International Journal of Advanced Science and Technology
Vol. 29, No. 7s, (2020), pp. 4429 - 4438

Driver Helper: A Mobile Based Application to Predict Driver Behaviour Using Classification Techniques

A. Christy[1], A.Jesudoss[2], M.S.Roobini[3]
[1,2,3] Department of Computer Science, Sathyabama Institute of Science and Technology, Chennai, India

ABSTRACT
Self-driving cars can reduce the number of accidents in comparison with the cars driven by a driver. According to a research 80% of automobile accidents occurs due to human error, which includes looking but not seeing, texting, internal distraction as well as external distraction. Every year approximately 1.3 million people are killed by road accidents. Monitoring the behaviour of driver and predicting the chances of occurrences of any mishap remains a challenge. Sensor based tracking and face-based tracking system using cameras can be followed in order to detect driver distraction in an efficient manner. When a driver can adopt to drive according to the vehicle performance, road structure and varying climatic situations, his driving will be safe. When he fails to adopt to the environment, the level of risk increases. It is required to monitor and predict the behaviour of driver. To overcome all these issues, we have developed an application named DRIVERHELPER which can virtually assist the driver by corelating the camera data with the sensor data. 4 sensors such as GPS, accelerometer, gyroscope and magnetometer were used for gathering real-time data. Data related to Harsh Acceleration, Harsh Braking, Out of Max speed, Harsh turn, Harsh Left turn and Harsh Right turn were studied. A classification task is performed with K-Nearest Neighbour (KNN), Support Vector Machine (SVM) and Decision Tree for filtering out the noise and taking the relevant data for prediction. It is found that decision tree acts as a best classifier and a mobile alert is send to the driver to bring back his attention..
KEYWORDS*: Sensor, Accelerometer, Gyroscope, KNN, SVM, Decision Trees, etc.*

INTRODUCTION

According to the World Health Organization, 1.24 million people die due to road accidents every year. The heart of any vehicle relies on the amount of assistance provided to the driver by the vehicle's support system. Autonomous cars provide technical solutions to tackle heavy traffic and congested locations efficiently. Internet of Things (IoT) is a networking paradigm designed to connect any devices for pervasive object identification data capture with intelligent information processing. According to a recent survey around 1,50,000 traffic accidents occurs per year and half a million Indians die due to accidents in highways. In-order to avoid the occurrence of accidents the highway department has placed the signboards. But accident occurs if signboards are not properly tracked.

According to a survey released by Cisco, by 2020 around 50 million things will get connected with the Internet. Also, there is a prediction that the IoT could produce profits related to projects which costs around $19 trillion. Collecting and analyzing real-world data in the IoT environment, assists in decision making. In-order to illustrate the functionality, consider a self-driving car. Important functions like maintaining the pressure, observation of the temperature and working of components inside a car done with the sensors which are available inside the car. These sensors guide the driver to take immediate action in cases of emergency.

Driver behavior profiling deals with the calculation of safety scores collected based on the data collected from the driver's driving pattern. The process is initiated by gathering data concerning the usage of acceleration, braking, steering, speed as well as location. The data is normalized, and a computational model is applied to generate a safety score. Sensors such as accelerometers, gyroscope, magnetometer as well as GPS are equipped inside vehicles in order to monitor driver's data. As the usage of GPS greatly increases battery consumption, accelerometer in smart phone can also be kept as a choice to data collection [8, 10, 11]. These days, smart phones are embedded with sensors which can provide data for safety score calculation as well as alerting the drivers from the occurrences of accidents.

LITERATURE SURVEY

Mercedes Ayuso et al (2014) had developed a model for implementing the insurance policy pay-as-you-drive. The driver's driving pattern is identified using Global Positioning System considering speed and distance [4]. According to Ellen De Pauw et al in 2014 speedy driving can be reduced by fixing cameras in accidents prone zones and in places across highways. As per the study performed by the International Symposium, 30% of accidents happen due to fatigue, sleep disorder and lack of concentration of the driver. Driving assistant systems available in cars like Ford, Skoda, Volkswagen finds drowsiness of driver considering steering movements, irregular braking system, lane movements, etc and providing a beep to the driver. Driver drowsiness can also be found using heart rate analysis and with the eye movement of the driver using a camera [3]. Damian Salapatek et al (2017) has developed a vision-based approach by implementing an image processing algorithm which comprise of main component analysis, neural networks, gabor filters, frequency spatial methods. Facial features can be obtained effectively using Gabor filters [2]. Colic et al in 2014 states that driver drowsiness causes road accidents up to 7% and critical collisions up to 18%. An analysis by the World Health Organization (WHO) reveals that among the persons facing road accidents 40% of the people are below 30 years of age [1]. Esra Vural et al (2008) has analyzed the facial expressions of the drivers while blinking and yawning using machine learning techniques like Adaboost and multinomial ridge regression [5]. Ian Y. Noy et al in 2018 have found that a driver can operate a vehicle and can play the role of a controller or a supervisor. Replacing the driver with state-of-the-art sensing, computing, communication, and robotic-control technologies, would eliminate human error as a causative factor [6]. Ramon Zatarain-Cabada et al (2019) has proposed an architecture for emotion classification using deep learning considering the factors like visibility constraints, improper judgment, etc which would benefit the system in alerting the driver [14]. Michael A Regan et al (2011) driver's distraction is caused by lack of attention during driving, mis-prioritized attention, neglected attention, Cursory attention and diverted attention which can be validated with event descriptions collected from accidents and live data through live video stream [9]. The major cause to unsafe driving is mobile usage, according to Pushpa Choudhary and Nagendra R. Velaga (2017). Mobile usage with simple and long conversation may create unsafe driving situation. Data is collected using questionnaire which are then modeled using statistical measures like ANOVA and Weibull AFT and the results have shown long message texting is the contributor for driver's distraction [13]. Kristie L. Young and Paul M. Salmon (2012) have classified the contributors of disaster as Technology based (Mobile phone usage or music usage), Non-Technology based (Distraction by fellow passengers or objects inside the vehicle) and External to Vehicle (Pedestrians, other traffic) etc [7]. Oliver Carsten and Marieke H. Martens (2018) expresses Human Computer interaction is essential in autonomous cars. Failure of expected action is the first factor, that the users come across. i.e., car speed is limited for a particular range and it is not possible to increase when speed limit goes up and the second factor is the presence of unexpected action, which means not decreasing speed in response to traffic light while there are no leading cars. The technology of auto cars will be critical, if the driver is not aware of the functionalities [12]. Sara Hernández Sánchez et al (2014), have deployed smart phone accelerometers for detecting the movement of vehicles using deep learning and convolution recurrent network methods. In their proposed method, mobile coordinates are converted into vehicle reference coordinates (world coordinates) [15]. The deep learning (DNN) architecture adopted, combines convolution neural networks to estimate the movement of vehicles. The movement of the vehicle is gathered as mobile coordinates. These mobile coordinates are converted into world coordinates which can then be applied to categorize the driving patterns such as application of accelerator, brake, harsh turn, etc. Security enhancement and effective testing of this system will be necessary for this system then it can be implemented with prevention measures as described [16- 22]. From the literature survey, it is evident that an automated system to alert the driving in case of critical situation which can prevent the occurrences of faults and crashes. In this approach, non-invasive sensors are used for collecting live data and based on the method of applying acceleration and brake, the driver is alerted as a preventive measure.

SYSTEM ARCHITECTURE

This work is modeled as a multi-label classification and prediction problem by studying driver behavior. The objective is to identify the type of sensors, the classification and prediction algorithms and to identify the events occurring through the sliding window. The system for modeling the system is depicted in Fig.1. In this implementation, four sensors, GPS, accelerometer, Gyroscope and Magnetometer in order to monitor and control the vehicle from varying speed, applying brake and variation of lane changes. An ultrasonic can monitor an angle of 45 degrees and so one is enough to sense all objects in front. GPS is a satellite based global positioning system, which is used to find the position of location on earth using latitude and longitude values. Smart Phone Hardware sensor converts physical quantity which could be interpreted using devices. Accelerometer is an electromechanical device which measure acceleration forces which could be static or dynamic. They are measured in x, y and z directions as m/s2 values. The Gyroscope can either detects or measures the orientation of the device from its angular rate. It functions adopting the principles of angular momentum and is represented in 3-axis by pitch, roll and yaw. The Magnetometer measures the magnetization or magnetic strength of materials like Ferro magnet. The pitch angles are received using sensor fusion and X-axis of the Accelerometer. Lane turn or change is recorded using yaw and Y-angles of the accelerometer. The location and speed limit is obtained through GPS. Harsh brakes are observed if a brake is applied to more than 20 mph within 2 sec. The GPS provides information to the sensor regarding the latitude, longitude value, max speed allowed, traffic details, etc. The data collected by the sensors are converted to normalized world coordinates. This conversion is mandatory in-order to obtain device coordinates inside the vehicle.

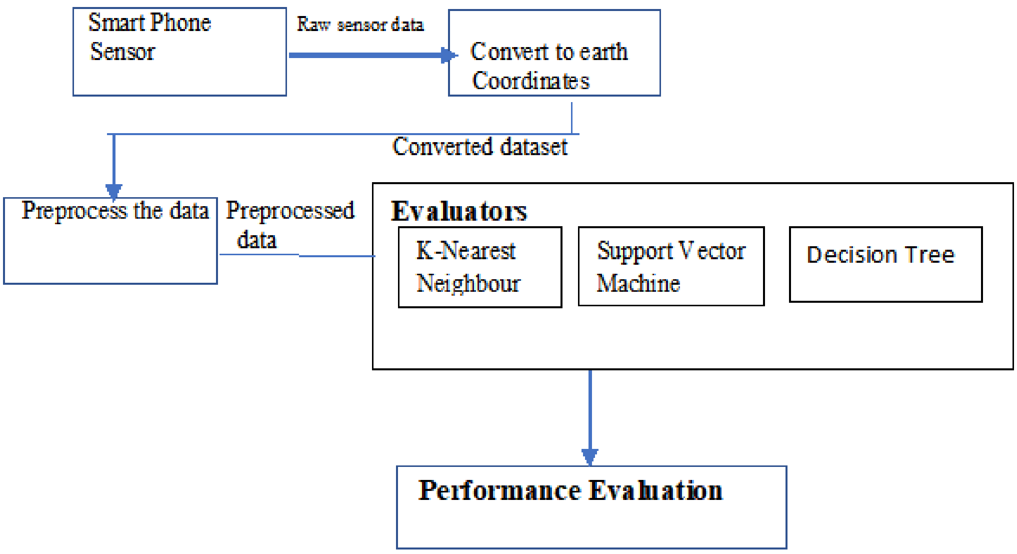


Fig.1 System Architecture using vehicle sensors

Converted sensor data is then preprocessed. The attributes best suited for prediction are obtained using methods such as KNN, SVM and Decision Tree. An android app is deployed which would communicate the driver to show his score and to alert him by providing signals on unsafe circumstances. Based on the sensor readings, Table 1 depicts the standards that exhibit the conditions for every safe and unsafe drive event.

Driving Event	Data Used (Accelerometer)	Threshold	Data Used (Sensor Fusion)	Threshold
Self acceleration	X-axis data	1.3 m/s^2 to 2.5 m/s^2	Pitch	-0.08 to -0.12 rad/s
Self deceleration	X-axis data	-1.3 m/s^2 to -2.5 m/s^2	Pitch	0.08 to 0.12 rad/s
Self left turn	Y-axis data	-1.8 m/s^2 to 3.0 m/s^2	Yaw	0.10 to 0.30 rad/s
Safe right turn	Y-axis data	1.8 m/s^2 to 3.0 m/s^2	Yaw	-0.10 to -0.30 rad/s
Hard acceleration	X-axis data	>2.5 m/s^2	Pitch	<-0.12 rad/s
Hard acceleration	X-axis data	<-2.5 m/s^2	Pitch	>0.12 rad/s
Sharp left turn	Y-axis data	< 3.0 m/s^2	Yaw	>0.30 rad/s
Sharp right turn	Y-axis data	> 3.0 m/s^2	Yaw	<-0.30 rad/s

Table 1 : Threshold values for Safe and Unsafe drive events

Source : "Beyond the Internet of Things Everything Interconnected"

GENERATION OF FEATURE VECTOR

The types of sensors supported by Android smart phones include the Accelerator (Acc), Linear Accelerator (LinAcc), Magnetometer (mag) and Gyroscope (Gyr). The force of gravity which is not included in LinAcc is available with LinAcc. The magnetometer functions like a magnet and measure the force applied in a magnetic field in terms of micro-Tesla(μT). The rate of rotation of the axes is measured by the gyroscope with the unit radians per second (rad/s). The sensors provide raw data from all (x, y and z) axis. The data from this input is converted to the world coordinates. The data are preprocessed, and the behavior of the system is studied using KNN, SVM and Decision Tree method.
Raw sensor data record the values of 3 axes and time stamp, which indicates the time of data collection. The sensor data cannot be passed to the classifier, as it may contain missing data. A sliding window with the sensor data is hence sliced to a time frame of one-second in-order to create the attributes required for classification. The process continues as the first frame is send across sliding window. A feature vector represents the details of sliding window collected from sensor data as depicted in Fig. 2. The feature vector is updated whenever a driving event happens.

Sliding Window

t_{-i}	……………	t_{-1}	t_o

Sliding Window

t_{-i}	……………	t_{-1}	t_o

Sliding Window

t_{-i}	……………	t_{-1}	t_o

Fig. 2 Sliding window representing the data frame

An instance of the feature vector is created by calculating the mean, median, standard deviation and central tendency upon the sensor data over the specified time frames. The number of features extracted depend upon the number of instances collected over the time frame. When the data flow from more than one axis, it gets appended to the existing feature vector. The structure of the feature vector is depicted in Table 2in which the class label vector value will be static until all data are collected.

ISSN: 2005-4238 IJAST

International Journal of Advanced Science and Technology
Vol. 29, No. 7s, (2020), pp. 4429 - 4438

Table 2 Attributes of Feature Vector

Mean	M_0….......M_i	$M(t_0)$ …………………$M(t_{i-1},t_0)$
Median	MD_0……MD_i	$MD(t_0)$ ………………$MD(t_{i-1},t_0)$
Standard Deviation	SD_0……SD_i	$SD(t_0)$ ………………$SD(t_{i-1},t_0)$
Central Tendency	$T1 = \frac{M(t-1)}{M(t0)}$	$T1 = \frac{M(t-1)}{M(t0)}$
Driving Event	Label	Label

In the definitions stated in Table 2, the values of i varies from 0 to n_{t-1}. ST(tj) represents the summation of Mean, Median and Standard Deviation applied with the data of frame j. $ST(t_j,t_k)$ represents the summation of data collected from j^{th} frame to k^{th} frame. Driving event is designated as the class label and it take four different values such as Timed Event, Distance Event, Network Event and System Event. The driving collects and organizes data collected between the time stamp $t\text{-}_{(nt-1)}$ and to. The features of data having similar driving pattern and time independence can be clustered together in the same data set . As soon as a driving event happens, the sensor collects raw data nt which are the instances of feature vectors. The sliding window frame records the start time of the event. If there are m instances of a driving type, then there will be m * nt feature vector instances for the same event type. This characteristic makes to capture different portions of the same event through different windows.

MOBILE SENSORS

Like sensors attached through the vehicle, mobile sensors can track the movement of the vehicle for high acceleration, brake and turn. The movement of the vehicle is represented by the mobile co-ordinates(X,Y,Z) and these values are constant with the assumption that, the position of the mobile is not altered. Once the movement is identified, its latitude and longitude corresponding to the vehicle movement is tracked for its safety. The architecture adopted for the implementation of mobile sensor is depicted in Fig. 3. From figure 3, initial stage involves vehicle movement detection. Sensor data from all x,y and z axis are gathered as it provides energy level to collect information regarding the acceleration forces applied. The accelerometers gather variations in the accelerations applied in the vehicle during its movement and as the vehicle stops, the accelerations along x, y, z directions remain a constant. The latitude angle provides the acceleration data on the move and this data is fed as input to machine learning algorithms for classifications. Once the sensor data are normalized and converted to device coordinates, data are clustered into 5 groups based on their similarity along x, y and z axis. This data along with the classification done with machine learning techniques are used to calculate the driving score. An accelerometer is used to read (x, y, z) coordinates at a fixed position inside the vehicle. An accelerometer sensor captures data during a trip, which is then segmented into a predefined size. The accelerometer can detect rash driving by observing the segmented data through the sliding window. The acceleration on the horizontal plane of driving is retrieved and checked with the defined threshold level.

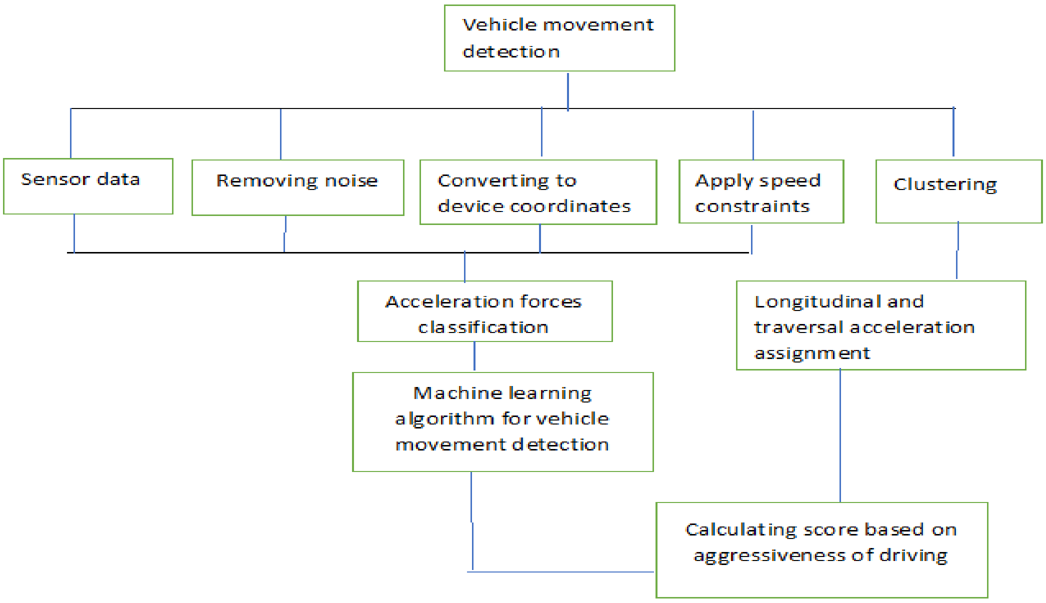


Fig. 3 Architecture using mobile sensors

This threshold level is set from the previous data available in the literature. Low pass filtering of the signals is carried out for filtering out noise of higher frequency. The accelerometer signals collected from an Android device is depicted in Figure 4, in which the X-axis moves along the width of the screen, the Y-axis along the length and Z-axis moves along the depth of the mobile.

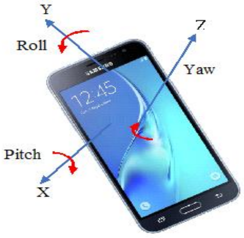


Fig. 4 Direction of acceleration signals

The reading from the acceleration position of the driver had during the trip is depicted in Fig. 5 and Fig.6 depict the status followed during the phone usage.

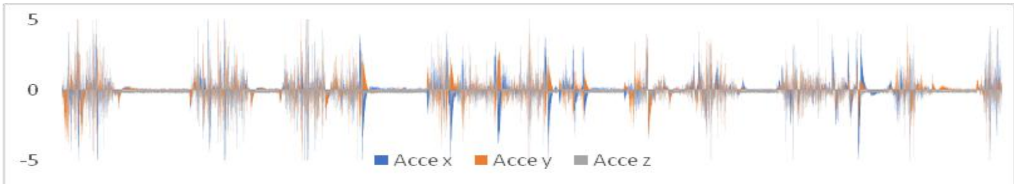


Fig. 5 Raw Accelerometer data

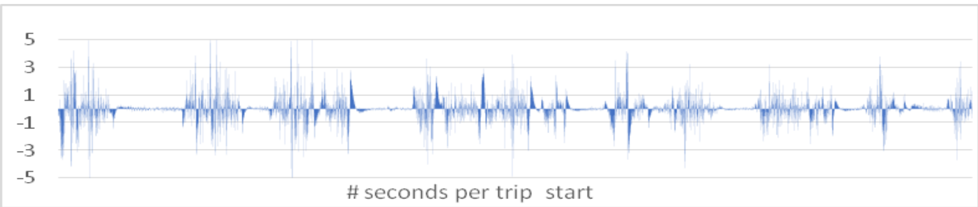


Fig. 6 Phone usage probabilities for a 8 minutes trip

The accelerometer of the mobile shows variations during the movement of the vehicle and it . The mobile can be placed in any orientation, preferably in the windshield. The difference between phone usage and vehicle movement recording can be identified by the sensor analytics pipeline. Figure 6 predicts the sensor data obtained in a 8-minute car trip along with the probability of phone usage received through the classifier.

RESULTS AND DISCUSSION

Driver monitoring and analysis is the process of automatically extracting driver data (eg., latitude, longitude, speed, acceleration, etc) and computes safety score. For collecting the sensor data, an Android application is deployed in-order to gather sensor data from certain events. The start and stop timings of the driving events are also collected. The experiments are conducted in five trips of 20 minutes as an average. The car is moved across a school and hospital and it was fixed to move at a rate of 5 km/hour. The car used for experimentation is SantroSportz and a smart phone used is Honor 6X having Android version. The smartphone with the app installed was fixed in the windshield of the car. The sampling rate of the motion sensors ranges between 70 and 100 Hz. Table 3 represents the number of driving event types considered with their instances as in table 3.

Table 3. Types of Driving event with Instances

Type of Driving Event	# of instances
Harsh Acceleration	5
Harsh Braking	7
Out of Max speed	8
Harsh Turn	7
Harsh left turn	8
Harsh right turn	6

Different datasets are gathered and using varying nt values. The data are achieved by different types of sensors trained and classified using KNN, SVM and Decision trees. Features are evaluated for Harsh turn with their Semantic Similarity, Out of Max Speed by geographic similarity (Latitude and longitude) and the behavior of the sensor data with response to Harsh Acceleration, Out of Max Speed and Harsh Turn are depicted in Fig. 7, 8 and 9. The classification of acceleration forces is done with SVM, Decision Trees and KNN. Due to the wide availability of data, the system should not suffer from overfitting. The purpose of using classification techniques is to avoid the existence of invalid patterns that can occur while driving. For getting the ground truth, the data from accelerometer and gyroscope sensors from mobile phones are retrieved. Once the data is gathered, they are checked for Harsh Acceleration, Harsh Braking and Harsh turn.

In the case of a mobile sensor, the mobile inside the vehicle grabs data in all positions making the process more complex. Preprocessing and classification is performed by treating the reading from raw accelerometers as a baseline. It is then compared with the same signals after filtering out the noise occurred from the accelerations by projecting the accelerations to a plane perpendicular to the gravity and then estimation is done. In-order to evaluate the traversal and longitudinal forces of acceleration, it is required to find out the true direction of vehicle movement. This is achieved by taking 3 vectors related to turns (done by accelerometer reading), acceleration and deceleration. Accelerometer and gyroscope data is sampled at a rate of 25 Hz and GPS data is sampled at 1 Hz. The GPS data is not used for prediction, it is used only to find out of max speed.

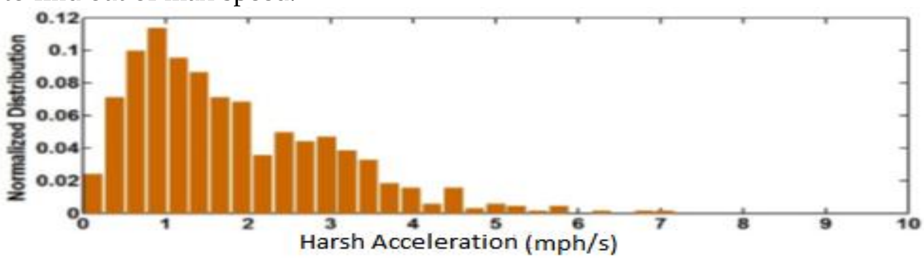


Fig. 7 Sensor data response to Harsh Acceleration

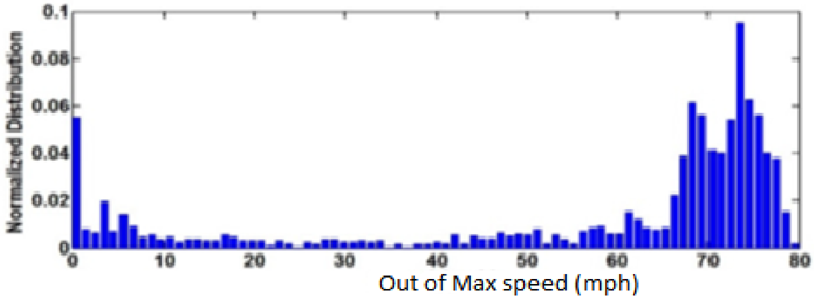


Fig. 8 Sensor data response to Out of Max Speed

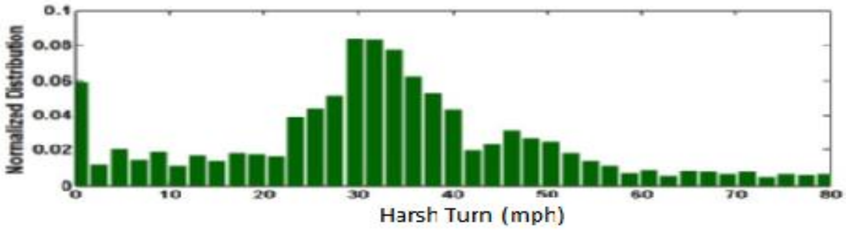
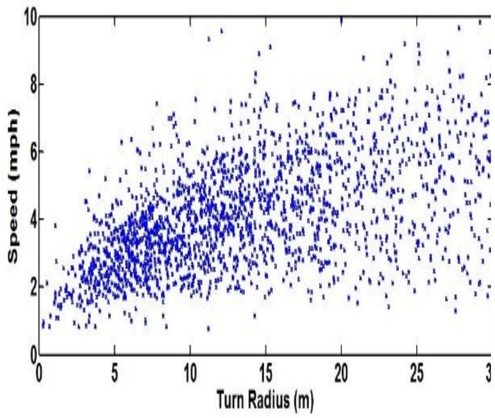


Fig. 9 Sensor data response to Harsh Turn

Performance of decision tree with confidence level 0.1, maximum depth 10 and minimal gain 0.01 the accuracy of decision tree is depicted in Table 4 along with KNN and SVM values. Results have shown that Decision tree trained with Information gain has been a good classifier. The mobile inside the vehicle grabs data in all positions making the process more complex. Preprocessing and classification are performed by treating the reading from raw accelerometers as the baseline. It is then compared with the same signals after filtering out the noise occurred from the accelerations by projecting the accelerations to a plane perpendicular to the gravity and then estimation is done.

Table 4. Performance of Classifiers KNN, SVM and Decision Tree

Dataset	KNN	SVM	Gain Ratio	Information Gain	Gini Index
Mobile Sensor	0.7148	0.813	86.20	87.0	86.92
Vehicle sensor	0.673	0.782	85.15	86.12	84.32



Fig, 10 Right Turn

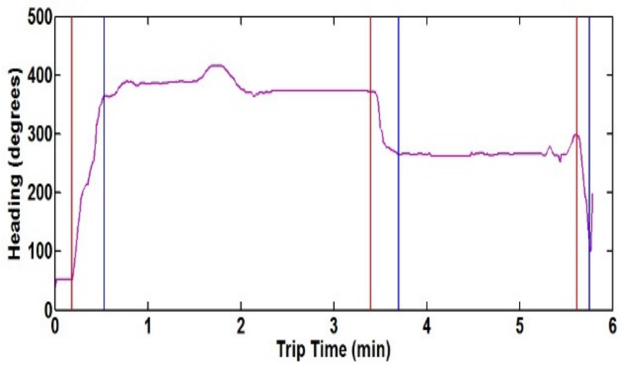


Fig. 11 Trip Time of sliding window

The Trip radius and trip time for a sliding window are depicted in Fig. 10 and Fig. 11 for the classified data. The parameter nt which indicates the number of features in the sliding window becomes the predictor for rashness in driving. As specified in Table 1, soft right turn takes on threshold 1.8 m/s2 to 3.0 m/s2. In this experimentation, 12 sets of data were executed using different nf values. Models were trained and tested with 12 different random values. Higher nf values give smooth drive. The aggressive turns fall above the threshold 0.6 whereas the non-aggressive turns lie between 0.3 to 0.6.

CONCLUSION

The gyroscope and the accelerometer play a major role in collecting raw data from sensors. All sensors perform well when data is collected from all three axes. The left turn can be detected correctly from the Z-axis of the gyroscope. The decision tree acts as a good classifier in classifying data which is obtained through the sliding window. Machine learning algorithms can perform well when the value received through the sliding window is higher (i.e., higher nt). The raw sensor signals from the vehicle sensors are fetched without any pre-processing. From each component of the accelerometer gravity is removed and is projected to a plane perpendicular to the gravity.

REFERENCES

1. Colic, A., Marques, O., & Furht, B. (2014). Driver drowsiness detection: Systems and Solutions (p. 55). Springer International Publishing.
2. Salapatek, Damian & Dybała, Jacek & Czapski, Paweł & Skalski, Paweł. (2017). Driver Drowsiness Detection Systems. Proceedings of the Institute of Vehicles. 3. 41-48.
3. De Pauw, E., Daniels, S., Brijs, T., Hermans, E., & Wets, G. (2014). An evaluation of the traffic safety effect of fixed speed cameras. Safety science, 62, 168-174.
4. Mercedes Ayuso, MontserratGuillénAna María, Pérez-Marín (2014), “Time and distance to first accident and driving patterns of young drivers with pay-as-you-drive insurance”, Accident Analysis and Prevention,Vol. 73, Pp. 125-131.
5. Esra Vural, Mujdat Cetin, Aytul Ercil, Gwen Littlewort, Marian Bartlett and Javier Movellan (2008), “Machine Learning Systems for Detecting Driver Drowsiness”, In-Vehicle Corpus and Signal Processing for Driver Behavior, Pp. 97-110.
6. Ian Y. Noy, David Shinar and William J. Horrey (2018),”Automated Driving:Safety Blind Spots”, Safety Science, Vol. 10, Pp. 68-78.
7. Kristie L.Young and Paul M. Salmon (2012),”Examining the relationship between driver distraction and driving errors: A discussion of theory, studies and methods”, Safety Science, Vol. 50, Issue 2, Pp.165-174.
8. Maurer, M., Gerdes, J. C., Lenz, B., & Winner, H. (2016). Autonomous driving. Berlin, Germany: Springer Berlin Heidelberg, 10, 978-3.
9. Michael A Regan, Charlene Hallett and Craig P. Gordon (2011), “Driver distraction and driver inattention: Definition, relationship and taxonomy”, Accident Analysis and Prevention 43, Pp. 1771-1781
10. Muhammad Rizwan Ullah, Muhammad Aslam, Muhammad Imran Ullah and Martinez-Enriquez Ana Maria (2017),” Driver’s Drowsiness Detection Through Computer Vision: A Review”, 16th Mexican International Conference on Artificial Intelligence, Pp 272-281.
11. Nanxiang Li and Carlos Busso (2013),” Modeling of Driver Behaviour in Real World Scenarios using Multiple Noninvasive Sensors”, IEEE Transactions on Multimedia, Pp. 1-25.
12. Oliver Carsten and Marieke H. Martens (2018),” How can humans understand their automated cars? HMI principles, problems and solutions”, Cognition, Technology & Work, Vol. 21, Issue1, Pp.3-20
13. Pushpa Choudhary and Nagendra R.Velaga (2017), “Modelling driver distraction effects due to mobile phone use on reaction time”, Transportation Research Part C: Emerging Technologies,Vol. 77, Pp.351-365.

14. Ramon Zatarain-Cabada, Maria Lucia Barron-Estrada, Francisco González-Hernandez and Hector Rodriguez-Rangel (2019), " Emotion Recognition Using A Convolutional Neural Network", Advances In Computational Intelligence, Pp 208-219.
15. Sara Hernández Sánchez, Rubén Fernández Pozo, and Luis A. Hernández Gómez (2018)," Estimating Vehicle Movement Direction from Smartphone Accelerometers Using Deep Neural Networks", Sensors (Basel). 2018 Aug; 18(8), doi: 10.3390/s18082624.
16. Vaithyasubramanian, S. (2017). Array CAPTCHAs (Completely Automated Public Turing test to tell Computer and Human Apart). Asia Life Sci.-The Asian Int. J. Life Sci., 1, 247-254.
17. Saikirthiga and S. Vaithyasubramanian, "Review on development of some strong visual CAPTCHAs and breaking of weak audio CAPTCHAs," 2016 International Conference on Information Communication and Embedded Systems (ICICES), Chennai, 2016, pp. 1-4. doi: 10.1109/ICICES.2016.7518939
18. Vaithyasubramanian. S, Lalitha. D, Christy. A "Study on user credentials in multi factor authentication using statistical analysis", Journal of Advanced Research in Dynamical and Control Systems, Volume 11, Issue 6, 2019, Pages 982-988.
19. Sai Kirthiga, Vaithyasubramanian. S "Survey on secured online authentication and defence technique against 3rd party human attacks using CAPTCHA" Journal of Chemical and Pharmaceutical Sciences, Volume 9, Issue 4, 1 October 2016, Pages 2009-201.
20. Vaithyasubramanian, S., & Christy, A. (2015). Authentication using string generated from chomsky hierarchy of formal grammars. International Journal of Applied Engineering Research, 10(8), 6269-6273.
21. Vaithyasubramanian, S., & Christy, A. An Analysis on 1-Step Transition Probability Matrix and 2-Step Transition Probability Matrix of Markov Passwords. International Journal of Applied Engineering research, 9(20).
22. Natarajan Rajarajeswari, Vaithyasubramanian. S "A white paper attempt on defining key process and benefits of effective defect management system in agile based projects-effective defect management system in agile methodology", International Journal of Engineering and Advanced Technology, Volume 8, December 2018, Pages 201-205.