On-Board Diagnostics and Driver Profiling to the user. The project proposes the use

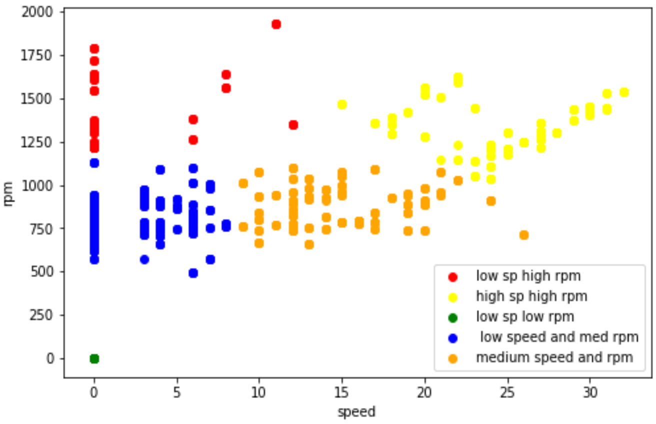

Fig- 10: Visual Interpretation of Driving Profiling

```
In [312]: c=0
          d=0
          for i in range (0,len(y['rpm'])):
              if ((y['relative rpm'][i]!=0 ) and (y['relative THROTTLE'][i]/y['relative rpm'][i]>=0.9
                  and y['relative THROTTLE'][i]/y['relative rpm'][i]<=1.3)
                  and (y['ENGINELOAD'][i]>=20 and y['ENGINELOAD'][i]<=50 )):
                  c=c+1
              else:
                  d=d+1

In [324]: e=d/(c+d)

In [326]: print('Number of times the driver was within the limits:',c)
          print('Number of times the driver was crossing the limits:',d)
          print("Driver risk score:",round(e,2))

Number of times the driver was within the limits: 3217
Number of times the driver was crossing the limits: 574
Driver score: 0.15
```

Fig- 11: Driver Score obtained through Analytical Analysis

of an Android application to fetch and display the Diagnostics Trouble Codes (DTC) from the car Engine Control Unit (ECU) and thereby facilitates self-car maintenance. The proposed android application processes hexadecimal data of the DTC stored in the car ECU and presents them in a user-readable manner to the user. The proposed android application is also capable of providing a simple, immediate and cost-effective profile of the driver's behaviour. This proposed method is by means of tracking GPS co-ordinates of the moving car resulting in a Driver Score. The Driver Score defined as the probability risk of the driving behaviour leading to an accident for the observed trip for this approach. This project also proposes an alternate, detailed and novel approach to profile the driver visually and analytically using Machine Learning and Data Analytics techniques. The proposed driving behaviour analysis method utilizes OBD interface to collect a number of critical driving operation data, i.e., vehicle speed, engine speed (RPM), throttle position, and calculated engine load. The driver behaviour is profiled visually using K-means clustering algorithm along with the Elbow method to engine speed and vehicular speed. The visualization assists the user to interpret bad driving behaviours. The driver's behaviour is profiled analytically by setting thresholds to engine speeds, throttle valve and engine load resulting in a Driver Score. The Driver Score is defined as the percentile of bad driving behaviours over the observed period. The results from all the proposed approaches assist the user in On-Board Diagnostics & Driver Profiling.

ACKNOWLEDGEMENT

We would like to express our heartfelt gratitude and respect to all those who guided us in the completion of this project. We would also like extend our heartfelt gratitude to our guide Dr. M. Rajeswari for her constant support and guidance.

REFERENCES

[1] Android-Based Universal Vehicle Diagnostic and Tracking System, 2012 IEEE 16th International Symposium – Ashraf Tahat, Ahmad Said, Fouad Jaouni, Waleed Qadamani, Communications Engineering Department, Princess Sumaya University for Technology, Amman, Jordan

[2] Vehicle Health Monitoring System – Vol. 2, Issue 5, September – October 2012, pp.1162-1167, M. Jyothi Kiran, S. Ravi Teja (IJERA).

[3] Driver behaviour profiles for road safety analysis, Science Direct Accident Analysis and Prevention 76 (2015), 118–132, Adrian B. Ellison, Stephen P. Greaves, Michiel C.J. Bliemer.

[4] Online Black Box System for Cars – International Journal of Engineering Science Invention Vol. 3 Issue 4 April 2014, PP. 06-10, Mr. Kishor Rane, Mr. Rahul Tihkule, Mr. Mohit Shinde, Prof. S.I. Nipainikar, PVPIT, India.

[5] Driving Behavior Analysis Based on Vehicle OBD Information and AdaBoost Algorithms, Proceedings of the International MultiConference of Engineers and Computer Scientists 2015 Vol I, IMECS 2015, March 18 - 20, 2015, Hong Kong, Shi-Huang Chen, Jeng-Shyang Pan, and Kaixuan Lu.

BIOGRAPHIES

Navneeth. S,
Student,
Dpt. Telecommunication Engg,
Bangalore Institute of Technology,
Bangalore, Karnataka.

Prithvil. K. P,
Student,
Dpt. Telecommunication Engg,
Bangalore Institute of Technology,
Bangalore, Karnataka.

Sri Hari. N. R,
Student,
Dpt. Telecommunication Engg,
Bangalore Institute of Technology,
Bangalore, Karnataka.

Thushar. R,
Student,
Dpt. Telecommunication Engg,
Bangalore Institute of Technology,
Bangalore, Karnataka.

Dr. M. Rajeswari,
Associate Professor,
Dpt. Telecommunication Engg,
Bangalore Institute of Technology,
Bangalore, Karnataka.
Qualification: M.E, Ph.D. Publications: 16.