Smart Accident Detection and Alert System

Uma N, Saktheeswari R, Indumathi A, Sugacini M, Kavishree S

Department of Information Technology, Sri Venkateswara College of Engineering,
Sriperumbudur, India
E-mail: numa@svce.ac.in, sakthi@svce.ac.in, aindumathi@svce.ac.in,
msugacini@svce.ac.in, kavishrees@svce.ac.in.

Abstract: - **Due to the sheer rise in vehicle accidents, there is an increasing demand for effective early vehicle crash detection and alert systems. We propose a vehicle crash detection system with sophisticated components such as raspberry pi, sensors, apis, web browser automation and communication modules to improvise vehicle crash detection performance. The existing concept presents eXtreme Gradient Boosting, a Machine Learning approach for identifying accidents utilizing a set of real-time data. Working on this method considering single dataset such as upstream and downstream averages, results in a lower efficiency. While most existing vehicle crash detection systems depend on single modal data, our proposed vehicle crash detection system uses an ensemble machine learning model based on multi modal data such as accelerometer, gyroscope, and shock sensor data. In order to verify that an accident has happened, a message is sent to the user, if the user does not respond to the call within 30 seconds, our detection system immediately conveys the exact location of the accident occurred to the nearest hospital within 82 seconds via voice message. Also another voice message enclosing the information about the location of the accident along with the hospital name will be sent to the trusted person specified by the user. The experimental results indicate that the proposed vehicle crash detection system performs significantly better than single classifiers. With the help of this application, we preserve the state of humanity and generate harmony in our society.**

Keywords - Deep Learning, Machine Learning, Convolution Neural Network, Lesion, YOLOV5.

1 INTRODUCTION

In recent years, traffic accidents have become increasingly common. As shown in figure 1.1 in 2015, the global status report on road safety from the World Health Organization noted that approximately 1.25 million people die each year from traffic accidents worldwide. Furthermore, accidents indirectly cause nonlethal injuries for 20 million to 50 million people, and 90% of road traffic accidents occur in low- to moderate income countries. Due to the low levels of emergency services and medical standards in many low- and moderate-income countries, the mortality rate from accidents in such countries is significantly higher than that in developed countries, and the economic losses caused by traffic accidents are extensive.

Recently, many studies have confirmed that rapid emergency rescue responses to crashes result in higher survival rates. For example, in China, approximately two-thirds of traffic accident victims died in 2016 because they did not receive timely and effective assistance.

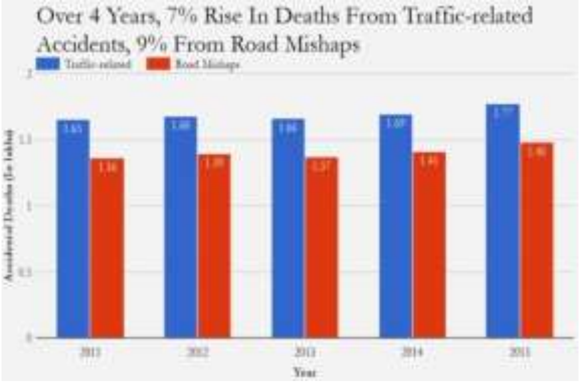


Figure 1. Rise in Deaths from traffic-related Accidents and Road mishaps

A statistical data of vehicle accidents describing the number of deaths and the change in percentage is shown in figure 1. Moreover, most of the injured patients died either at the accident scene or while in-transit to a hospital; these trips averaged approximately 30 minutes. The period between an accident occurrence and arrival at a medical facility for those who sustained injuries was divided into four stages: dispatch time, response time, scene time, and transport time. Mortality during these four stages was compared in urban and rural settings. The statistics indicated that when an accident is detected immediately, the response time required for emergency services can be greatly reduced, and the survival rate of traffic accident victims increases. Additionally, most individuals involved in traffic accidents receive appropriate assistance from drivers, passengers, or other people. However, when a traffic accident occurs in the suburbs or the driver is the only person in the vehicle and the crash results in loss of consciousness, no one is available to notify the proper authorities within the ‘‘golden window period’’ for medical treatment, and this delay reduces the survival rate.

Whenever an accident occurs it is detected by a machine learning algorithm by using a shock sensor and accelerometer sensor. Once an accident is detected the communication process gets started in which a voice API sends a voice message to the hospitals and to the phone numbers which are previously saved with the data such as geological data, timestamp, direction of the hit. The shock sensor has a threshold value if it breached an alert is sent to the driver with an option of accident occurred or accident not occurred if the driver does not reply for more than a defined time then the algorithm considers it as an accident and initiates to execute the communication process. The algorithms achieve an overall true-positive rate of 68.8% for accident predicted and 72.2% for correct accident detection.

The high demand of automobiles has also increased the traffic hazards and the road accidents. Life of the people is under high risk. This is because of the lack of best emergency facilities available in our country. An automatic alarm device for vehicle accidents is introduced in this work. This design is a system which can detect accidents in significantly less time and sends the basic information to the first aid center within a few seconds covering geographical coordinates, the time and angle in which a vehicle accident had occurred.

2 PROPOSED ARCHITECTURE

We propose a vehicle crash detection system with sophisticated components to improvise vehicle crash detection performance. While most existing vehicle crash detection systems depend on single modal data, the proposed vehicle crash detection system uses an ensemble deep learning model based on multi-modal data. To verify that an accident has happened, a message is sent to the user, if the user does not respond to the message within 30 seconds, our detection system immediately conveys the exact location of the accident occurred to the nearest hospital within 10 seconds via voice message. Also another voice message enclosing the information about the location of the accident along with the hospital name will be sent to the trusted person specified by the user. The experimental results indicate that the proposed vehicle crash detection system performs significantly better than single classifiers.The architecture diagram in figure 2. Illustrates the components of the system. The Shock sensor SW-420 combined with accelerometer sensor works to initiate the accident detection process. The Impact sensor triggered when the airbag gets released during an accident, parallel shock sensor reads vibration data throughout the body of the car to detect vibration caused by the collision. Then data from the shock sensor and impact sensor is transmitted to Raspberry pi, then the Accelerometer ADXL 345 data, to find the direction of the accident. The microcontroller is programmed to perform the required actions. After the occurrence of the accident a prompt is sent to the victim for confirmation of incident with the option of declining if the victim declines then it is considered as no need of medical emergency and the process terminates, if the victim doesn’t declines within certain seconds the geological data from the GPS module NEO 6M the accident details is sent to the Hospital and Relatives using the preinstalled WiFi via voice message - Voice Messaging API.

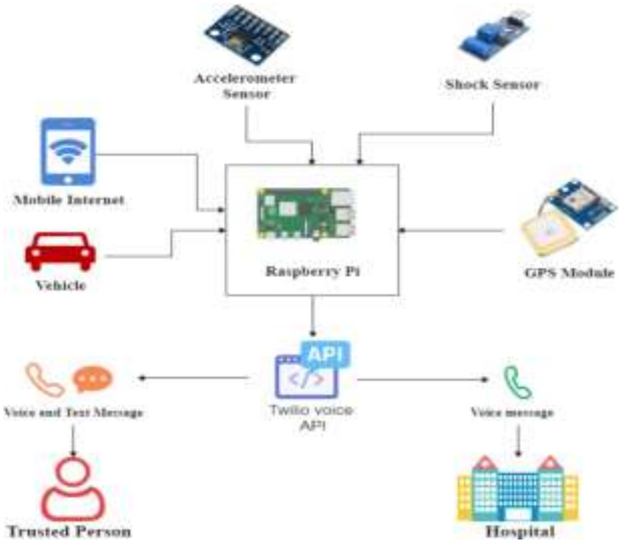


Figure 2. Smart Accident Detection System Architecture

Extreme gradient boosting – XGBoost

XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. It implements machine learning algorithms under the Gradient Boosting framework. XGBoost provides a parallel tree boosting also known as GBDT, GBM that solves many data science problems in a fast and accurate way.

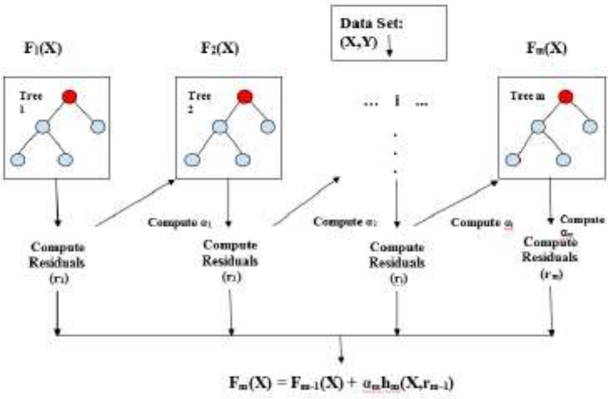


Figure 3. Work Flow of XGBoost Algorithm

Gradient boosting is a type of machine learning boosting. It relies on the intuition that the best possible next model, when combined with previous models, minimizes the overall prediction error. The key idea is to set the target outcomes for this next model in order to minimize the error. The weak learners in gradient boosting for classification are regression trees, and each regression tree transfers an input data point to one of its leafs, which carries a continuous score. XGBoost combines a convex loss function based on the difference between the predicted and target outputs with a penalty term for model complexity to minimize a regularized L1 and L2 objective function. The training process is repeated iteratively, with new trees being added that forecast the residuals or errors of previous trees, which are then integrated with previous trees to provide the final prediction. Gradient boosting gets its name from the fact that it uses a gradient descent approach to minimize loss when adding new models. A brief illustration on how gradient tree boosting works is depicted in figure 3.

Algorithm:

Input: Training set a differentiable loss function (y, $F(x)$), number of iterations M.
1. Initialize the model with a constant value:

$$F_0(x) = arg_\gamma min \sum_{i=1}^{n} L(y_i, \gamma). \qquad \text{---- (1)}$$

In Eq 1, (y_i,γ) is the loss function

2. For m = 1 to M.

i) Calculate the Similarity Score:
The similarity score of each current root node is calculated. The current root node is split into left and right nodes using this score, i.e., each data in the left node have the same similarity score. Similarly, for the right node.

ii) Calculate the Gain:
The Gain for each node is calculated using. There are two criteria for this split up. If the gain is greater than zero, the node will split. If the gain is lesser than zero, break the loop.

iii) Residuals are Objective loss Function calculated at each step.
Compute so-called pseudo-residuals:

$$r_{im} = -\left[\frac{\partial L(y_i, F(xi))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)} \quad for\ i\ =\ 1, \ldots.., n. \qquad \text{------- (2)}$$

In Eq 2, $Fm{-}1(x)$ denotes , similarity score of the previous node. For each iteration new set of residuals is calculated using previous residual values,
(i.e)
New Residuals = *Old Residuals* + ρΣ *Predicted Residuals* -------(3)
where ρ is the learning rate.

iv) Compute Gain as a Objective function:

$$Gain \ = \ \frac{1}{2}\left[\frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{(G_L+G_R)^2}{H_L+H_R+\lambda}\right] - \gamma \qquad \text{--- -----(4)}$$

v) Fit a base learner (e.g. tree) $h(x)$ to pseudo-residuals, i.e. train it using the training set. $\{(x_i, r_{im})\}_{i=1}^{n}$
vi) Compute multiplier γ_m by solving the following one-dimensional optimisation problem:

$$\gamma_m = arg_\gamma min \sum_{i=1}^{n} L(y_i, F_{m-1}(x_i) + \gamma h_m(x_i)) \qquad \text{------ (5)}$$

In Eq 5, γ_m is the L1 Regularization Value.

vii) Update the model:

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x). \qquad \text{---------(6)}$$

3. Output $F(x)$.
With Eq 6, the Resultant Formula for XGBoost is ,

$$F_m(X) = F_{m-1}(X) + \gamma_m h_m(X, r_{m-1})$$

where α_i and r_i are the regularization parameters and residuals computed with i^{th} tree respectfully, and is a h_i function that is trained to predict residuals, r_i using X for the i^{th} tree.

3 EXPERIMENTAL RESULTS

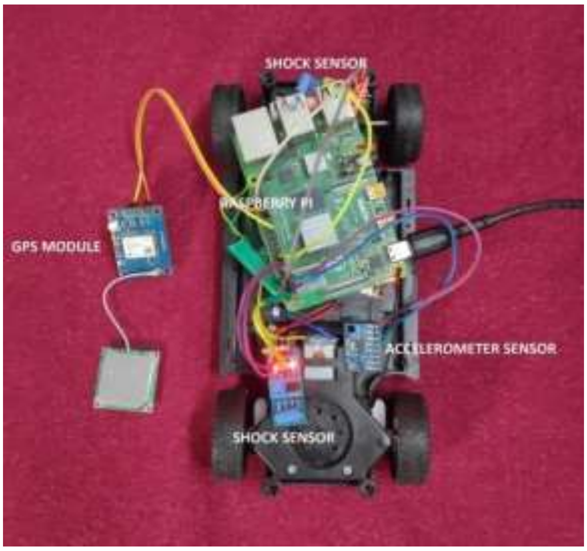


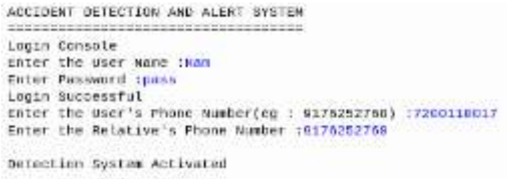
Figure 4. Circuit Connections

The **accelerometer sensor** transmits data via the I2C and SPI interfaces. The raspberry pi is interfaced by ADXL345 by connecting the accelerometer's SDA and SCL pins to the raspberry pi's GPIO2 and GPIO3 pins respectively as shown in the figure 4. Since the accelerometer sensor operates on 3.3 volts, the VCC pin is linked to the Raspberry Pi's physical Pin 1 and the ground pin is attached to the Raspberry Pi's Physical Pin 6. The **Vibration sensor SW-420** is a high sensitivity non-directional vibration sensor that operates on 5 volts, so the VCC pin is connected to the Raspberry Pi's physical Pin 2, the ground pin is connected to the Raspberry Pi's Physical Pin 9, and the Digital Output pin is connected to the Raspberry Pi's GPIO17. The Digital output pin of another vibration sensor is connected to the Raspberry Pi's GPIO27. These sensors are fitted to the front and rear positions of the vehicle. The **GPS module** is used to retrieve the current location and is powered by 5 volts, therefore the VCC pin is linked to the Raspberry Pi's physical pin 4, the ground pin is attached to the Raspberry Pi's physical pin 14, and the TX pin is connected to the Raspberry Pi's GPIO18.

3.1 IMPLEMENTATION

i. *USER DETAIL*

The first step is to use the command line interface to obtain user information such as name, phone number, and trusted person phone number, as shown in figure 5. These details are necessary to initiate the **Detection System**.



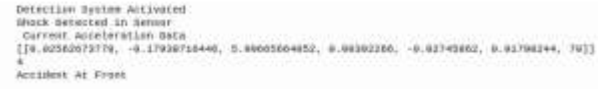
```
ACCIDENT DETECTION AND ALERT SYSTEM
===================================
Login Console
Enter the user Name :Ram
Enter Password :pass
Login Successful
Enter the User's Phone Number(eg : 9176252768) :7200110017
Enter the Relative's Phone Number :9176252768

Detection System Activated
```

Figure 5. User Details

ii. *ACCIDENT DETECTION*

The shock sensor value varies when the car is crashed. It starts the detection phase, in which the accelerometer value is dynamically obtained using the accelerometer sensor, as shown in figure 6, and it returns gyroscope and accelerometer values in the x,y, and z planes. These data are predicted using the Xgboost algorithm, which returns an outcome of Accident i.e.., along with which side of the vehicle the collision happened or Not.

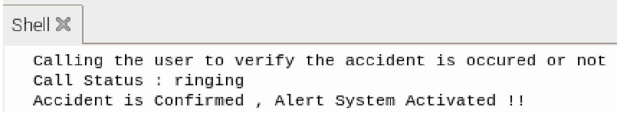


```
Detection System Activated
Shock Detected in Sensor
 Current Acceleration Data
[[0.02562673779, -0.17930718440, 5.80665664852, 0.08302286, -0.02745862, 0.01798244, 79]]
4
Accident At Front
```

Figure 6. Fetching Accelerometer Data Dynamically

iii. *ACCIDENT VERIFICATION*

For example, if a user unintentionally crashes his or her car, it is not deemed an accident. As a result, we must confirm whether the accident happened. We do this by making a voice call to that user; if the user answers the call, it is not a major issue. If the user does not answer the phone, a serious accident has occurred, which triggers the hospital detection phase as shown in figure 7.

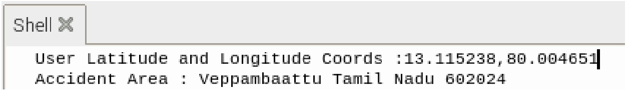


```
Calling the user to verify the accident is occured or not
Call Status : ringing
Accident is Confirmed , Alert System Activated !!
```

Figure 7. Confirmation of Accident

iv. *IDENTIFYING LOCATION OF ACCIDENT*

The GPS module is used to determine the latitude and longitude coordinates of the accident once it has been validated. We can get the pincode of that specific location using Selenium, as shown in figure 8 and figure 9.



```
User Latitude and Longitude Coords :13.115238,80.004651
Accident Area : Veppambaattu Tamil Nadu 602024
```

Figure 8. Fetching Location of Accident Spot

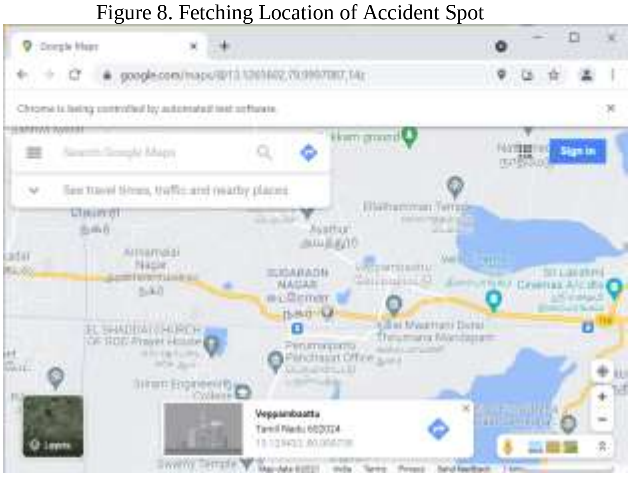


Figure 9. Map Image Pointing to Accident Location

v. *FINDING NEAREST HOSPITAL*

In the mysql database, there is a list of hospital data that has been predefined. As shown in figure 10, specific hospitals that have pincode are fetched from the database using the user's location pin-code. The nearest hospital to the accident location is then found from the list of hospitals using the Euclidean Distance formula as shown in figure 11.

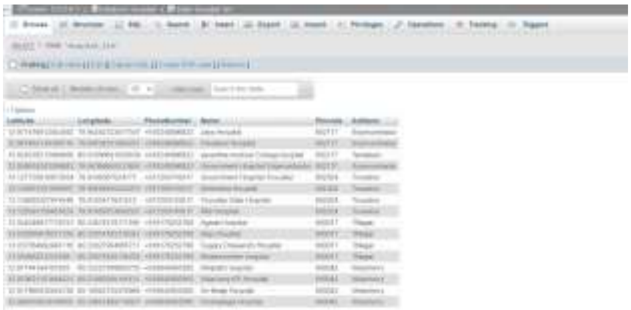
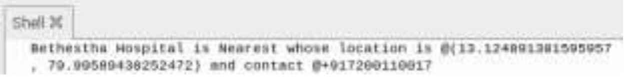


Figure 10. Hospital Database List

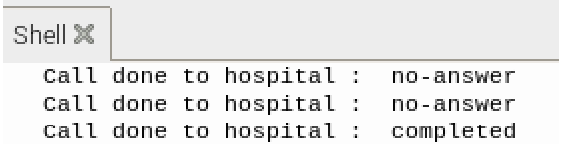


```
Bethestha Hospital is Nearest whose location is @(13.124891381595957
, 79.99589438252472) and contact @+917200110017
```

Figure 11. Selected Nearest Hospital

vi. *NOTIFY HOSPITAL*

A voice call is dispatched to the designated hospital using twilo's voice API, as shown in figure 12. The text format of the audio call contains information regarding the accident's location, such as lat long coordinates.

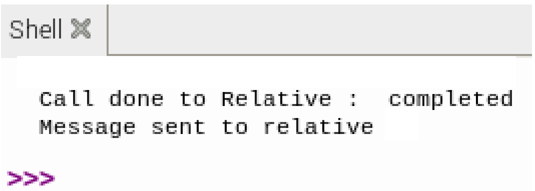


```
Call done to hospital :  no-answer
Call done to hospital :  no-answer
Call done to hospital :  completed
```

Figure 12. Passing Information to Hospital via Voice Call

vii. *NOTIFY TRUSTED PERSON*

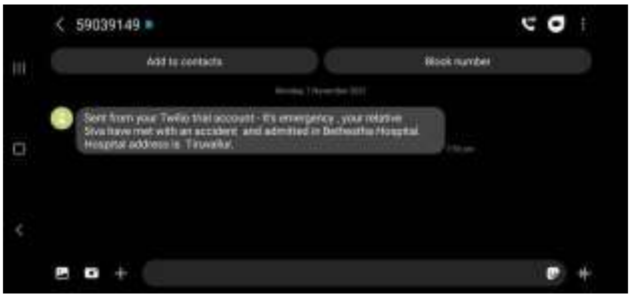
As demonstrated in figure 13, a message is sent to the trusted person using twilo's message API. As illustrated in figure 14, the message format contains information regarding his or her relative's accident as well as the hospital to which the person has been admitted.



```
Call done to Relative :  completed
Message sent to relative

>>>
```

Figure 13. Passing Information to Trusted Person via Voice Call

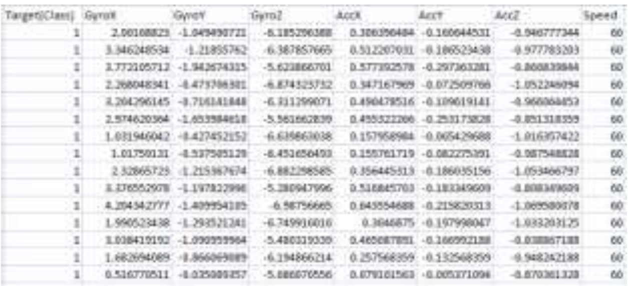


Sent from your Twilio trial account - It's emergency , your relative Siva have met with an accident and admitted in Bethestha Hospital. Hospital address is Tiruvallur.

Figure 14. Passing Information to Trusted Person via Text Message

3.2 RESULTS

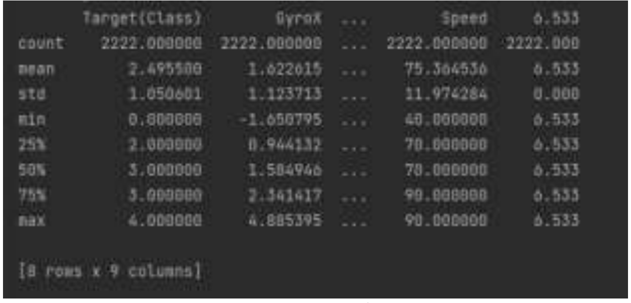
Dataset



Target(Class)	GyroX	GyroY	GyroZ	AccX	AccY	AccZ	Speed
1	2.00168823	-1.049490721	-6.185296388	0.306356484	-0.160644531	-0.946777344	60
1	3.346248534	-1.21855762	-6.387857665	0.512207031	-0.186523438	-0.977783203	60
1	3.772105712	-1.942674315	-5.623866701	0.577392578	-0.297363281	-0.860839844	60
1	2.268048341	-0.473786381	-6.874325732	0.347167969	-0.072509766	-1.052246094	60
1	3.204296145	-0.716141848	-6.311299071	0.490478516	-0.109619141	-0.966064453	60
1	2.574620364	-1.653984618	-5.561662839	0.455322266	-0.253173828	-0.851318359	60
1	1.031946042	-0.427452152	-6.639863038	0.157958984	-0.065429688	-1.016357422	60
1	1.01750131	-0.537505129	-6.451656493	0.155761719	-0.082275391	-0.987548828	60
1	2.32865725	-1.215367674	-6.882298585	0.356445313	-0.186035156	-1.053466797	60
1	3.376552978	-1.197822998	-5.280947996	0.516845703	-0.183349609	-0.808349609	60
1	4.204342777	-1.409954185	-6.58756665	0.643554688	-0.215820313	-1.069580078	60
1	1.990523438	-1.293521241	-6.749916016	0.3046875	-0.197998047	-1.033203125	60
1	3.038419192	-1.090559964	-5.480319339	0.465087891	-0.166992188	-0.838867188	60
1	1.682694089	-0.866069889	-6.194866214	0.257568359	-0.132568359	-0.948242188	60
1	0.516770511	-0.035089357	-5.886670556	0.079101563	-0.005371094	-0.870361328	60

Figure15. Dataset in CSV format

Summary of XGBoost Model



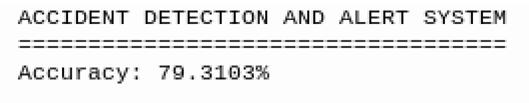
```
       Target(Class)        GyroX  ...         Speed     6.533
count    2222.000000  2222.000000  ...   2222.000000  2222.000
mean        2.495500     1.622615  ...     75.364536     6.533
std         1.050601     1.123713  ...     11.974284     0.000
min         0.800000    -1.650795  ...     40.000000     6.533
25%         2.000000     0.944132  ...     70.000000     6.533
50%         3.000000     1.584946  ...     70.000000     6.533
75%         3.000000     2.341417  ...     90.000000     6.533
max         4.000000     4.885395  ...     90.000000     6.533

[8 rows x 9 columns]
```

Figure 16. Summary of Model

Accuracy Score

accuracy = sklearn.metrics.accuracy_score(y_test, XGBoostmodel.predict(X_test))

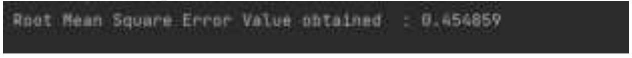


```
ACCIDENT DETECTION AND ALERT SYSTEM
===================================
Accuracy: 79.3103%
```

Figure 17. Accuracy Score determined by using XGBoost model

Loss function value:

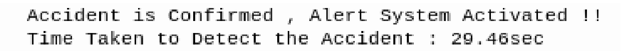
rmse = math.sqrt(sklearn.metrics.mean_squared_error(y_test,XGBo ostmodel.predict(X_test))



```
Root Mean Square Error Value obtained  : 0.454859
```

Figure 18. Loss Function Error Value

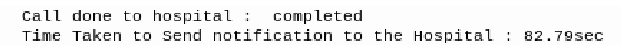
Time For Detection Phase



```
Accident is Confirmed , Alert System Activated !!
Time Taken to Detect the Accident : 29.46sec
```

Figure 19. Time Interval for Detection Phase

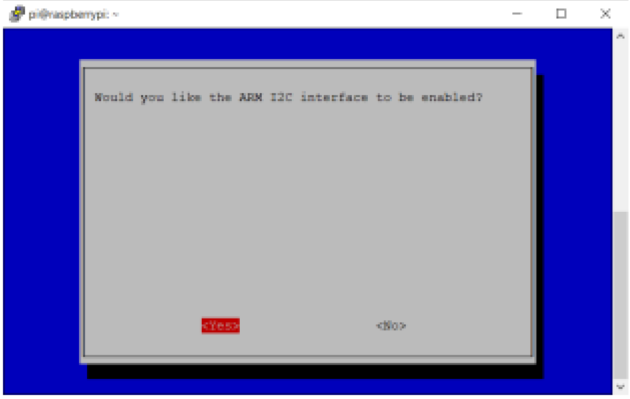
Time For Alert Phase



```
Call done to hospital :  completed
Time Taken to Send notification to the Hospital : 82.79sec
```

Figure 20. Time Interval for Alert Phase

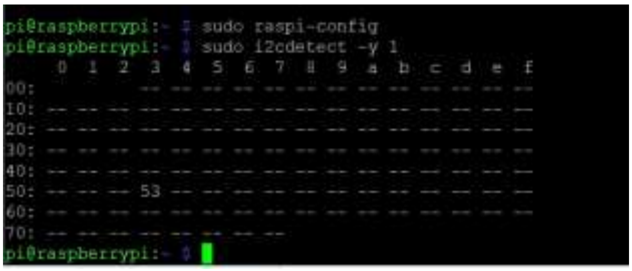
Invoking I2C Connection Establishment



```
Would you like the ARM I2C interface to be enabled?

          <Yes>                    <No>
```

Figure 21. InterIntegratedCircuit Connection Establishment



```
pi@raspberrypi:~ $ sudo raspi-config
pi@raspberrypi:~ $ sudo i2cdetect -y 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:          -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- 53 -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
pi@raspberrypi:~ $
```

Figure 22. Checking I2C Connection

4 CONCLUSION

In recent years Artificial Intelligence and Machine Learning have been growing rapidly. Many useful applications have emerged with AI and ML concepts. Automatic accident detection became a very important topic in traffic management systems. Detection of accidents will avoid future accidents and will help authorities to make road segments available for traffic again. The proposed system deals with the accident alert detection and alerting. Raspberry Pi is the heart of the system which helps in transferring the message to different devices in the system. Vibration sensor will be activated when the accident occurs and the accelerometer sensor detects the position of the accident using a machine learning algorithm then information is transferred to the hospital through the GSM module. The Hospital name will be alerted to the Trusted person..Let this effective algorithm make our travel journey smooth and hassle-less while we enjoy it.

REFERENCES

[1] Conway, A. M., Durbach, I. N., McInnes, A., & Harris, R. N. (2021). Frame-by-frame annotation of video recordings using deep neural networks. Ecosphere,12(3). https://doi.org/10.1002/ecs2.v12.310.1002/e cs2.3384

[2] Nikhil Kumar, Debopam Acharya, and Divya Lohani., 2021, An IoT-Based Vehicle Accident Detection and Classification System Using Sensor Fusion.

[3] Tomáš Zavodjančík,Gustáv Kasanický, Lýdia Demčáková. Reduction of Pedestrian Accidents–Automated Road Vehicles 2021.

[4] Soleimani, S., Mousa, S.R., Codjoe, J., Leitner, M., 2019. A comprehensive railroad highway grade crossing consolidation model: a machine learning approach. Accid. Anal. Prev.128 (April), 65–77. https://doi.org/10.1016/j.aap.2019.04.002.

[5] Amir Bahador Parsa, Ali Movahedi , Homa Taghipour , Sybil Derrible, Abolfazl (Kouros) 2019. Mohammadian. Toward safer highways, application of XGBoost and SHAP for real-time accident detection and feature analysis www.elsevier.com/locate/aap.

[6] Schlögl, M., Stütz, R., Laaha, G., Melcher, M., 2019. A comparison of statistical learning methods for deriving determining factors of accident occurrence from an imbalanced high resolution dataset. Accid. Anal. Prev. 127 (February), 134–149. https://doi.org/10.1016/j.aap.2019.02.008.

[7] Samira Ahangari, Mansoureh Jeihani, Abdollah Dehzangi., August 2019. A Machine Learning Distracted Driving Prediction Model.Article No.:77 Pages 1–6 https://doi.org/10.1145/3387168.3387198.

[8] WAN-JUNG CHANG , (Member, IEEE), LIANG-BI CHEN , (Senior Member, IEEE), AND KE-YU SU December 2019. DeepCrash: A Deep Learning-Based Internet of Vehicles System for Head-On and Single-Vehicle Accident Detection With Emergency Notification.

[9] Amir Bahador Parsa , Homa Taghipour , Sybil Derrible , Abolfazl (Kouros) Mohammadian, May 2019. Real-time accident detection: Coping with imbalanced data.

[10] T. Mekki, I. Jabri, A. Rachedi, and M. Ben Jemaa, “Vehicular cloud networking: Evolutionary game with reinforcement learning-based access approach,” Int. J. Bio Inspired Comput., vol. 13, no. 1, pp. 45–58, 2019, doi: 10.1504/IJBIC.2019.097730.

[11] Global Status Report on Road Safety 2018: Summary, WHO, Geneva,Switzerland,2018. https://www.who.int/violence_injury_prevention/road_safety_status/2018/English-Summary-GSRRS2018.pdf

[12] E. K. Priya et al., “IoT based vehicle tracking and accident detec-tion system,” Int. J. Innov. Res. Comput. Commun. Eng., vol. 5, no. 3,pp. 4020–4026, 2017.

[13] Alwan, Z.S., Muhammed, H., Alshaibani, A., 2016. Car accident detection and notification system using smartphones. Int. J. Comput. Sci. Mob. Comput.(January).

[14] Fernandes, B., Alam, M., Gomes, V., Ferreira, J., Oliveira, A., 2016. Automatic accident detection with multi-modal alert system implementation for ITS. Veh. Commun. 3, 1–11. https://doi.org/10.1016/j.vehcom.2015.11.001.

[15] H. A. Ibrahim, A. K. Aly, and B. H. Far, “A system forvehicle collision and rollover detection,” in Proc. Can. Conf. Elect. Comput. Eng., 2016, pp. 1–6, doi: 10.1109/CCECE.2016.7726720.

[16] Kwak, H., Kho, S., 2016. Predicting crash risk and identifying crash precursors on Korean expressways using loop detector data. Accid. Anal. Prev. 88, 9–19. https://doi.org/ 10.1016/j.aap.2015.12.004.

[17] J. Smolka and M. Skublewska-Paszkowska, "A method for collision detection using mobile devices," in Proc. 9th Int. Conf. Human Syst. Interact. (HSI), 2016, pp. 126–132, doi: 10.1109/HSI.2016.7529620.

[18] H. M. Ali and Z. S. Alwan, ''Car accident detection and notification system using smartphone,'' Int. J. Comput. Sci. Mobile Comput., vol. 4, no. 4, pp. 620–635, Apr. 2015.

[19] F. Aloul, I. Zualkernan, R. Abu-Salma, H. Al-Ali, and M. Al-Merri, "IBump: Smartphone application to detect car accidents," Comput. Elect. Eng., vol. 43, pp. 66–75, Apr. 2015, doi: 10.1016/j.compeleceng.2015.03.003.

[20] K. K. Patel, S. M. Patel, P. G. Scholar and C. Salazar, "Internet of Things-IOT: Definition characteristics architecture enabling technologies application & future challenges", *Int. J. Eng. Sci. Comput.*, vol. 6, no. 5, pp. 6122-6131, 2014.

[21] Amin, S., Jalil, J., 2012. Accident detection and reporting system using GPS, GPRS and GSM technology. Int. Conf. Informatics, Electron. Vis 640–643.
https://doi.org/10.1109/ICIEV.2012.6317382

[22] M. Viswanath and K.M.Arunraja, "A Literature Review on Hybrid Electric Vehicles", International Journal of Engineering Research & Technology, Vol.6 Issue 04,Special Issue on 2018