Driver Accident Prevention System

Submitted in partial fulfillment of the requirements for the degree of

Bachelor of Technology

in

Electronics and Communication Engineering

by

KATHAKOLI SENGUPTA
18BEC0053

ABHYUDAY TRIPATHI
18BEC0085

NANDINI AWTANI
18BEC0538

Under the guidance of

Prof. Vaegae Naveen Kumar

School of Electronics engineering

VIT,

Vellore.

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

April, 2022

DECLARATION

We hereby declare that the thesis entitled **“Development of an Integrated Driver Accident Prevention System**” submitted by us, for the award of the degree of *Bachelor of Technology in Electronics and Communication Engineering* to VIT is a record of bonafide work carried out by me under the supervision of Prof. Vaegae Naveen Kumar.

We further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place : Vellore

Date :

Signature of the Candidate

CERTIFICATE

This is to certify that the thesis entitled "**Development of an Integrated Driver Accident Prevention System**" submitted by **Kathakoli Sengupta (18BEC0053), Abhyuday Tripathi (18BEC0085) and Nandini Awtani (18BEC0538)**, VIT University, for the award of the degree of *Bachelor of Technology in Electronics and Communication Engineering*, is a record of bonafide work carried out by them under my supervision during the period, 01. 12. 2021 to 30.04.2022, as per the VIT code of academic and research ethics.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The thesis fulfills the requirements and regulations of the University and in my opinion meets the necessary standards for submission.

Place : Vellore

Date :

Signature of the Guide

Internal Examiner

External Examiner

Dr. Thanikaiselvan V.

Electronics and Communication Engineering

ACKNOWLEDGEMENTS

We would like to express our gratitude to the chancellor of Vellore Institute of Technology, Dr. G Viswanathan, for giving me the opportunity to pursue my studies in this prestigious University.

We are very much grateful to **Dr. Sivanantham S**, Professor and Dean, School of Electronics Engineering for this level of encouragement for all endeavors upon the project.

We are taking this opportunity to extend our feeling of gratitude and sincere thanks to **Dr. Prakasam P**, Head, Department of Communication Engineering, School of Electronics Engineering for his concern, blessings, ever helping attitude and incessant encouragement.

We express our profound sense of gratitude and indebtedness to our revered guide Prof. Vaegae Naveen Kumar, Associate Professor, Department of Electronics and communication, Vellore Institute of Technology, Vellore for his valuable and gifted guidance, keen interest, constant encouragement, and unfailing source of inspiration, critical supervision and painstaking effort during the entire course of our study.

Our words are inadequate to acknowledge our parents whose attention, patience, sacrifice and courage provided us strength to move forward towards the completion of this study.

Kathakoli Sengupta, Abhyuday Tripathi and Nandini Awtani

Executive Summary

Recent reports show that there have been around 12256 road accidents due to drunk driving in India in the year 2019. Given the surge of road vehicles in every city in the current scenario and its necessity in people's daily lives it has become inevitable to come up with a solution to accident problems. Various research is performed around this domain but none of them has yet given any integrated solution to each system with accuracy high enough to be used for industry level manufacturing. It addresses three major causes of accidents i.e., collision with another vehicle, fire outbreak due to overheating and driver's lack of concentration due to drunk or drowsy state and proposes prediction system designs prior to any of these incidents. The project is implemented using sensors, microcontroller and camera for real-time data collection and CNN and ML algorithms for its classification and decision making. A comparison of accuracy of the algorithms tried for each system were made and the best in each case showed a detection accuracy of 85%. Though the system is not fast and sensitive enough to be on the road at this moment, usage of GPU instead of computer CPU and industry level highly sensitive sensors will successfully solve this issue. It is cost-effective since it uses very less software, robust, accurate enough and hence an appropriate solution for real-time road safety and risk management.

CONTENTS

List of Figures

List of Tables

List of Abbreviations

NCRB	National Crime Record Bureau
LPG	Liquid Petroleum Gas
CO	Carbon Monoxide
EN	Electronic Nose
CNN	Convolutional Neural Network
ML	Machine Learning
IDE	Integrated Development Environment
MHz	Mega Hertz
PWM	Pulse Width Modulation
USB	Universal Serial Bus
ICSP	In Circuit Serial Programming
SRAM	Static Random Access Memory
EEPROM	Electronically Erasable Programmable Read-Only Memory
MOS	Metal Oxide Semiconductor
VGG	Visual Geometry Group
LSTM	Long Short-Term Memory
ReLU	Rectified Linear Unit
RGB	Red Green Blue
FC	Fully Connected
MLP	Multilayer Perceptron
SVM/C	Support Vector Machine/Classifier
IG	Information Gain

1. INTRODUCTION

In today's date, transport by road is the most common means of transport. People either travel on their own vehicles or use the public transport system, adding up to a massive figure of 23000 billion passenger kilometers. And the number just keeps booming. All other means of transportations combined together are still just a fraction of this number.

But with the increase in vehicles on the road, the number of vehicles has also increased several times. There are approximately 1.3 million road deaths every year, the per day average coming to an average of 3600. The major cause of road accidents is driver error which is mostly caused by speeding, drunken driving and vehicle collisions. Multiple traffic regulation and driver monitoring systems have been introduced over the years but the deaths still continue and the number is certainly going up. Some main reasons being the lack of development and unawareness in some remote areas of the world. Even in our country, the traffic system is awry in the rural areas. Situations are no better in urban areas because despite having a comparatively better traffic system, the seer concentration of vehicles is too much to handle and hence, the accidents keep happening. Moreover, the safety measures that vehicle manufactures are introducing aren't enough either.

There is a need to introduce affordable, accurate systems that can help a driver and the authority to act before the danger arrives. It is necessary to ensure that the loss of lives stops before it's too late.

1.1. OBJECTIVE

The main objective of this project is to develop a system, a collection of safety measures developed with the help of technology which can help reduce the number of road accidents that occur on such a huge scale, caused by several factors like drunk driving, over-speeding and sleepy driving.

The objective has been further divided into the following sub-objectives to make the goal achievable:

- To research on the already adapted measures in this field.
- To prepare a blueprint of the system.

- To advance the features and algorithms for the system.
- To the hardware model by combining and interfacing all the hardware components.
- To create an electronic nose that can detect alcohol and hazardous gases.
- To develop the drunk face and drowsiness detection systems.
- To advance on the vehicle collision detection system.
- To analyze the finalized model of the accident prevention system.

1.2. MOTIVATION

The National Crime Record Bureau (NCRB) showed that there were almost 3.55 lakh road accidents in India in 2020 and about 1.33 lakh were injured. More than 60% of these road accidents occurred because of over-speeding, making it the obvious leading reason. It was followed by dangerous and careless driving. About 3% of road accidents are caused by drunk driving. Many other unreported cases happen too where the driver is drunk or sleepy. Several of the overspeeding and careless driving cases also have drinking as a component.

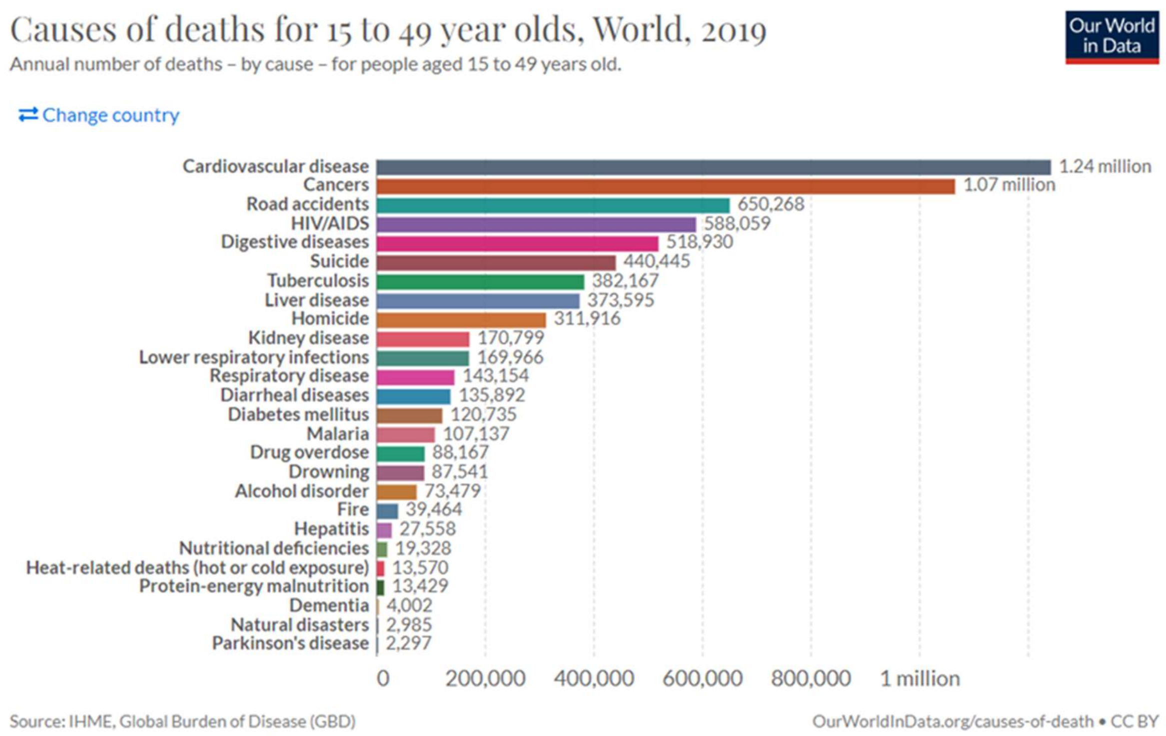


Fig 1.1: Causes of Death Data

Road accidents are also the third most common reason for deaths in the age group of 15-49 years in the entire world following cardiovascular diseases and cancers. This proves that road accidents are more common and fatal than a usual human being thinks. There is an urgent necessity to take preventive measures and spread awareness to the common man.

Not to be mistaken, governments and authorities have made several attempts to reduce the numbers but it hasn't made that huge of an impact because there is still room for mishaps. Something has to be done so that the room for error is almost 0. Each and every human life values the same and its safety should be ensured. In order to assist in resolving the issue and bring down the deadly numbers, a system is introduced which will have drunken face and drowsy face detection systems, a vehicle collision prevention system and an alcohol and hazardous gases detection system.

1.3. BACKGROUND

The most undesirable thing that happens to a road user is an accident, even though they happen quite frequently. There are approximately 1.3 million road deaths every year. The major cause of road accidents is driver error which is mostly caused by speeding, drunken driving and vehicle collisions Most of the fatal accidents are caused by over-speeding [1]. Faster vehicles are more prone to accidents than slower ones and the severity of accidents will also be more in case of faster vehicles. Higher the speed, the greater the risk. [2] Another issue that leads to multiple road accidents is drunk driving. Alcohol reduces concentration. It decreases the reaction time of the human body. Limbs take more to react to the instructions of the brain. It hampers vision due to dizziness. Alcohol dampens fear and incites humans to take risks. All these factors while driving cause accidents and many times it proves fatal. [3][4]. These road accidents are a major topic of concern for most governments because it causes a major loss of life and property.

The problem of road accidents is growing rapidly, and several techniques have been proposed to solve it, including the concept of the Electric Nose. This can help reduce the number of accidents in many ways by detecting whether or not a person is intoxicated. It is based on the concept that the device can detect various gasses such as LPG, Propane, Hydrogen, Methane and other combustible gasses using MQ2 using which we can develop an engine locking system that will stop the driver to drive further.[5][6]

Electric Nose can also in detecting the air quality of the cabin/car which can help us to know if there is any spark generating gasses leaked in the car so that driver can be informed at the correct time and can be saved from unwanted injuries this can be done by developing a real-time cloud-based air quality system that enables the prediction of current and future air quality of the car.[7]

The efficiency of the electric Nose depends on its capacity to differentiate different gasses so to make the EN most efficient more sensors should be used so that it can become more sensitive to different types of gasses and after the detection part, a pattern recognition algorithm should be designed to train the data in a right manner.[8] To improve the detection efficiency of EN CNN can be used because not only takes less time to sample but also had a great classification accuracy, it also uses fewer sampling points for the classification and also generates features automatically without the need for signal pre-processing.[9]

Apart from Electronic Nose data gathered from analyzing an image of the driver using 3 layered neural networks, by analyzing the speech of the driver and classifying as drunk in case of distorted speech, by biological sensor system fixed to the driver seat, by driving pattern can also be used in determining whether the driver is drunk or not and so further the engine can be locked.[10]

Vehicle Collision due to over-speeding also has mohair hand in road accidents detecting the vehicle with high speed and alerting them can save not only the drivers life but also other road users lives so this can be done with the help of video surveillance and then followed by Deep learning Techniques which can estimate the speed and trajectories of the object of interest with the objective of predicting and controlling the occurrence of traffic accidents in the area.[11][12]

Drawbacks

1.None of the researches mentioned above could build a model efficient enough to predict hazardous as well as alcoholic gasses at the same time.

2.Automation is the future so we are developing a system that would automatically detect smoke inside car in autopilot mode, drunk or drowsy driver in driver mode and will have an automatic vehicle collision prevention system to facilitate its autopilot mode along with an engine locking system in case of accidents.

3.All our systems are efficient enough to be implemented in the real-time scenario since it collects all the data while the car is in operation.

1.4. ORGANIZATION OF THE REPORT

The report has been organized as follows:

Chapter 2 discusses the goals of the project and provides the project description. Here, we have discussed the features available in the proposed design and the workflow of the model.

Chapter 3 gives the technical specifications of the hardware used in the project and the software used to develop the system. The hardware used are Arduino UNO, MQ-2 Sensors, MQ-3 Sensor, Servo Motor. It discussed the Software used, Programming Languages and platforms that have been used to code the Arduino UNO.

Chapter 4 gives the details about the approach that has been used to develop the prototype. It provides information about the algorithms developed for the functioning of the system, along with this it discusses the use of hardware and software in the project. It gives a detailed explanation of the working of the hardware circuit along with the software sections. Finally, it discusses the codes and standards, constraints, alternatives and tradeoffs pertaining to this project.

Chapter 5 gives the schedule, tasks and milestones of our work.

Chapter 6 gives the demonstration of the project in a written form. It discusses system activation and operation.

Chapter 7 presents the cost analysis of the project. Along with this, it discusses the results obtained in detail.

Chapter 8 gives an overall summary of the project. It concludes our work and gives the future scope of this prototype. It provides key points to get an understanding of the complete project.

Chapter 9 gives the references used in this project.

2. PROJECT DESCRIPTION AND GOALS

2.1. GOAL OF THE PROJECT

The goal of this project is to develop one system that is an ultimate combination of all the necessary components that are required to ensure the safety of a driver. The mindset is to design the system to be simple and affordable. It is not only meant to improve the driving experience and ensure the safety of the driver; it is also expected to not interfere with the other systems and smooth driving experience. It has to be accessible all around the world and every vehicle manufacturer should introduce a similar, if not the same system, in all their products.

It is desired to develop a system which drastically decreases the numbers of accidents that occur on a daily basis. It will ultimately contribute to the betterment of the society and improving human life.

2.2. PROJECT DESCRIPTION

The proposed accident detection system is an amalgamation of prevention systems designed for four different kinds of factors that can potentially cause a car accident. The system comes with a hazardous gas detection technology that can distinguish the concentration of inflammable gas to protect the car from fire breakouts along with an Alcohol detection system that can recognize drunk drivers and can lock the engine if anyone is detected since recent reports identify it as one of the main causes of car accidents. Drivers feeling sleepy and drowsy can also be a major reason for accidents so we have our second system which cures this problem by detecting when the driver is sleepy. Despite all this precautionary system to rule out the chances of another car hitting yours, we incorporate a Vehicle Collision Prevention System that automatically slows down or stops in case it detects a collision.

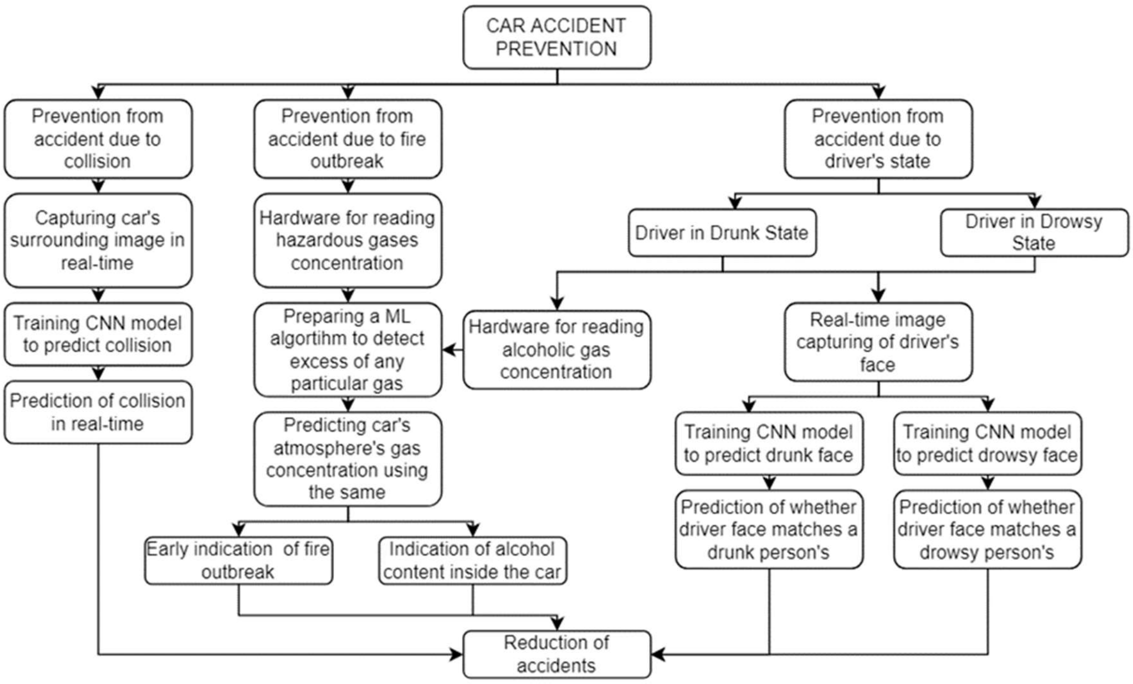


Fig 2.1: Block Diagram for the Proposed System

The above flow chart provides a detailed working of the entire system. The first system i.e., the system predicting chances of collision is completely dependent on the surroundings of the car. The system continuously takes input of the activities of the cars surrounding ours and predicts the chances of accident at every instance using a CNN model designed in the project. Hence, the driver is alerted by “COLLISION” in the display window in such dangerous cases, otherwise it shows “SAFE” for driving. This certainly helps in driver’s better understanding of traffic conditions and reduces chances of accidents.

The next cause of concern, outbreak of fires, can be easily detected by testing the presence of smoke or CO, since these are released in case of sparks. For this purpose, a hardware system is set up with MQ2 and MQ3 sensors for detecting a large range of gases. A ML model is trained to check increase in any particular gas among smoke, CO etc. In case an abnormal rise of anyone is found above the threshold value a fire possibility is predicted. This early indication of fire before any mishap definitely will be very useful in future to reduce accidents.

Despite the presence of all these detection mechanisms the main reason for accidents still should be driver’s carelessness and we cannot neglect chances of the driver being drunk or in a drowsy state. For addressing drunk drivers, the first preliminary test would be checking the presence of any alcoholic gasses inside the car. This was achieved by introducing a similar hardware as used in smoke detection, but this time an excessive rise of alcoholic gasses above threshold was concluded as presence of a drunk person inside the car.

To ensure that the drunk person is the driver and not one of the other people, the project also has a system for face analysis of the driver. The face analysis is carried on with the help of a CNN model trained with a drunk face dataset and is capable of predicting whether a person is drunk or not from their real-time image.

Further, the last but not the least concern would be drivers falling asleep while driving. A similar CNN based system is introduced inside the car but this time trained with open and closed eyes dataset and eyes closed for a long period of time was concluded as a sleepy state. These systems combined deal with all possible accident-causing factors in a car and hence these are integrated into this project to design a full accident proof system.

3. TECHNICAL SPECIFICATIONS

There are various hardware and software that have been used for the development and execution of the Driver Accident and Prevention System. The hardware includes Arduino UNO, MQ-2 Sensor, MQ-3 Sensor, Servo Motor, video camera. The software aspects are Arduino IDE, Android Application and the software platforms used to develop them. The details and specifications have been discussed below.

3.1. HARDWARE

3.1.1 Arduino UNO

Arduino/Genuino Uno is a microcontroller board based on the ATmega328P.

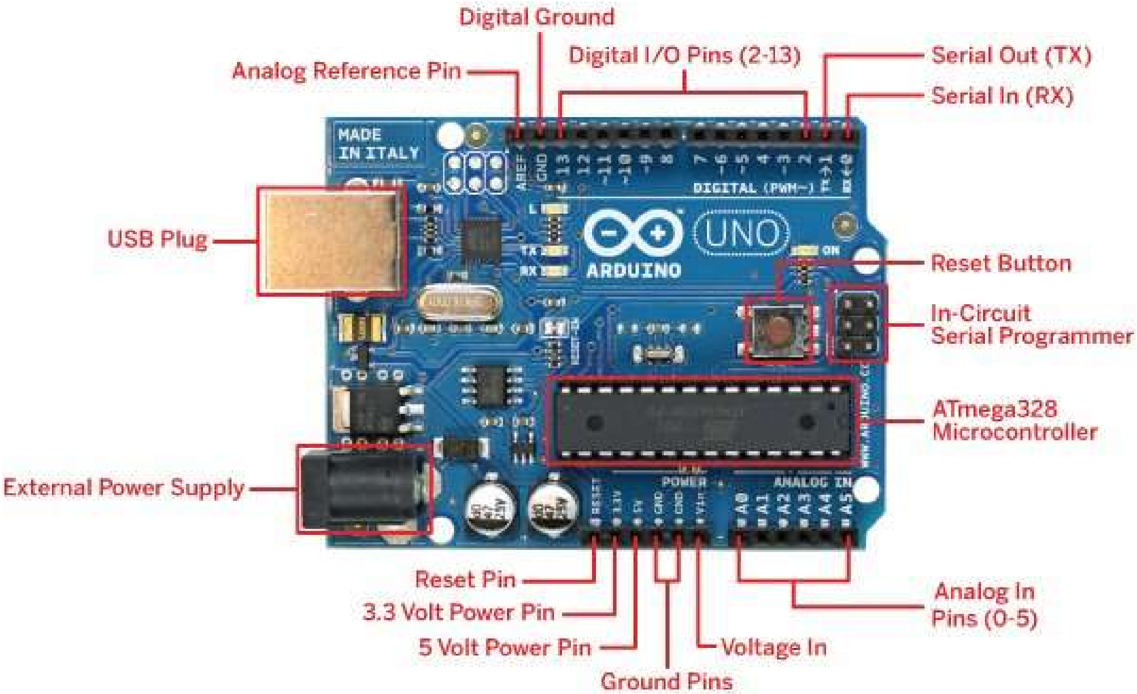


Fig 3.1. Arduino UNO

It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started. We can tinker with your UNO without worrying too much about doing something wrong, worst-case scenario you can replace the chip for a few dollars and start over again.

"Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform; for an extensive list of current, past or outdated boards see the Arduino index of boards.

Technical Specifications

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V

Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P)
	of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

3.1.2 MQ-2 Sensor

This is a robust Gas sensor suitable for sensing LPG, Smoke, Alcohol, Propane, Hydrogen, Methane and Carbon Monoxide concentrations in the air. It is a Metal Oxide Semiconductor (MOS) type Gas Sensor also known as Chemiresistors as the detection is based upon change of resistance of the sensing material when the Gas comes in contact with the material. Using a simple voltage divider network, concentrations of gas can be detected.

The sensor is actually enclosed in two layers of fine stainless-steel mesh called Anti-explosion network. It ensures that the heater element inside the sensor will not cause an explosion, as we are sensing flammable gases.

Fig 3.2. MQ2 sensor

The tubular sensing element is made up of Aluminum Oxide (AL_2O_3) based ceramic and has a coating of Tin Dioxide (SnO_2). Tin Dioxide is the most important material being sensitive towards combustible gases. However, the ceramic substrate merely increases heating efficiency and ensures the sensor area is heated to a working temperature constantly.

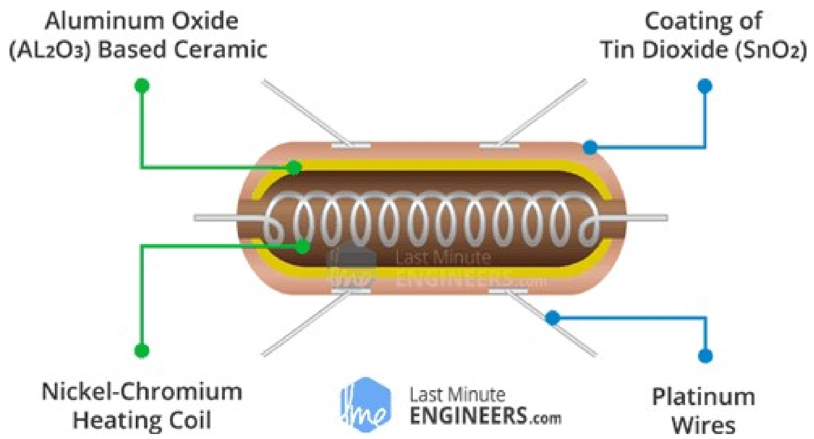


Fig 3.3. Inner Configuration of MQ2

So, the Nickel-Chromium coil and Aluminum Oxide based ceramic forms a Heating System; while Platinum wires and coating of Tin Dioxide forms a Sensing System.

Technical Specifications

Operating voltage	5V
Load resistance	20 KΩ

Heater resistance	33Ω ± 5%
Heating consumption	<800mw
Sensing Resistance	10 KΩ – 60 KΩ
Concentration Scope	200 – 10000ppm
Preheat Time	Over 24 hours

3.1.3 MQ-3 Sensor

It is a low-cost semiconductor sensor which can detect the presence of alcohol gases at concentrations from 0.05 mg/L to 10 mg/L. The sensitive material used for this sensor is SnO_2, whose conductivity is lower in clean air.

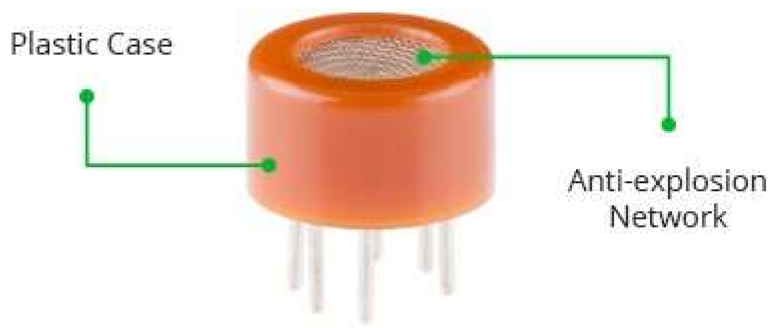


Fig 3.4. MQ3 sensor

When the SnO_2 semiconductor layer is heated at high temperature, oxygen is adsorbed on the surface. In clean air, electrons from the conduction band in tin dioxide are attracted to oxygen molecules. This forms an electron depletion layer just below the surface of SnO_2 particles and forms a potential barrier. As a result, the SnO_2 film becomes highly resistive and prevents electric current flow.

In the presence of alcohol, however, the surface density of adsorbed oxygen decreases as it reacts with the alcohols; which lowers the potential barrier. Electrons are then released into the tin dioxide, allowing current to flow freely through the sensor.

Technical Specifications

Operating voltage	5V
Load resistance	200 KΩ
Heater resistance	33Ω ± 5%
Heating consumption	<800mw
Sensing Resistance	1 MΩ – 8 MΩ
Concentration Scope	25 – 500 ppm
Preheat Time	Over 24 hours

3.2. SOFTWARE

3.2.1. Jupyter Notebook

JupyterLab is the latest web-based interactive development environment for notebooks, code, and data. We will use it to code our neural networks model.

3.2.2. Carla

CARLA has been developed from the ground up to support development, training, and validation of autonomous driving systems. In addition to open-source code and protocols, CARLA provides open digital assets (urban layouts, buildings, vehicles) that were created for this purpose and can be used freely. We will use it to create samples of vehicle collisions.

3.3. LIBRARIES

3.3.1 Keras

Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library.

3.3.2. Tensorflow

TensorFlow is a free and open-source software library for machine learning and artificial intelligence. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks.

3.3.3. OpenCV

OpenCV is a library of programming functions mainly aimed at real-time computer vision.

3.3.4. Pandas

pandas are a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series.

3.3.5. Numpy

NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

3.3.6. Matplotlib

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy.

3.3.7. Sklearn

Scikit-learn (formerly scikits.learn and also known as sklearn) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support-vector machines, random forests, gradient boosting, *k*-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

3.4. PROGRAMING LANGUAGES

3.4.1 Python

We will use it to code the neural network model.

3.3.2. C++

We will use it for writing the program for arduino.

4. DESIGN APPROACH AND DETAILS

4.1. DESIGN APPROACH

The proposed accident detection system is an amalgamation of prevention systems designed for four different kinds of factors that can potentially cause a car accident. The system comes with a hazardous gas detection technology that can distinguish the concentration of inflammable gas to protect the car from fire breakouts along with an Alcohol detection system that can recognize drunk drivers and can lock the engine if anyone is detected since recent reports identify it as one of the main causes of car accidents. Drivers feeling sleepy and drowsy can also be a major reason for accidents so we have our second system which cures this problem also by detecting when the driver is sleepy. Despite all this precautionary system to rule out the chances of another car hitting yours, we incorporate a Vehicle Collision Prevention System that automatically slows down or stops in case it detects a collision.

4.1.1. Vehicle Collision Detection System

The purpose of this system is to alert the driver in a dangerous accident situation. It captures real- time video of the car's surroundings and predicts whether there is a possibility of an accident.

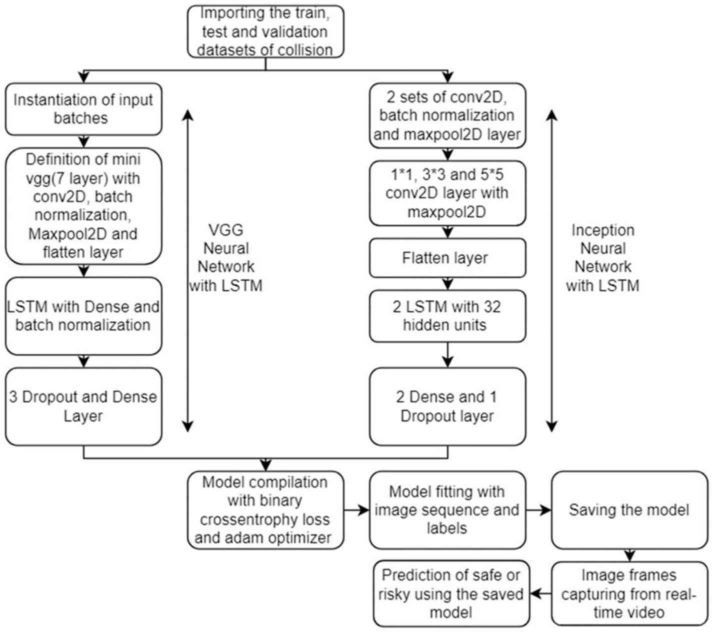


Fig 4.1. Flowchart of Vehicle Collision

The above flowchart depicts the process flow of this particular system.

Collision Dataset : Carla is an open automobile simulation software with the autopilot and self-driving modes. A total of 1392 instances were chosen in collision and safety situations. At every instance, 8 images were taken. Due to big data size, the images are converted into NumPy pixel arrays and stored in a pixel file with proper labels. These images are divided into train, test and validation dataset and used. The test video is a general recorded car movement video.

VGG and inception networks are used for the analysis along with long-short term memory for the model definition. The defined model is trained and tested with a pre-recorded collision dataset. Real-time video of the drive is captured and tested with the model to predict whether collision is possible at every instance. A detailed overview of the models defined and used is given below,

4.1.1.1. VGG with LSTM :

VGG(Very Deep Convolutional) network : VGG is a very efficient and widely used object recognition network. It typically uses 3 ReLU activation functions for accurate classification and works with many fewer parameters and more weight layers improving performance of the network.

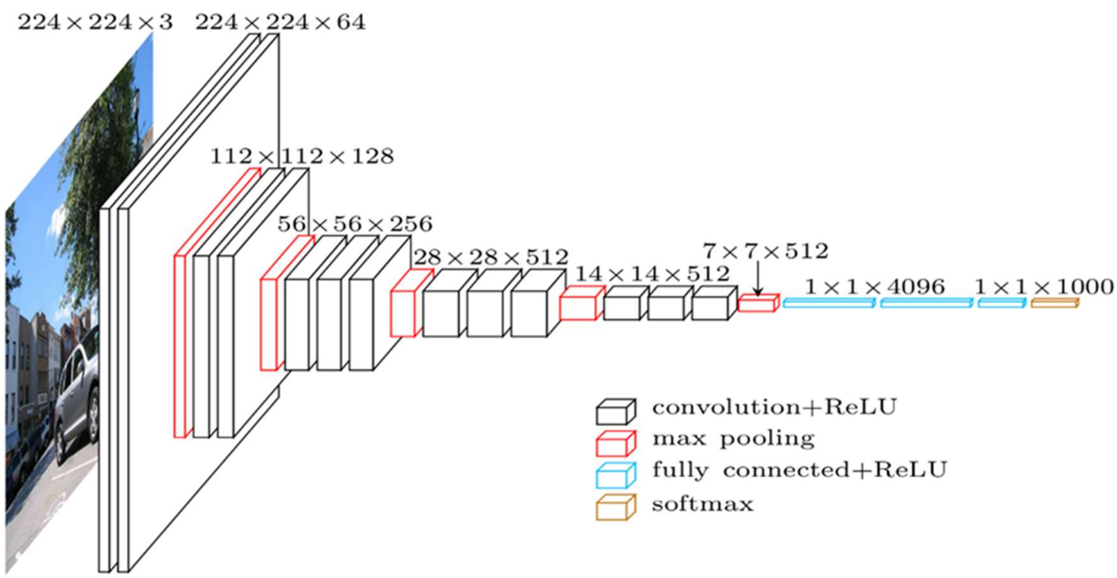


Fig 4.2. VGG Network

A typical layer consists of

Input : The network takes an input of 224*224 pixels from the center of the RGB image captured and passed to the subsequent layers.

Convolutional Layer : In this layer, a filter/kernel matrix is assigned and matrix multiplication is carried on over the entire RGB image pixel matrix in order to extract the high- and low-level features from the same. ReLU or Rectified Linear Activation function is a piecewise activation function giving input as output for all positive values and 0 for all negative ones.

Mathematically,

$$f(x) = \max(0, x)$$

Graphically,

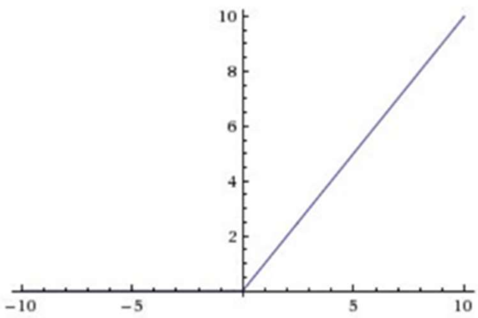


Fig 4.3. RELU Activation

The VGG uses a 3*3 matrix which is the smallest possible for complete extraction with a stride or shift of only 1. There are convolution layers with 64, 128, 256 and 512 filters respectively for detecting lower-level features like edges to higher level features.

Batch-Normalization : This layer restricts the input values within a standard value and hence reduces the number of epochs required for proper training.

Max pool Layer : This layer returns the maximum pixel value from the portion of the image under a given kernel. It extracts dominant features, reduces the computational complexity by reducing the number of units to be processed and also acts as noise suppressant.

FC Layer : VGG has 3 fully connected layers, with 2 containing 4096 channels and the last contains 1000 channels.

Flatten layer : This is typically used for converting multidimensional pixel matrices into one directional vector for further processing.

LSTM Network : The LSTM is used when a sequence data is captured over a period of time and in each stage some data from the previous stage needs to be stored and used.

A typical LSTM consists of

Inputs : Some inputs are the data that are saved from the prediction of previous stages and new inputs are added to it.

Forget Gate : This stage is characterized by the sigmoid activation function that helps to keep the values that are needed by multiplying by 1 and discard the rest. Sigmoid Activation Function : The sigmoid or logistic activation functions maps the entire data between 0 and 1 depending on the closeness of the value to these.

Mathematically,

$$S(x) = \frac{1}{1 + e^{-x}}$$

Graphically,

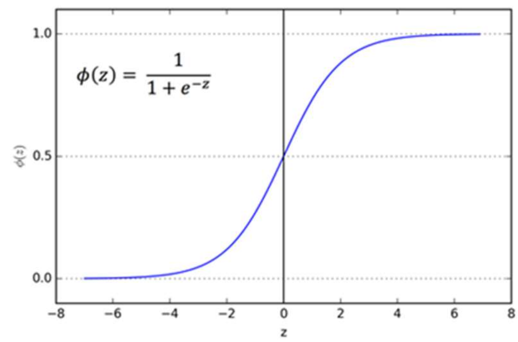


Fig 4.4. Sigmoid Activation

Input Gate: This decides which features are to be stored for the next stage of processing and adds the same to the cell state.

Update cell state: The previous cell state is then updated with the new cell state formed by the input gate.

Update hidden state: The cell state is therefore passed through the activation function and multiplied by output values to form the hidden stage for next layer processing. Tanh activation function: This function maps the output values between -1 and 1 depending on their closeness to the same.

Mathematically,

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

Graphically,

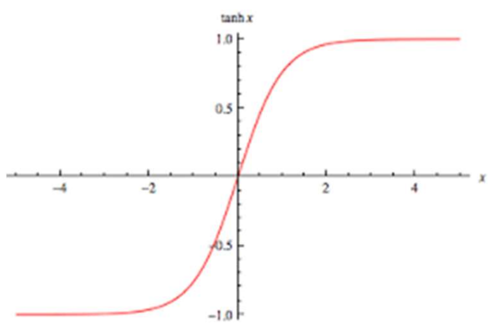


Fig 4.5. Tanh Activation

The LSTM used in this paper uses 32 hidden units.

Dense Layer : The dense layer with 1024,512, 64 and 2 units are defined followed my SoftMax activation function to finally converge to a single output value. SoftMax Activation function : Softmax activation function in the output layer helps to predict multinomial probabilities and hence is used when multiclass classification is required.

Mathematically,

$$S(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}}$$

Graphically,

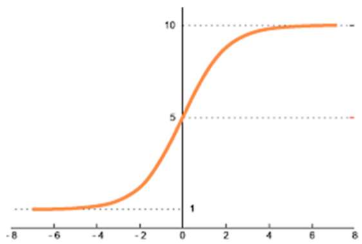


Fig 4.6. Softmax Activation

Dropout layer : This eliminates a part of the pixel values to reduce its number and facilitate easy and fast processing.

4.1.1.2. Inception with LSTM :

Inception Network : Inception network is considered to be one of the strongest networks capable of reducing computational loads due to reduction of input data at every stage and high-performance output.

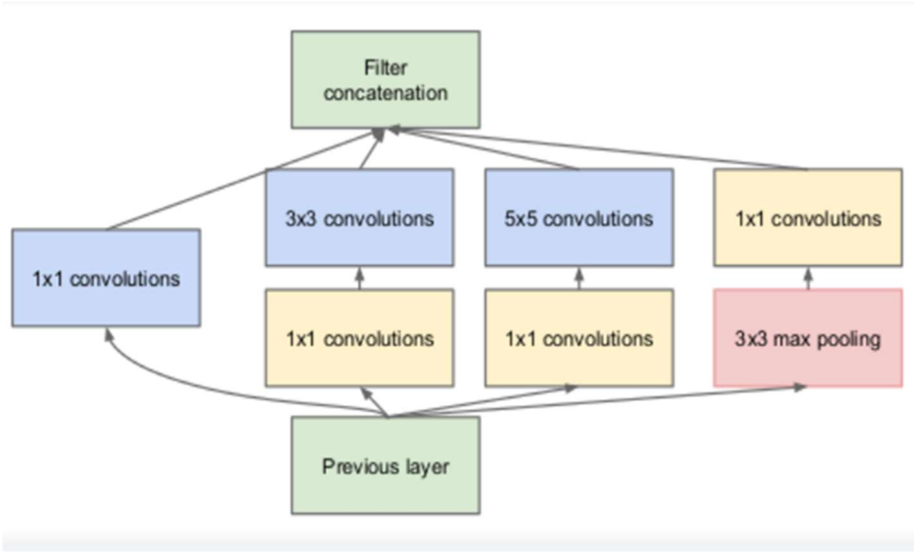


Fig 4.7. LSTM Network

A typical inception layer consists of

Convolutional Layer : The algorithm consists of two convolutional networks with 64 and 128 filters, followed by batch normalization and maxpooling. The output of this layer is then given to the inception module.

Inception Module : The inception module has convolutional networks with 1*1, 3*3 and 5*5 kernel size. The 1*1 kernel learns the pattern across depth of the input, 3*3 and 5*5 learns spatial pattern across all dimensions. The convolutional layers are followed my maxpooling and concatenation. This increase computational efficiency manifold by reducing number of multiplications involved.

The inception layer is followed by LSTM, dense and dropout layers to converge to an output from the same.

4.1.2. Hazardous and alcoholic gas detection :

The Hazardous and Alcoholic gas detection system aims at preventing accidents by locking the car engine as soon as the system detects presence of any hazardous gas or if it detects that the driver is drunk.

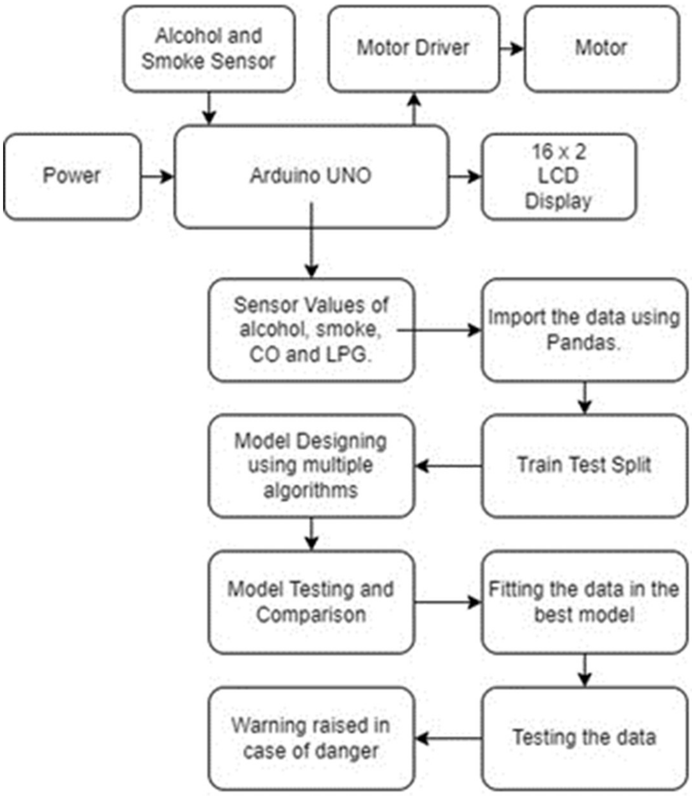


Fig 4.8. Flowchart of Hazardous Gas and Alcohol Detection

The above flowchart defines the working of the system and how the values detected by sensors are used by Arduino to lock the engine and the same values are further used for testing, training and preparing the model so that warning signs can be raised.

Alcohol and hazardous gasses dataset: The dataset was collected and recorded using our very own hardware model. We used the MQ2 and MQ3 sensors to collect data. The dataset contains 5 columns, 4 regarding values of LPG, CO, Smoke and Alcohol. The fifth and the last column has a number in the range 0 to 3, which is used to classify each data record as No Danger, Drunk, LPG and Smoke. There is no data for the case of high CO concentration and we planned not to experiment with such a toxic gas in home environment.

4.1.2.1. Detection of Hazardous Gasses and Alcohol:

Implementing the same on hardware to make this project to be used as a real time working device, we need to have essentials like all the components of some software and we should also have the basic knowledge of programming so that we can make our system work as we like.

Components Required

1. Arduino UNO

Arduino UNO is a low-fee, flexible, and easy-to-use programmable open-source microcontroller board that may be included into a selection of electronic initiatives.

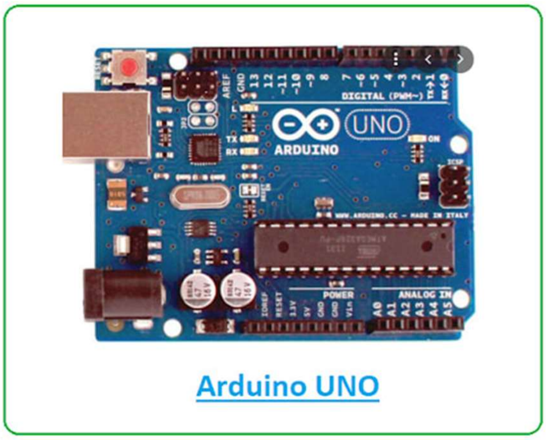


Fig 4.9. Arduino Uno

This board consists of a USB interface i.e., USB cable is used to attach the board with the pc and Arduino IDE (included development surroundings) software program is used to program the board.

The unit comes with 32KB flash reminiscence that is used to keep the range of instructions while the SRAM is 2KB and EEPROM is 1KB.

The running voltage of the unit is 5V which projects the microcontroller at the board and its related circuitry operates at 5V whilst the input voltage tiers between 6V to 20V and the advocated input voltage degrees from 7V to 12V

2. MQ-2 Sensor

This is a robust Gas sensor suitable for sensing Hazardous gases concentrations in the air.

It is a Metal Oxide Semiconductor (MOS) type Gas Sensor also known as Chemiresistors as the detection is based upon change of resistance of the sensing material when the Gas comes in contact with the material. Using a simple voltage divider network, concentrations of gas can be detected.

Fig 4.10. MQ-2 sensor

The tubular sensing element is made up of Aluminium Oxide (AL_2O_3) based ceramic and has a coating of Tin Dioxide (SnO_2). Tin Dioxide is the most important material being sensitive towards combustible gases.

When tin dioxide (semiconductor particles) is heated in air at high temperature, oxygen is adsorbed on the surface. In clean air, donor electrons in tin dioxide are attracted toward oxygen which is adsorbed on the surface of the sensing material. This prevents electric current flow.

In the presence of reducing gases, the surface density of adsorbed oxygen decreases as it reacts with the reducing gases. Electrons are then released into the tin dioxide, allowing current to flow freely through the sensor.

2. MQ-3 Sensor

It works in the similar way as the MQ-2 sensor but it is used to detect alcohol.

Fig 4.11. MQ-3 Sensor

3. Servo Motor

A servo motor (or servo motor) is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration. It consists of a suitable motor coupled to a sensor for position feedback.

Fig 4.12. Servo Motor

On detection of any presence of hazardous gas servo motor rotates and locks the engine.

Circuit Diagram

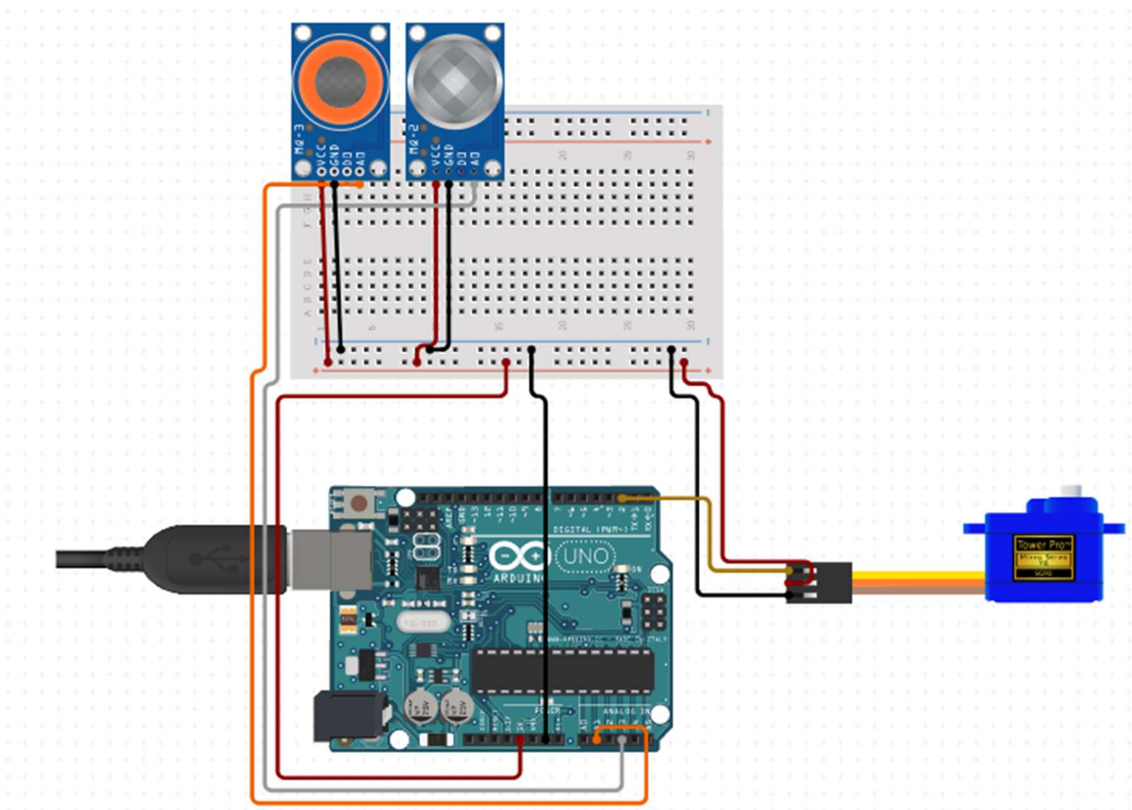


Fig 4.13. Circuit Diagram

Working of the model

As soon as the ignition key of the vehicle is turned on power is supplied to the installed system. After receiving input power, the heating element present inside the sensor starts working. As the tin dioxide gets heated up the donor electrons of tin dioxide get attracted by oxygen molecules thus blocking the current flow. But as soon as any hazardous gas is detected by the sensor this bond between oxygen molecule and donor electrons breaks thus giving voltage between 0-5V and so the analog reading detected by Arduino increases. As the value increases a signal is sent to the servo motor which is connected to the digital pin of the Arduino, servo motor receives the signal and locks the ignition key preventing the driver from driving further. Apart from engine locking the data that is collected by Arduino is sent for testing and training so that we can get 100 cent results.

After the sensors have detected the level of gases/chemicals present in the surroundings, the data is forwarded to help figure out if the individual is drunk. The data recorded over a particular interval time is tested through a machine learning model. The machine learning model is trained using already available data recorded with MQ3. Multiple algorithms (Machine learning and deep learning) were tried and tested and the three most significant of them are mentioned below.

Multi-Layer Perceptron: The feed forward neural network augmentation shown in Fig. 11 is a multi-layer perceptron (MLP). It has three layers, as shown in: an input layer, an output layer, and a hidden layer. The input layer receives the signal that will be processed. Prediction and categorization are tasks that fall under the output layer's purview. Between the input and output layers, the MLP's true computational engine is an arbitrary number of hidden layers. Data flows in the forward direction from the input to the output layer of an MLP, comparable to a feed forward network. The MLP's neurons are trained using the back propagation learning algorithm. MLPs are designed to approximate any continuous function and can handle problems that aren't linearly separable. MLP's most typical uses include pattern categorization, recognition, prediction and approximation.

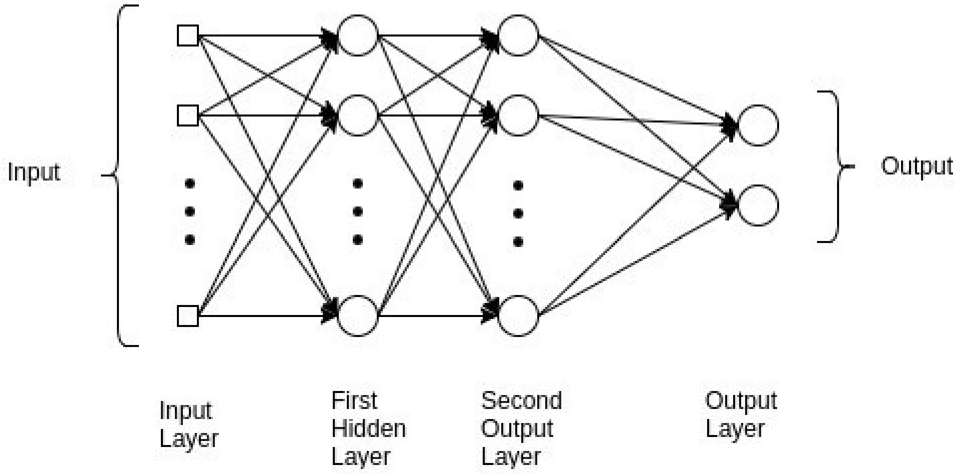


Fig 4.14. Multilayer Perceptron Model

Training of MLP: The degree of error in an output node j in the nth data point can be represented by:

$$e_i(n) = s_i(n) - y_i(n)$$

where s_i is the goal value and y_i is the perceptron's output value.

The node weights can then be modified depending on adjustments that reduce the overall output error, as determined by:

$$\varepsilon(n) = \frac{1}{2}\sum_j e_j^2(n)$$

The change in each weight is calculated using gradient descent.

$$\Delta w_{ji}(n) = -\eta \frac{\delta\varepsilon(n)}{\delta v_k(n)} y_i(n)$$

where y_i is the preceding neuron's output and η is the learning rate, which is chosen to ensure that the weights converge fast and without oscillations to a response.

The induced local field v_j, which fluctuates, determines the derivative to be determined. It is simple to show that this derivative can be simplified to for an output node.

$$-\frac{\delta\varepsilon(n)}{\delta v_j(n)} = e_j(n)\phi'(v_j(n))$$

where ϕ' is the derivative of the above-mentioned activation function, which does not vary. The analysis of a change in weights to a hidden node is more challenging, although it can be demonstrated that the appropriate derivative is:

$$-\frac{\delta\varepsilon(n)}{\delta v_j(n)} = \phi'(v_j(n))\sum_k -\frac{\delta\varepsilon(n)}{\delta v_k(n)} w_{kj}(n)$$

The change in weights of the k_{th} nodes, which represent the output layer, determines this. The output layer weights vary according to the derivative of the activation function to modify the hidden layer weights; hence this algorithm is a backpropagation of the activation function.

Support Vector Classifier: : The Support Vector Machine (SVM) shown in the Fig. 12 is a well known supervised learning algorithm that can be used to solve any of the two problems i.e. classification and regression. But it is mostly used in Machine Learning to seek solution to classification related problems.

The purpose of the SVM method is to determine the best line or decision boundary for categorising n-dimensional space into classes or labels so that subsequent data points can be easily placed in the right class. The best choosen boundary is known as the hyperplane.

SVM is used to select the extreme points/vectors that help build the hyperplane. The algorithm is known as a Support Vector Machine, and support vectors are the extreme examples.

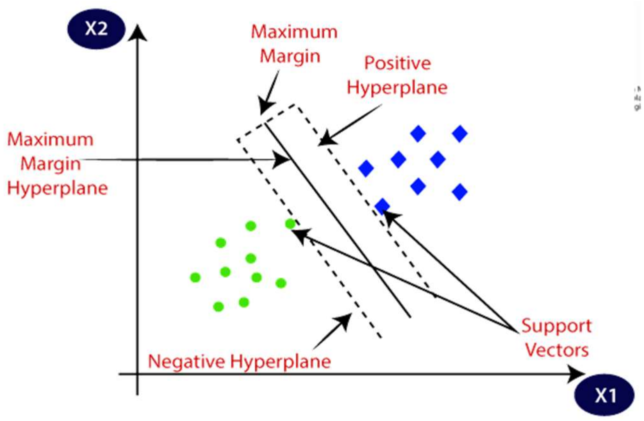


Fig 4.15. Support Vector Machine

Random Forest: The supervised machine learning algorithm Random Forest, shown in Figure 4.16, is often used to address classification and regression issues. It uses the majority vote for classification and the average for regression to generate decision trees from various samples.

The Random Forest Algorithm's ability to handle data sets with both continuous and categorical variables, as in regression and classification, is one of its most important features. It outperforms the competition when it comes to classification problems.

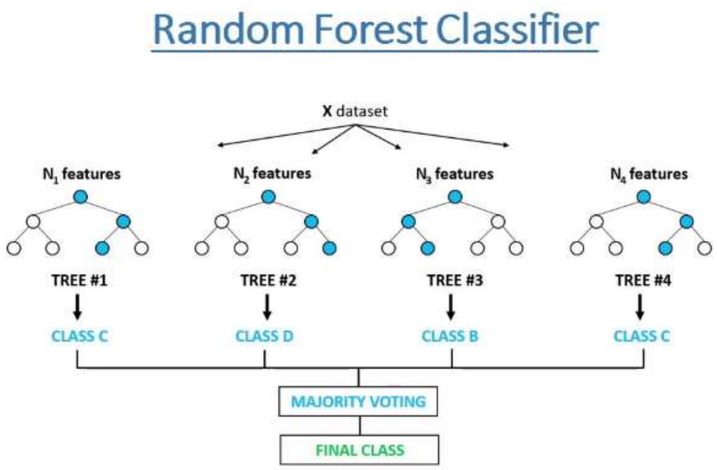


Fig 4.16. Random Forest Classifier

Functioning:

- In Random Forest, n random records are chosen at random from a data set with k records.

- For each sample, a unique decision tree is created.
- Each decision tree will produce a result.
- For classification and regression, the final output is based on Majority Voting or Averaging, respectively.

4.1.3. Driver Monitoring System

The driver monitoring system broadly aims to answer two questions, whether the driver is drunk or if he is drowsy using CNN for facial analysis. Drunk and sleepy driving both can be reasons for dangerous accidents and hence this is an important module of accident prevention.

4.1.3.1. Drunk Driver Detection by facial analysis:

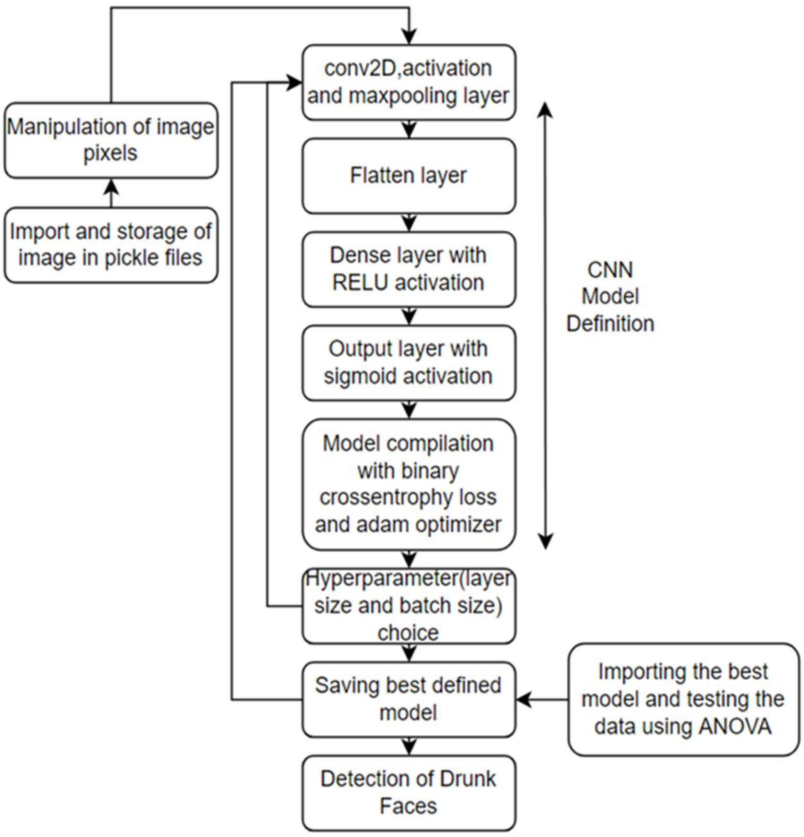


Fig 4.17. Flowchart of Drunk Detection System

The above block diagram displays the process flow of this system. A CNN model is defined and trained using intoxicated faces dataset.

Drunk Face Dataset: The predefined dataset consists of drunk face images of 41 subjects, 16 images for each subject. The drunk images were collected from YouTube videos of drunk reviews, drunk challenges, drunk reactions etc. Some images of sober subjects are also present in it and divided into sober and drunk classes. The dataset contains images from all age groups, gender, color etc, so the data can be treated as an unbiased data. For SVM classifiers, a test-train split of 25 and 75 percent is used, whereas for CNN classifiers, a training set of 65 percent, validation set of 15 percent and testing set of 20 percent is used.

The main stages in the paper are

Data Preprocessing: Thermal data is extracted from pixel data and it is labeled with 0 for sober and 1 for drunk. The total dataset is divided into 65 percent training data, 15 percent validation data, 20 percent testing data and stored in pickle files for later usage.

Image Manipulation: The raw image files need to be polished and manipulated before feeding it into a CNN network. Each image was flipped and concatenated in its original form. Pixel values were then normalized, followed by shuffling and reshaping of the entire dataset to avoid any biasing in the dataset.

Model Definition: A typical CNN model was defined as trained with binary-crossentrophy loss, Adam optimizer and 0.00001 learning rate. It is fitted with a training dataset. Hyperparameters like layer size and batch size are varied between 2,8,32 and 128 and the best model is saved.

The CNN architecture defined is as follows,

1. Three consecutive convolutional layers were defined with 2, 2 and 32, ReLU activation function and a kernel size of 3*3 was used for extracting high level features from the images.

2. Each convolution layer is followed by a maxpool layer of 2*2 kernel.

3. The flatten layer is defined after that for conversion of the input into a one-dimensional vector.

4. This is followed by a dense layer for converging to an output.

5. The output i.e., the last layer is characterized by sigmoid activation function.

The CNN model is then fitted with the perceived intoxicated faces dataset and tested with the validation dataset.

Drunk face detection: Real-time video of the driver is captured, image frames are extracted from it and fed into the defined model. This successfully detects whether the face inserted is drunk or not.

4.1.3.2. Drowsiness detection using facial analysis:

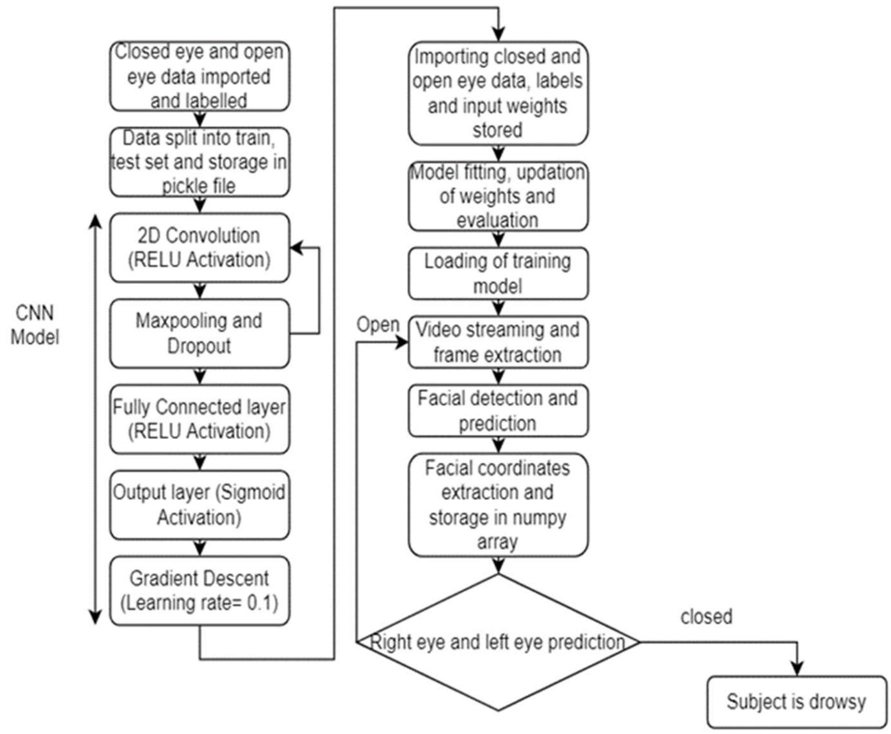


Fig 4.18. Flowchart for Drowsiness Detection System

Drowsy Face Dataset: This dataset consists of 2423 subjects, among which 1192 have eyes closed whose data is collected from the internet and 1231 subjects have eyes open from a labeled face in a wild database. Both datasets are further divided into left or right eye. The detection needed the faces to be cropped in 100*100 pixels and the exact eye area of 24*24 pixels was extracted from it for detection.
The flowchart above in Fig 4.18 broadly depicts the drowsy detection procedure.

Eye-Pre-processing: The open and closed eyes dataset are imported and then closed are annotated as 1 and open as 0. Further, they are split in a train-test ratio of 80 and 20 percent. The data is stored in pickle files for future usage.

Eye-training: In the training phase, a Sequential CNN model is trained whose architecture is explained in detail below with the train set stored in pickle files. Specifically, 12 epochs were trained since it showed optimum accuracy by trial and error method. The model is then saved with updated weights.

The Sequential CNN architecture used is as follows

Input Layer : 24*24 pixel images are fitted in the input layer of the CNN model.

Convolution Layer : Convolution layers with 32,64 and 128 filters and ReLU activation are used respectively with alternate maxpooling(2*2 kernel) and dropout layers.

FC Layer : The three dimensional image matrix is now converted into one dimensional vector in this layer. This also uses RELU activation function.

Output layer: The output layer is the final layer giving a decision of whether driver is drowsy or not using sigmoid activation function.

Backpropagation Algorithm: This uses the stochastic gradient descent for updation of weights with a learning rate of 0.01. The entropy between true labels and predicted labels are calculated using the binary-crossentrophy loss parameter.

Drowsiness detection: The real-time video of the driver is captured while driving and image frames are extracted from it. The images are tested with the trained model and faces showing closed eyes for more than 20 frames are detected as drowsy.

CODES AND STANDARDS

The codes and standards used in this project are as follows: -

1. Arduino IDE 2.0

   The Arduino IDE is the well-known software we all use to program our boards. Its development started in 2005 based on the graphical interface of the Processing project and has never stopped since. During these years, countless hours of development by the Arduino team with the help of a vibrant community made the Arduino IDE the de facto standard for electronics prototyping. Arduino IDE 2.0 beta carries a modern editor and provides a better overall user experience thanks to a responsive interface and faster compilation time. Don't be afraid of trying it today: the upgrade will be frictionless as the interface will look very familiar. The new IDE is based on the Eclipse Theia framework, which is an open-source project based on the same architecture as VS Code (language server protocol, extensions, debugger). The front-end is written in TypeScript, while most of the back end is written in Golang.

2. Python 3.10

   Python 3.10.0 is the most recent major release of the Python programming language, and it includes a slew of new features and improvements. Python has grown in popularity through time to become one of the most widely used and recognized programming languages on the planet. Machine learning, web development, and software testing are just a few examples of where it's used. It is appropriate for both developers and individuals with no prior knowledge in this field. It's a high-level scripting language that's interpreted, interactive, and object-oriented, with a focus on readability. It typically uses words used in normal English language instead of punctuation, and it has fewer syntactical structures than other languages.

a. CONSTRAINTS, ALTERNATIVES AND TRADEOFFS

CONSTRAINTS

1. Computational Time: Computation time of the CNN based algorithms with images extracted from real-time video was much more than what could be spared for instance-to-instance detection.

2. Sensor sensitivity: Since the sensor was moderately sensitive, sometimes introduction of a particular kind of gas resulted in slight change in all the values. So, it can be said the sensor was not sensitive enough to do clear classification between the gas components.

3. Outliers: Some outliers or erroneous data showing excessively high or low values were present in small numbers due to experimental errors or unexpected situations.

ALTERNATIVES

1. Instead of using a computer CPU we can use a GPU based atmosphere that expedites the computation speed manifold and hence makes it very suitable to be used in real-time and even in situations of very fast driving.

2. Instead of using these sensors, industries dealing with hazardous and alcoholic gases have more sensitive sensors which can be used to get more accurate and usable data.

3. Excessive pre-processing and data visualization were carried out to ensure that the data is well within range and ready for analysis.

TRADE OFFS

1. The management and preprocessing of these huge datasets and making it suitable to be used in the algorithm was a huge challenge. Hence, normalization and manipulation of pixel values, converting into grayscale, cropping into proper sized image to be used as input to CNN networks has to be done that could have ignored few important features.

2. Though trained with two very different datasets i.e., intoxicated faces and open and closed eyes dataset there can be situations where the system can wrongly predict drunk as drowsy or vice versa due to similarity in facial changes but the backup of detection of alcoholic gases will ensure whether possibility of being drunk is there.

3. Since MQ3 is capable of detecting smoke and CO as well there were situations where once alcohol is detected, the same sensor was not detecting smoke for a certain period of time till it will come to its normal state. To solve this, we introduced two sensors one for smoke and CO alone and the other detecting all three to deal with situations where more than one gas is present.

3. SCHEDULE, TASKS AND MILESTONES

SCHEDULES AND TASKS

In our first schedule, we studied the existing works and identified the key reason behind accidents. It includes learning about the operations and specifications of components required to make our system to work as a single solution to all the problems.

In our second schedule, we designed the algorithm for the working of the main system and associated components of the proposed overall system. Also, for implementing the hardware we prepared a base model using proteus.

In our next schedule, we implemented a hardware system and also worked on increasing the accuracy and also, we used the values that we received from the Arduino to test and train our model.

MILESTONES

Milestone 1

The first milestone was successfully determining the software and hardware provisions that would constitute our proposed design for Driver Accident Detection System

Milestone 2

The second milestone was developing and implementing the different algorithm for all the three systems i.e., Vehicle Collision System, Drowsiness Detection System and Hazardous gas and Alcohol Detection system respectively and preparing a base model on proteus. Python was used to develop different models in Jyupter.

Milestone 3

The third milestone was implementing the hardware using an Arduino, sensors and servo motor. Also, we used the values obtained from Arduino to train and test the model and gain high accuracy.

4. PROJECT DEMONSTRATION

Vehicle Collision Detection System:

This system is implemented using VGG and inception i.e., two CNN based networks followed by Long Short-Term Memory algorithm. Initially an open accident image dataset is taken and preprocessed to generate the perfect size pixelated and gray-scaled images for training and testing. Further, the CNN models are defined and trained with the same. The algorithm that showed best accuracy was chosen and real-time video was fed as input to it. The output was a predicted accidently probability that was displayed on the display window at every instance of driving.

Hazardous Gas and Alcohol Detection System:

As soon as the ignition key of the vehicle is turned on power is supplied to the installed system. After receiving input power, the heating element present inside the sensor starts working. As the tin dioxide gets heated up the donor electrons of tin dioxide get attracted by oxygen molecules thus blocking the current flow. But as soon as any hazardous gas is detected by the sensor this bond between oxygen molecule and donor electrons breaks thus giving voltage between 0-5V and so the analog reading detected by Arduino increases. As the value increases a signal is sent to the servo motor which is connected to the digital pin of the Arduino, servo motor receives the signal and locks the ignition key preventing the driver from driving further. Apart from engine locking the data that is collected by Arduino is sent for testing and training so that we can get 100 cent results.

First of all, the files were imported using pandas. Then the entire dataset is then scaled using MinMaxScaler to boost the speed and the efficiency. Following that, the data is split in testing and training dataset as in an 8:2 ratio respectively. Then various machine learning models are trained using the data and using the test data, accuracy is determined. After that, Accuracy and Cross validation scores are calculated to figure out the metrics. Classification report and confusion matrix are used to get a closer look at precision, recall and f1-score.

Drunk Face Detection: This system used Sequential CNN algorithm as the classification algorithm. It was trained with 65% of pre-processed intoxicated faces dataset, validated with 15% of the same and tested with remaining 20%. The test image data was given as input to the model defined and the number of faces labeled drunk and predicted drunk was huge showing the efficiency of the model formed.

Drowsy Face Detection: This system used a Sequential CNN architecture but used open and closed eye images for training. Hence, the images trained the model in such a way that it could efficiently detect whether the driver's eye was open or closed. In real-time, the driver's video was recorded, images extracted from it and fed to the model for detection. If the system detected the eye to be closed for more than 20 frames, it would conclude the driver to be in a sleepy state and hence ring an alarm.

5. COST ANALYSIS, RESULTS AND DISCUSSION

a. COST ANALYSIS

Table 7.1. Cost Analysis

COMPONENTS	UNIT PRICE (Rs.)	QUANTITY
Arduino UNO	749	1
MQ-2	210	1
MQ-3	237	1
Servo Motor	299	1
Jumper wires	240	1
TOTAL	**Rs. 1,735**	

b. RESULTS AND DISCUSSION

7.2.1. Vehicle Collision System

The vehicle collision system was implemented using two different algorithms. VGG Net used fixed kernels for convolution and max pooling with a fixed stride of two. All the variable kernels previously used in Alex Nets to determine high- and low-level features were now replaced by low fixed kernels (like 5*5 by two 3*3 kernels, 7*7 by three 3 *3 kernels), capable of efficiently determining the same features. This reduces the number of computations required since the same kernel is used multiple times and also reduces overfitting compared to conventional nets like Alex Net.

Inception, however, is an even better solution. Though it uses variable kernels like 5*5 for global features, 3*3 for area-specific features and so on and equivalent time consumption as in Alex Net but its first i.e., 1*1 layer is used for depth reduction, and results from different depths are concatenated to determine a final result, which ensures almost 10 times less memory consumption and improved accuracy.

Despite this comparison, both the networks could detect collision or safe situations for a car depending on the other vehicles in its vicinity.

Fig 7.1. Vehicle Collision with No car in front

The image above shows a situation where there is no car in front of the vehicle in consideration and hence it's not showing any threats of an accident. It is showing the situation as safe for driving.

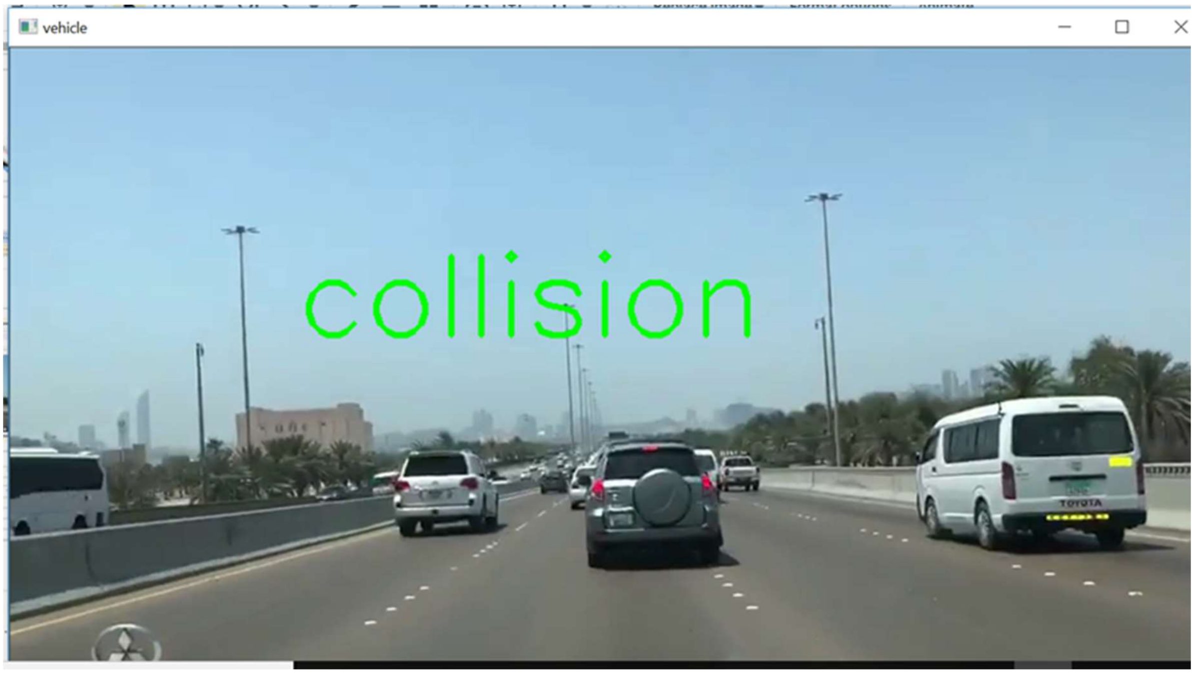


Fig 7.2. Vehicle Collision with Cars in front

In the second situation, there is a car just in front and in close proximity to the vehicle in consideration and hence it is notifying the situation as Collision possible. These detections are taking place in real-time and hence give results for every changing surrounding of the vehicle.

7.2.2. Hazardous and Alcohol detection System

The system used two different kinds of sensors MQ-2 and MQ-3 to detect hazardous gases and alcohol respectively. Here are the results of the values detected on testing the system by spraying deodorant for alcohol detection, using matchsticks for smoke detection and also exposing the system for LPG detection.

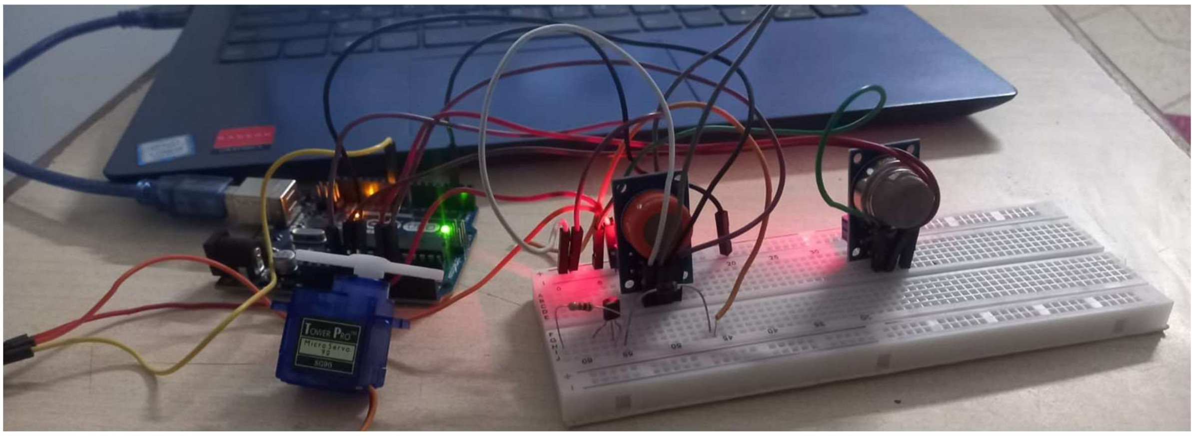


Fig 7.3. Hardware Setup

Spraying Deodorant

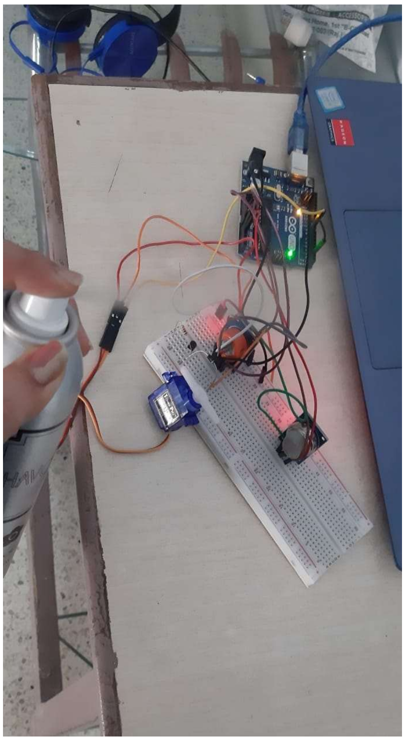


Fig 7.4. Spraying Deodorant

Serial Monitor Readings

0.74511	0.00004	0.00062	522	1
0.33496	0.00004	0.00062	519	1
0.2534	0.55023	0.00061	519	1
0.00908	0.00006	0.00056	512	1
0.00592	0.00005	0.00062	513	1
0.28331	0.00005	0.8891	512	1
0.24844	0.00005	0.92727	512	1
0.26192	0.00005	0.91441	513	1
0.29081	0.00005	0.96028	514	1

Fig 7.5. Deodorant Readings

The value of the alcohol increases as the deodorant is sprayed in the room.

Lightning Matchstick

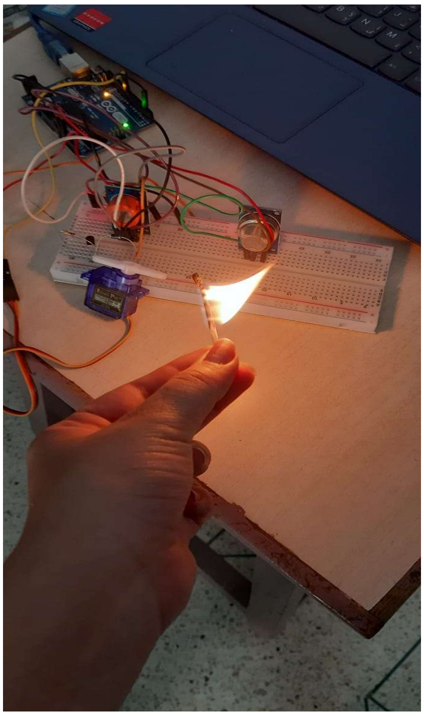


Fig 7.6. Lightning Matchstick

Serial Monitor Readings

0.00647	0.00408	6.19763	322	3
0.00672	0.01637	11.34891	323	3
0.00628	0.00371	1.77925	323	3
0.00532	0.00485	1.09488	312	3
0.00565	0.01561	1.95489	322	3
0.00552	0.00797	151.63421	331	3
0.00572	0.00348	11.32647	339	3

Fig 7.7. Matchstick Readings

The value of smoke goes too high after matchstick is lit.

Exposing system to LPG

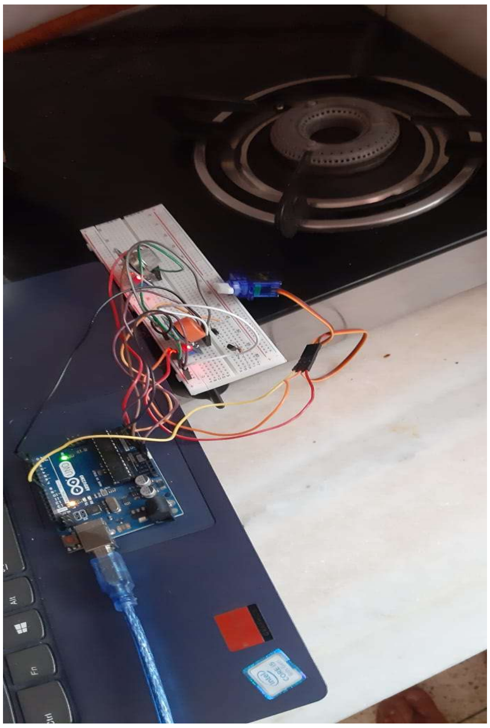


Fig 7.8. LPG

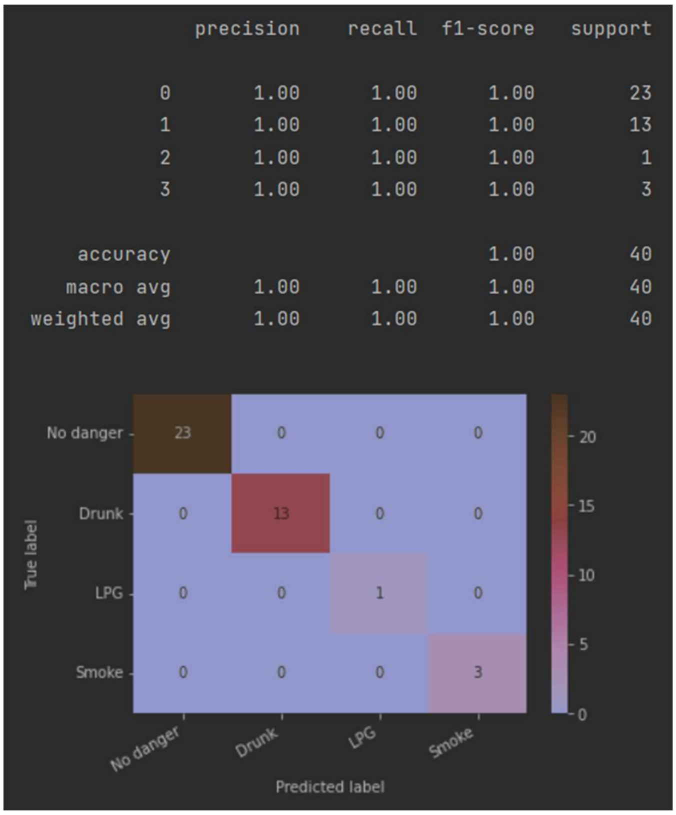
Serial Monitor Readings

21.03282	0.01239	0.018	325	2
31.18002	0.00326	0.04743	337	2
22.88868	0.00371	0.01632	340	2
37.0731	0.00485	0.02085	340	2
31.30726	0.01561	0.0315	347	2
3.37941	0.00797	0.01514	343	2
66.36772	0.00348	0.01572	343	2
6.3577	0.00508	0.01936	346	2

Fig 7.9. LPG Readings

The value of LPG increases from almost 0 to above 20 as the system is exposed to LPG and then decreases as the gas is switched off.

Following is the Confusion matrix after the machine learning model was trained and tested. The accuracy is 100% for this set of data. It depicts several data such as precision, recall and f1-score. These act as the performance metrics as they tell how well the machine learning model is working. This will be more useful in the cases where the accuracy isn't exactly 100%. In such cases, precision and such other metrics come in handy.



	precision	recall	f1-score	support
0	1.00	1.00	1.00	23
1	1.00	1.00	1.00	13
2	1.00	1.00	1.00	1
3	1.00	1.00	1.00	3
accuracy			1.00	40
macro avg	1.00	1.00	1.00	40
weighted avg	1.00	1.00	1.00	40

Fig 7.10. Confusion Matrix of Gas Detection

After the data was collected and recorded in a file, it was used to train machine learning models using multiple algorithms. Not all performed very well. Multiple tests were run and some of the algorithms and their results are mentioned here.

The following 3 algorithms were used and their respective accuracies and cross validation scores were as follows in Table 7.2.

Table 7.2. Comparison of gas detection algorithms

Algorithm	Accuracy	Cross Validation Score
Multilayer Perceptron	Almost 100%	0.65
Random Forest	Almost 100%	0.95
Support Vector Classifier	77.5%	0.6

Random forest and Multilayer Perceptron had almost a 100% accuracy but the cross-validation score was very low in the case of MLP compared to Random Forest. Support Vector Classifier had the quickest run time but the accuracy and cross validation score are very low. Hence, Random Forest will be used to train the model and for analyzing the presence of alcohol and other hazardous gasses.

7.2.3. Driver Monitoring System

7.2.3.1. Drunk Face Detection:

A convolution model when trained with the intoxicated faces dataset could successfully determine most of the sober and drunk faces accurately. Before the classification, the images needed manipulation and preprocessing. A typical pre-processed image in RGB and grayscale looks like the one below.

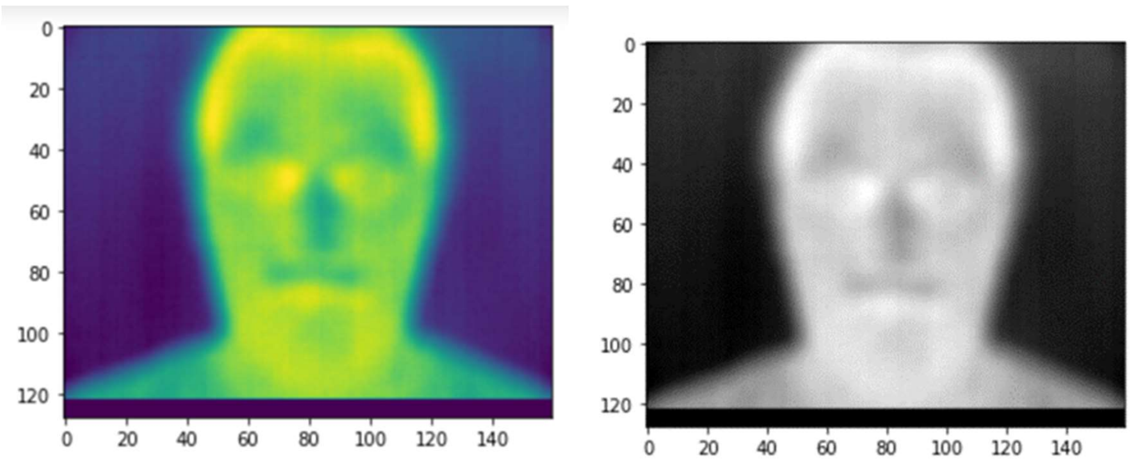


Fig 7.11. Augmented Drunk Face Image

The training was carried out for 1000 kernels. The hyperparameters like layer size, batch size for the training is chosen based on trial-and-error method and the hyperparameters for the model that gave the highest accuracy are saved as the final model and used for testing. A typical accuracy plot for train, test and validation sets for a particular set of hyperparameters is shown below.

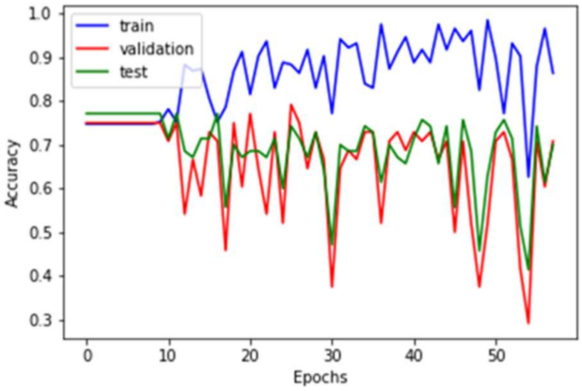
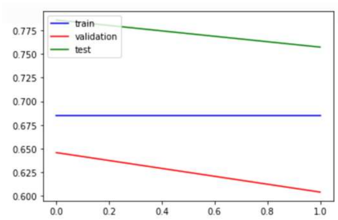
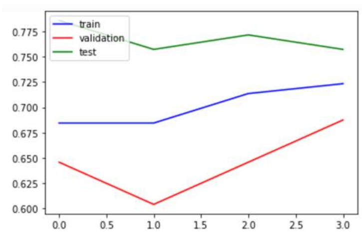
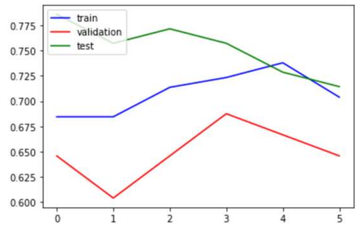
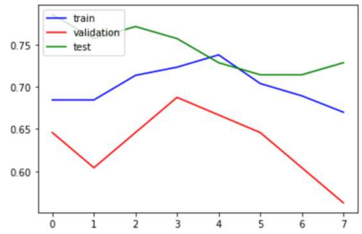
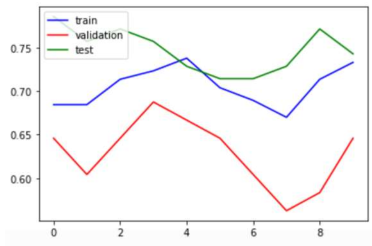


Fig 7.12. Accuracy plot for Test, Train and Validation Data

The CNN model for drunk detection was initiated training for 1000 epochs and was only interrupted in case it showed the same accuracy for more than 30 epochs. It was then considered as the saturation point for the training process. The training accuracy loss for the first 1,3,5,7 and 9 epochs are listed below in Table 7.3.

Table 7.3. Loss and Accuracy Table for Drunk Detection

Epoch	Loss	Accuracy	Plots
1	0.2154	0.5485	
3	0.2138	0.6117	

5	0.2129	0.6602	
7	0.2115	0.6699	
9	0.2106	0.6845	

This training is carried out for hyperparameter values by varying it in 2,8,32,128 and accuracy is stored for each model. The best accuracy is shown by 2,2,32, 8 parameters. Parameters above that showed overfitting calculating accuracy as high as 96.88% while the basic set 2,2,2,2 showed accuracy as low as 69.28%. Hence an optimum accuracy is chosen to be suitable for the detection procedure.

The convolution model was then tested to see how many actually sober faces (faces labeled as 0) can the model detect (predict them as 0). This was executed for all the test data and the few results are shown below.

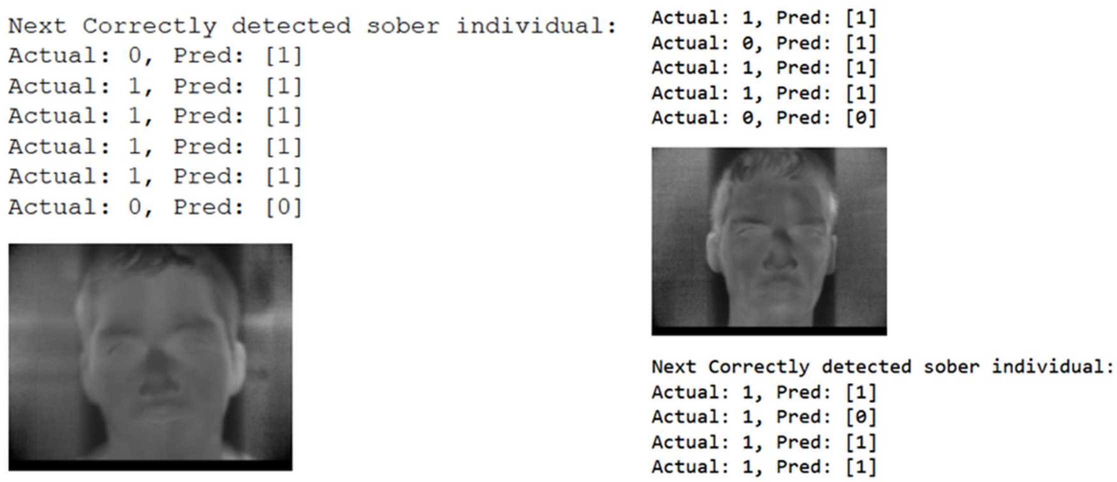


Fig 7.13. Sober Faces Detected

All those displayed as actual: 1 and pred: 1 are actual drunk faces that the model correctly detected as drunk. The particular model showed an accuracy of 84.285 % which is much higher than any other classification model, e.g., an SVM model trained with the same dataset showed an accuracy of only 57.89%. The image below depicts the accuracy matrix for the CNN model.

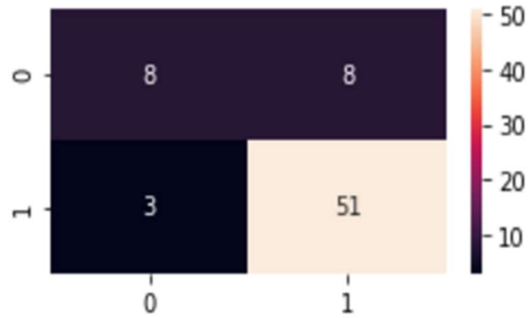


Fig 7.14. Confusion Matrix for Drunk Detection

7.2.3.2. Drowsy Detection:

The system displayed best results when trained for 12 epochs and tested and they are given in Table 7.4. The learning rate parameter is initialized to 0.01 and a decay rate of 10^{6}.

Table 7.4. Loss and Accuracy Table for Drowsy Detection

Loss	**Accuracy**	**Value Loss**	**Accuracy Loss**
0.6904	0.5211	0.6851	0.6308
0.6376	0.6397	0.6403	0.6227
0.6092	0.6646	0.6193	0.6531
0.5893	0.6899	0.5714	0.6846
0.5203	0.7461	0.4573	0.7647
0.3965	0.8258	0.2962	0.9026
0.2753	0.8908	0.2150	0.9300
0.2199	0.9198	0.1558	0.9533
0.1897	0.9274	0.1282	0.9594
0.1666	0.9370	0.1368	0.9584
0.1582	0.9358	0.1553	0.9402
0.1320	0.9472	0.1060	0.9645

The number of epochs is determined by trial-and-error methods. The corresponding test accuracy scores for each case are tabulated in Table 7.5.

Table 7.5. Loss and Accuracy for different epochs

Number of Epochs	**Loss Score**	**Test Accuracy (percent)**
10	0.1721	95.13
11	0.1518	95.43
12	0.106	96.45
13	0.1293	95.84
14	0.096	96.85

The above observation shows that while for epochs 10,11,13 the accuracy rates are less so they are discarded and for 14 epochs it’s as high as 97 %. Though this could have been an optimum parameter but this tends towards overfitting. Hence 12 epochs are chosen as optimum choice.

The drowsy detection model was trained with 3938 samples and tested with 986 samples. It was able to achieve an accuracy of 96.45 % and a loss of only 0.106 after 12 epochs. The number of epochs was chosen by trial-and-error method based on the best accuracy shown by it. Hence the model is suitable for real-time drowsy detection.

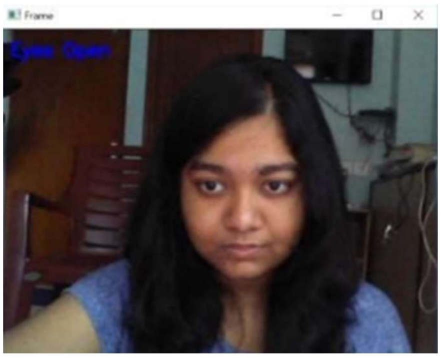


Fig 7.15. Eyes Open Instance for Drowsiness Detection

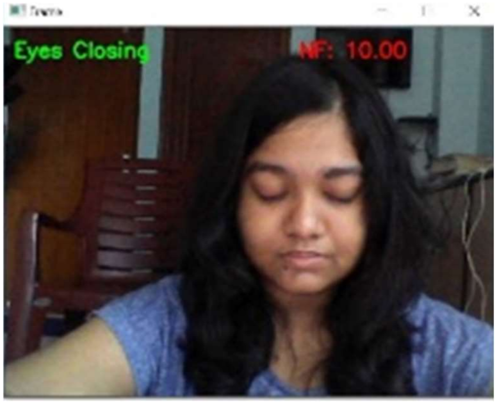


Fig 7.16. Eyes Closed Instance for Drowsiness Detection

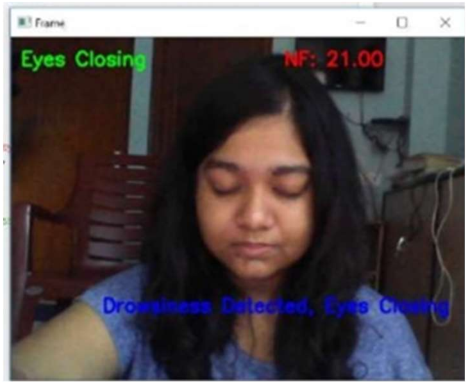


Fig 7.17. Eyes closed for more than 20 frames

A threshold of 20 is set as a bench of closed eye frames above which the person is detected as drowsy. Since the number of frames is 21, the message drowsiness detected is displayed and an alarm is rung to wake the person. Hence an efficient eye drowsiness system is developed, successfully tested and verified. The system initially shows “Open Eyes” shown in Fig. 29 when the subject is perfectly awake. As soon as the eyes are closed, the system recognizes the same and starts showing “Eyes Closing”, shown in Fig. 30. It waits for 20 frames of closed eyes to confirm that a blink is not detected as drowsiness. After 20 frames, the system starts showing “Drowsiness Detected” as shown in Fig. 21. This is the entire procedure of detection.

6. SUMMARY

In a world where road accidents are one of the major reasons for human life losses, such safety measures are desired and needed that can handle various circumstances and factors that can lead to accidents. The National Crime Record Bureau (NCRB) showed that there were almost 3.55 lakh road accidents in India in 2020 and about 1.33 lakh were injured. More than 60% of these road accidents occurred because of over-speeding, making it the obvious leading reason. It was followed by dangerous and careless driving.

This project deals with many such scenarios and alerts the driver before the problem goes awry. The first component handles the detection of alcohol and hazardous gases in the vehicle's environment and alerts in case the levels go above the threshold, and hence preventing fatal accidents.

There is also an Eye Drowsiness system that can detect drowsiness very accurately. Tendency of eye closing frequently, most commonly when one is sleepy is considered as sign of drowsiness and in any such case this system perfectly detects and alarms the driver about this condition. This system will be a very important device to reduce road accidents and death rate due to this.

It is accompanied by a system that detects the driver's face and by analyzing multiple factors, it alerts when the driver is drunk. The last system alerts the driver in case they are overspeeding which can lead to vehicle collisions. It notifies one when they are getting too close to another vehicle which in most cases will lead to fatal accidents. All the components combined transform this project into something phenomenal. It is expected to help reduce the accident and improve driving conditions worldwide.

References

[1]https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries.

[2]Francisco Alonso1, Cristina Esteban1, Constanza Calatayud1, Jaime Sanmartín, "Speed and Road Accidents: Behaviors, Motives, and Assessment of the Effectiveness of Penalties for Speeding", Accepted December 12, 2013, American Journal of Applied Psychology, 2013, Vol. 1, No. 3, 58-64

[3] Xiaohua Zhao, Xingjian Zhang and Jian Rong, "Study of the Effects of Alcohol on Drivers and Driving Performance on Straight Road", published:13 Feb 2014, vol 2014.

[4] Qiang Luo, Xiaodong Zang, Jie Yuan, Xinqiang Chen, Junheng Yang, and Shubo Wu, "Research of Vehicle Rear-End Collision Model Considering Multiple Factors", published:10 Apr 2020, vol 2020.

[5] Vaishnavi.M, Umadevi.V, Vinothini.M, Bhaskar Rao. Y, Pavithra.S, "INTELLIGENT ALCOHOL DETECTION SYSTEM FOR CAR", International Journal of Scientific & Engineering Research, vol 5, Issue 11, November-2014

[6]Ira C. Valenzuela, Lean Karlo S. Tolentino, Ronnie O. Serfa Juan, "Utilization of e-Nose Sensory Modality as Add-On Feature for Advanced Driver Assistance System", International Journal of Advanced Trends in Computer Science and Engineering, vol 8, No.4 , July-August 2019

[7] Chew Cheik Goh, Latifah Munirah Kamarudin , Ammar Zakaria, Hiromitsu Nishizaki, Nuraminah Ramli, Xiaoyang Mao, Syed Muhammad Mamduh Syed Zakaria , Ericson Kanagaraj, Abdul Syafiq Abdull Sukor and Md. Fauzan Elham, "Real-Time In-Vehicle Air Quality Monitoring System Using Machine Learning Prediction Algorithm", published: 21 July 2021

[8]Hong Men, Haiyan Liu, Yunpeng Pan, Lei Wang and Haiping Zhang, "Electronic Nose based on an Optimized Competition Neural Network", Sensors 2011, 11, 5005-5019; doi:10.3390/s110505005,published on 2011 May 4

[9] Pei-Feng Qi, Qing-Hao Meng and Ming Zeng, "A CNN-Based simplified data processing method for electronic noses"

[10] Kavish Atul Sanghvi, “Drunk Driving Detection, Computer Science and Information Technology”, 6(2): 24-30, 2018

[11] Sergio Robles-Serrano, German Sanchez-Torres, and John Branch-Bedoya, “Automatic Detection of Traffic Accidents from Video Using Deep Learning Techniques” published: 9 November 2021

[12]. Jagannath Aghav, Poorva Hirwe, Mayura Nene, “Deep learning for Real Time Collision Detection and Avoidance”

[13] V. B. Kamath, S. S. Pai, S. S. Poojary, A. Rastogi and K. S. Srinivas, "Drunkenness Face Detection using Graph Neural Networks," 2021 IEEE International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS), 2021, pp. 1-6, doi: 10.1109/CSITSS54238.2021.9682956

APPENDIX A

Publication Information

Manuscript number: AMAR-D-22-00088

Analytic Methods in Accident Research
Development of an Integrated Driver Accident Prevention System
--Manuscript Draft--

Manuscript Number:	
Article Type:	Full length article
Keywords:	Accident Prevention; Road Safety; Convolutional Neural Network; Machine Learning; Sensors
Corresponding Author:	Naveen Kumar Vaegae, Ph.D, M.Tech, B.Tech VIT University Vellore, Tamilnadu INDIA
First Author:	Kathakoli Sengupta
Order of Authors:	Kathakoli Sengupta
	Abhyuday Tripathi
	Nandini Awtani
	Naveen Kumar Vaegae, Ph.D, M.Tech, B.Tech
Abstract:	Recent reports show that there have been around 12256 road accidents due to drunk driving in India in the year 2019. Given the surge of road vehicles in every city in the current scenario and its necessity in people's daily lives it has become inevitable to come up with a solution to accident problems. The paper addresses three major causes of accidents i.e., collision with another vehicle, fire outbreak due to overheating and driver's lack of concentration due to drunk or drowsy state and proposes prediction system designs prior to any of these incidents. The project is implemented using sensors, microcontroller and camera for real-time data collection and CNN and ML algorithms for its classification and decision making. A comparison of accuracy of the algorithms was made and the best in each case showed a detection accuracy of around 85%. The system is cost-effective since it uses very less software, robust, accurate enough and hence an appropriate solution for real-time road safety and risk management.

From: em.amar.0.7b0d25.307629ef@editorialmanager.com <em.amar.0.7b0d25.307629ef@editorialmanager.com> on behalf of Analytic Methods in Accident Research
Sent: Monday, May 2, 2022 11:52 PM
To: Naveen Kumar Vaegae
Subject: Submission to Analytic Methods in Accident Research - manuscript number

This is an automated message.

Manuscript Number: AMAR-D-22-00088
Development of an Integrated Driver Accident Prevention System

Dear Dr. Vaegae,

Your above referenced submission has been assigned a manuscript number: AMAR-D-22-00088.

To track the status of your manuscript, please log in as an author at https://www.editorialmanager.com/amar/, and navigate to the "Submissions Being Processed" folder.

Thank you for submitting your work to this journal.

Kind regards,
Analytic Methods in Accident Research

More information and support

You will find information relevant for you as an author on Elsevier's Author Hub: https://www.elsevier.com/authors

APPENDIX B

Curriculum Vitae

Name: Kathakoli Sengupta
Email id: kathakoli.sengupta2018@vitstudent.ac.in
Mobile No.: +91-8697671050
Address: C/O Surajit Sengupta, Flat B2, Block 9, 263, G.T. Road(South), Howrah-711102, West Bengal, India

Degree/Examination	Name of the Institution	University/ Board	Percentage/ CGPA
B. Tech (Electronics and Communication Engineering)	VIT University, Vellore	VIT University	9.24(2018-2022)
12^{th}	Gokhale Memorial Girls' School	CBSE	95%
10^{th}	Gokhale Memorial Girls' School	WBBSE	90.57%

Name: Abhyuday Tripathi
Email id: abhyudaytripathi15@gmail.com
Mobile No.: +91-9453545865
Address: Village + Post - Patherwa, District - Kushinagar, 274401, Uttar Pradesh

Degree/Examination	Name of the Institution	University/ Board	Percentage/ CGPA
B. Tech (Electronics and Communication Engineering)	VIT University, Vellore	VIT University	8.38 (2018-2022)
12^{th}	GN National Public School, Gorakhpur	CBSE	93.2%

10^{th}	Nav Jeevan Mission School, Kasia	CBSE	10.00 CGPA

Name: Nandini Awtani
Email id: nandiniawtani11@gmail.com
Mobile No.: 7231939556
Address: House No. 481 4th B road Sardarpura Jodhpur,Rajasthan

Degree/Examination	Name of the Institution	University/ Board	Percentage/ CGPA
B. Tech (Electronics and Communication Engineering)	VIT University, Vellore	VIT University	8.91(2018-2022)
12^{th}	Kautilya Sr. Secondary School	CBSE	82%
10^{th}	St. Patrik's Vidya Bhawan	CBSE	9.8