**ACCIDENT DETECTION SYSTEM BY USING CNN**

**A major project report submitted to**

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD**

**In partial fulfillment of the requirements for the award of the degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE & ENGINEERING**

Submitted by

**K.KAVYASREE (18C61A0522)**

**N.JAYASRI (18C61A0533)**

**R.RAKESH (18C61A0544)**

**M.SRAVANI (18C61A0529)**

**Under the Esteemed Guidance of**

**Dr.B.N.V.MADHUBABU M.Tech,Ph.D**

**Associate Professor**

****

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**MOTHER TERESA INSTITUTE OF SCIENCE & TECHNOLOGY**

**Accredited by NAAC with 'B' Grade**

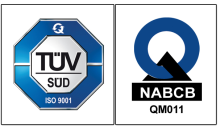
**(Approved by AICTE,Permanently Affiliated to JNTUH, Hyderabad)**

**AN ISO 9001:2008 CERTIFIED INSTITUTION**

**SANKETIKA NAGAR, SATHUPALLY-507303, KHAMMAM Dist., TELANGANA**

**(2021-2022)**

**MOTHER TERESA**

**INSTITUTE OF SCIENCE AND TECHNOLOGY**

**(Approved by AICTE, Govt. of Telangana**

**Affiliate to JNTUH & SBTET, Hyderabad)**

**Recognition under Section 2(f) & 12 (B) of the UGC Act, 1956**

**SANKETIKA NAGAR, KOTHURU (V), SATHUPALLY – 507303, KHAMMAM Dist., TELANGANA**

**Phone: 9494641251, Email ID: info@mistech.ac.in**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**CERTIFICATE**

This is to certify that the main project work entitled " **ACCIDENT DETECTION SYSTEM BY USING CNN**” is the bonafied work done by K.KAVYASREE(18C61A0522) N.JAYASRI(18C61A0533) R.RAKESH(18C61A0544) M.SRAVANI(18C61A0529)

In the Department of Computer Science and Engineering **Mother Teresa Institute of Science and Technology ,** Sathupally is affiliated to JNTU-Hyderabad in partial fulfillment of the requirements for the award of Bachelor of Technology in Computer Science and Engineering during 2018-2022.

This work has been carried out under my Guidance and Supervision.The results embodied in this project report have not been submitted in any university or Organisation for the award of any other degree or diploma.

**INTERNAL GUIDE HEAD OF THE DEPARTMENT**

**Dr.B.N.V.MADHU BABU M.Tech,Ph.D Dr.B.N.V.MADHU BABUM.Tech,Ph.D**

**ASSOCIATE PROFESSOR ASSOCIATE PROFESSOR**

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**DECLARATION**

We here by declare that the main project report entitled ‘**ACCIDENT DETECTION SYSTEM BY USING CNN**’ is a genuine project work carried out by us, in **B.Tech (Computer Science and Engineering)** degree course of JAWAHARLAL NEHRU TECHNOLOGY UNIVERSITY,HYDERABAD and has not been submitted to any other course or University for award of any degree.

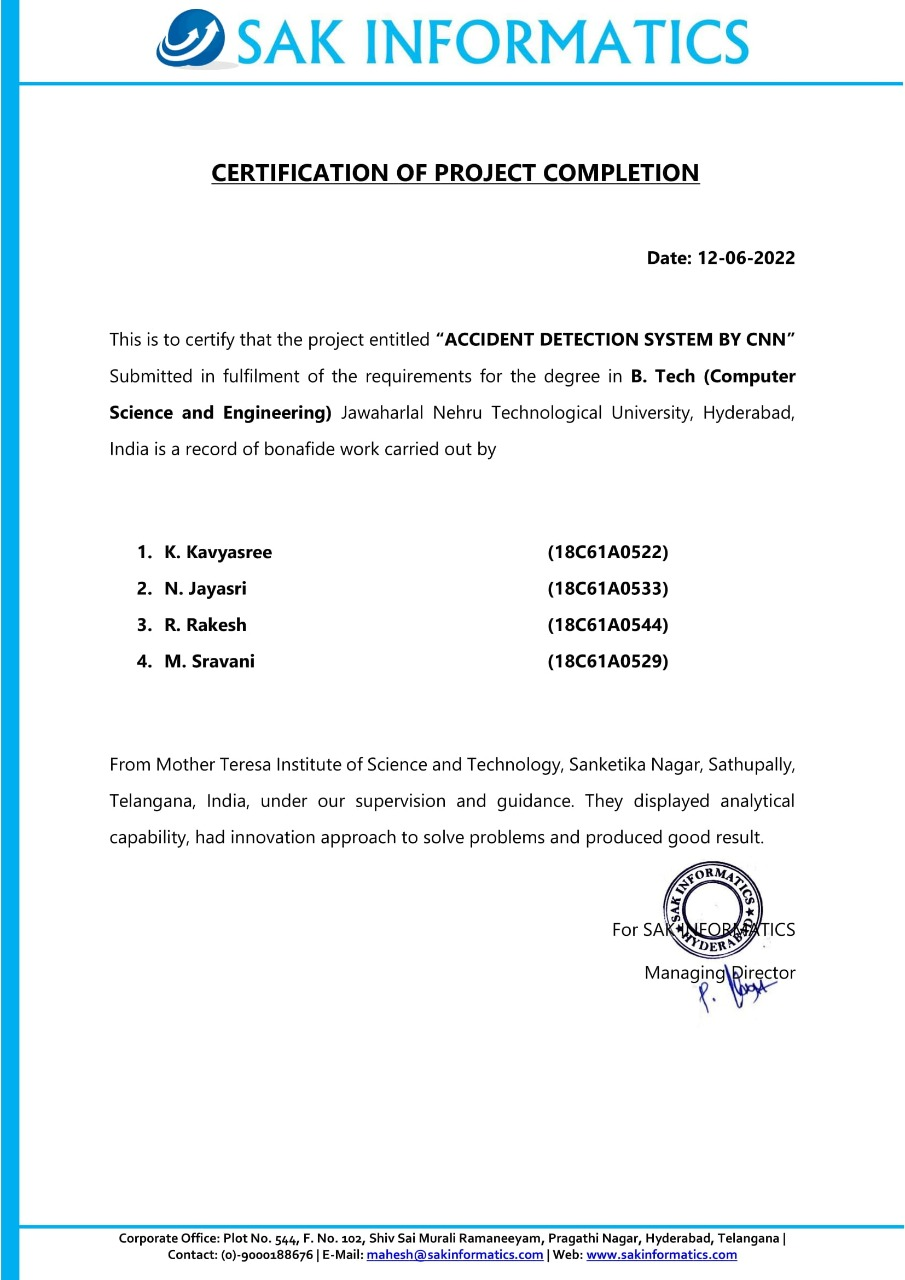
**Signature of the Students**

**K.KAVYASREE (18C61A0522)**

**N.JAYASRI (18C61A0533)**

**R.RAKESH (18C61A0544)**

**M.SRAVANI (18C61A0529)**



**ACKNOWLEDGEMENT**

This project work is the several days, study of the various aspects of the project development. During this the effort in the present study, we have received a great amount of help from our Secretary & Correspondent **Mr. C. SAMBASIVA RAO**, which we wish to acknowledge and thank from depth of our hearts.

We are grateful to **Dr. C. HARI KRISHNA**, principal of our college for his valuable suggestions.

We express our gratitude to **Dr. B.N.V. MADHU BABU** Head of the Department of Computer Science and Engineering.

We express our thankfulness and utmost regards to **Dr.SK.JAKIR HUSSAIN** Dean of Engineering ,Mother Teresa Institute of Science & Technology for his constant support and encouragement during the completion of the project.

We are very much obliged and thankful to our internal guide **Dr. B.N.V. MADHU BABU** for the super vision of our project, for providing this opportunity and constant encouragement given by her during course. We are grateful to his valuable guidance and suggestions during our project work.

Finally, we express our thanks to all our other Professors / Staff, Classmates,Bloved Parents,Friends, Neighbours and Industry Persons who helped us for the completion of this work.

**K.KAVYASREE (18C61A0522)**

**N.JAYASRI (18C61A0533)**

**R.RAKESH (18C61A0544)**

**M.SRAVANI (18C61A0529)**

|  |  |
| --- | --- |
|  | **MOTHER TERESA INSTITUTE OF SCIENCE & TECHNOLOGY**  Sathupally, Khammam Dist.,Telangana |

|  |  |
| --- | --- |
|  | **MOTHER TERESA INSTITUTE OF SCIENCE & TECHNOLOGY**  Sathupally, Khammam Dist.,Telangana |

|  |  |
| --- | --- |
|  | **MOTHER TERESA INSTITUTE OF SCIENCE & TECHNOLOGY**  Sathupally, Khammam Dist.,Telangana |

|  |  |
| --- | --- |
|  | **MOTHER TERESA INSTITUTE OF SCIENCE & TECHNOLOGY**  Sathupally, Khammam Dist.,Telangana |

**Vision and Mission of the Institute**

**Vision:**   
To be a state-of-the-art centre for learning with a social commitment transforming the youth into dynamic professionals.

**Mission:**

**IM1:** Foster unmatched excellence in professional education.

**IM2**: Provide quality eco-system to inspire learning aligned to needs.

**IM3**: Inculcate ethical and moral values to groom good citizens.

**IM4**: Involve in activities with team spirit and collaborations towards nation building.

**Department Vision**

To be in the frontier of Computer Science and Engineering education with moral values towards nation building.

**Department Mission**

**DM1**: To strengthen the core competence in Computer Science and Engineering through analytical learning.

**DM2**: To develop personal and professional skills and commitment to lifelong learning.

**DM3**: To uplift innovative research in Computer Science and Engineering to serve the needs of Industry, Government and Society

|  |  |
| --- | --- |
|  |  |
|  | **MOTHER TERESA INSTITUTE OF SCIENCE &TECHNOLOGY**  Sathupally, Khammam Dist.,Telangana |

**Department Program Educational Outcomes (PEOs)**

**PEO 1:** Develop innovative computing products and complex solution techniques by utilizing strong technical proficiency and critical thinking.

**PEO 2:** Demonstrate professionalism, ethical attitude, teamwork and leadership skills with lifelong learning.

**PEO 3:** Exhibit effective communication skill and social concerns to meet the challenges of software industries.

**Department Program Specific Outcomes (PSOs)**

**PSO 1:** Anability to specify, analyze & design a usable computing system that efficiently utilizes system software and hardware to cover current user requirement in a socially and economically acceptable form.

**PSO 2:** An ability to state, design and implement a secure and reliable information communication system by using concepts of computer networks, network security, information theory and parallel algorithm.

**PSO 3:** Successful Career and Entrepreneurship Perspectives: Fulfilling desire by attaining Employment, Excel in competitive examinations, Higher studies, Research, and initiate startup’s.

|  |  |
| --- | --- |
|  | **MOTHER TERESA INSTITUTE OF SCIENCE & TECHNOLOGY**  Sathupally, Khammam Dist.,Telangana |

**Program Outcomes (POs)**

1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data and synthesis of the information to provide valid conclusions.
5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts and demonstrate the knowledge of and need for sustainable development.
8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams and in multidisciplinary settings.
10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  | | --- | |  | |  |  |  |  |  |  |  |  |  |
|  |  | **MOTHER TERESA INSTITUTE OF SCIENCE & TECHNOLOGY** | | | | | |  |  |  |
|  |  | Sathupally, Khammam Dist., Telangana | | | | | |  |  |  |
|  | **DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING** | | | | | | | |  |  |
|  |  | **Course Outcomes for Project Work** | | | | |  |  |  |  |
|  | Course: | Project work | |  | AY: | 2021-22 |  |  |  |  |
|  | Class: | IV Year II Semester | | |  |  |  |  |  |  |

|  |  |
| --- | --- |
| **CO#** | **CO Statement** |
| **CO1** | Demonstrate the knowledge and skills of a professional engineer to make a project. |
| **CO2** | Collect latest information related to the project from various sources to analyse the project. |
| **CO3** | Apply knowledge and demonstrate to manage project in multidisciplinary areas. |
| **CO4** | Form a team for carrying the project and perform documentation effectively. |
| **CO5** | Prepare abstract for given project by identifying the requirements and prospective solution. |
| **CO6** | Design the necessary module of the selected project as per specifications. |
| **CO7** | Obtain and analyse the results of the designed module or circuit. |
| **CO8** | Develop a prototype/model of the project by distribution of tasks among the team. |
| **CO9** | Prepare a good report of the project as per the guidelines and present to the panel of experts. |
| **CO10** | Prepare Documentation and Presentation. |

Project In-charge

|  |  |
| --- | --- |
|  | **MOTHER TERESA INSTITUTE OF SCIENCE & TECHNOLOGY**  Sathupally, Khammam Dist., Telangana |

**Department of Computer Science and Engineering**

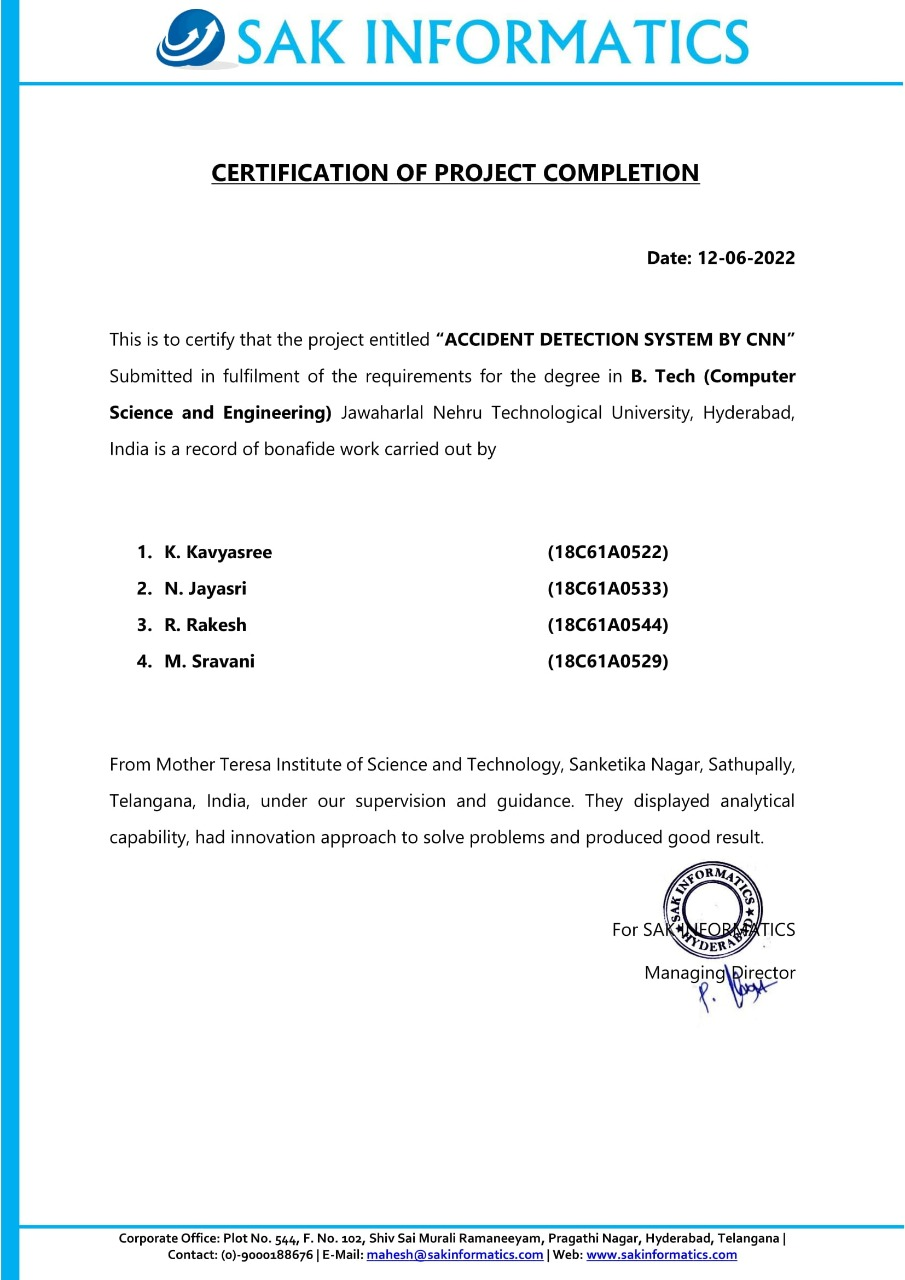
**Project – PO Mapping**

|  |  |
| --- | --- |
| Project Title: | ACCIDENT DETECTION SYSTEM BY USING CNN |
| Guide(s): | Dr.B.N.V.MADHU BABU |
| Student Name(s): | KUKKA KAVYA SREE, NAMA JAYASRI, REDLA RAKESH,  MOTAPOTHULA SRAVANI |
| Student Roll No(s): | 18C61A0522,18C61A0533,18C61A0544,18C61A0529 |
| Academic Year: | **2021-22** |

|  |  |  |
| --- | --- | --- |
| **Name of Course from which Principles are applied in this project** | **Description of the application, page number in the report** | **Attained PO** |
| Deep Learning | To detect road accidents and will provide an alert message to the most proximate control room. |  |
| UML Diagrams | Use Case diagram,Class diagram,Sequence diagram,Activity diagrams are used for implementation of the project. | PO3 |
| Project work | Acknowledged the various authors findings by way of references (Pg --) | PO8 |
| Project Seminar  Project work | Able to prepare a thesis and presented to a panel of experts | PO10 |
| Project work | Identifying and Analyzing the problem | PO3 |
| Project work | Creating,Selecting and Applying Appropriate techniques | PO5 |
| Project work | IR sensors and Arduino technology is the existing system |  |
| Project work | Deep learning techniques that use Convolutional Neural Network is the proposed system. |  |
| Algorithm | Convolutional Neural Network algorithm is used. |  |
| Project work | By using accident and non-accident images and videos we will train the system, Then it will able to detect the accidents. |  |
| Project work | The final conclusion is when an accident is detected an alert message is sent to nearby control rooms using the GSM module |  |
| Project work | Students collaborately completed the project | PO9 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 | PSO1 | PSO2 | PSO3 |
| Mapping |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Guide/Supervisor Signature



**ABSTRACT**

In this fast-paced world, the number of deaths due to accident is growing at an expeditious rate. Major reasons for these accidents are tash driving drowsiness, drunken driving, carelessness, etc. An indicator of survival rates after detecting accidents is the time between the occurrence of accidents and the advent of medical care to the victim. The rapid growth of technology has made everything more facile and this advancement in technology additionally increased accidents. Due to this delayed medical attention, the accident victims might die as well. As a solution to these problems, we introduce a system that detects road accidents and will provide an alert message to the most primate control room immediately. The camera module of the system is deployed in accident-prone areas. Whenever an accident occurs, it will detect the accident and immediately report about it to the nearby control room. The working of the system is based on deep learning techniques that use comolutional neural networks. By utilizing this system, many people can be saved from death.

**INBOX**

**CONTENT PAGE NUMBER**

1.INTRODUCTION **1**

* 1.1. Structure of Project **2**

2.LITERATURE SURVEY **3-4**

3.PROBLEM ANALYSIS **4**

* 3.1. Existing system **5**
* 3.2. Proposed system **5**
* 3.3. Software & Hardware Requirements **6**
* 3.4. Algorithm **7-8**
* 3.5. Problem statement **9**

4. IMPLEMENTATION **10-37**

* 4.1. Code **38-45**

5.UML DIAGRAMS

* 5.1. Use Case Diagram **46**
* 5.2. Class Diagram **46-47**
* 5.3. Sequence Diagram **47**
* 5.4. Activity Diagram **48**

6.TESTING **49-51**

7.RESULTS **52-56**

8.CONCLUSION **57**

9.REFERENCES **58**

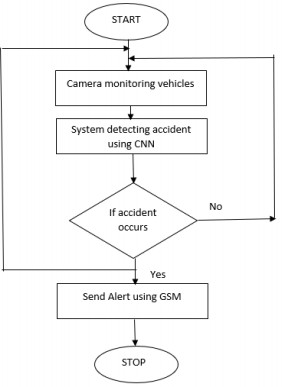
# ACCIDENT DETECTION SYSTEM BY USING CNN

## 1.INTRODUCTION

Road accidents in India are a major cause of decreasing life expectancy with road accidents contributing to over 148,000 deaths out of 467,000 deaths in 2016. Indian Economy has a hit of 3 percent of GDP growth due to road accidents as per the United Nations with an estimated loss of $58,000 in terms of value every year. The metropolitan cities such as Chennai, Mumbai and New Delhi have been increasingly highlighted for lack of road safety and rash driving cases. The recent trends show that there has been an increase in the global number of road accidents even in developed countries. However, under-developed and developing countries suffer a more significant impact due to life and economic losses. These accidents occur due to violation of traffic safety rules, careless rash driving, driver drowsiness and lack of good quality roads. The problem becomes more adverse for highways and hilly areas where accidents are unavoidable. Road accidents are characterized by high death rates due to delay in arrival of help and inefficient systems of mitigation to alert the concerned authorities. Road accidents on the highways are typically caused by natural reasons such as extreme weather conditions such as fog and consecutive collision of vehicles are common on Indian highways due to lack of visibility. The states of Maharashtra, Tamil Nadu and Uttar Pradesh account for the highest number of road accidents in India.The problem can be handled by making use of computer vision and low-cost sensor networks. The current solutions involved heavy dependency on sensor networks and area coverage. This can be substantially replaced by making use of object detection and image segmentation for accident classification. The system identifies the accident-prone areas which are the target stakeholders for the deployment and sets it apart from other implementations since it provides a feasibility factor associated with it. Furthermore, the system provides enhanced mitigation alert to the concerned authorities which helps in preventing any consecutive collisions that could possibly lead to greater loss of lives .

## 

## 1.1 Structure of Project



### 2 .LITERATURE SURVEY

### Various literature papers were studied and analyzed to understand their work and techniques. Thus, we studied the demerits and merits of various ideas related to accident detection. In one paper , two phases are used, an accident detection phase and an accident prevention phase. The authors have mentioned that they used IR sensors and Arduino Uno technology. But in , they do not provide an accurate result and also the sensors are costly. In another paper , an accident detection system using Inertial Measurement Unit (IMU) and 3G cellular module using an accident detection method, but the IMU suffer accumulated error and it is complex. Another technology discussed in the paper was a vision-based accident detection system for detecting, recording, and reporting accidents at intersections. They extract features of moving vehicles using cameras in order to detect accidents. In , the results may be wrong when vehicles move fast. There is a method in which accidents are detected using a set of speed sensors spatially located in a street that can communicate using a particular network . Also, there is a smart phone-based accident detection system, in which data is continuously collected from smartphone’s accelerometer and analyzed using Dynamic Time Warping (DTW) . In another paper , Accident Detection and Reporting System (ADRS) is placed inside the vehicle that uses a sensor to detect accidents. The sensor output is monitored using a microcontroller. In a paper , two methods for detecting and reducing accidents were discussed, one was to use a Smart Helmet in which the mechanism automatically checks whether the rider is wearing a helmet, and another system is used to detect accidents and report it using GSM module. But this system mainly focuses on two-wheeler accidents.

### Machine Learning Classifiers:

These are used to predict the class/target/labels/categories of a given data points. Classification belongs to the category of supervised learning in which the targets are provided with input data. They are used in many applications like medical diagnosis, spam detection, target marketing etc. They use a mapping function (f) from input variables (X) to discrete output variables(Y).

### Opencv:

OpenCV is an open-source library which is primarily used for Computer Vision Applications. This contains many functions and algorithms for Motion tracking, Facial recognition, Object Detection, Segmentation and recognition and many other applications. Images and real time video streams can be manipulated to suit different needs using this library.

**Tensorflow:**

It is an open-source machine learning framework to build and train neural networks. It has a collection of tools, libraries and community resources which helps in easy building of deployment of ML powered applications. This is developed and maintained by Google and was released in 2015.

# 3.PROBLEM ANALYSIS

## EXISTING SYSTEM

Two phases are used, an accident detection phase and an accident prevention phase. The authors have mentioned that they used IR sensors and Arduino Uno technology. But in they do not provide an accurate result and also the sensors are costly. In another paper an accident detection system using Inertial Measurement Unit (IMU) and 3G cellular module using an accident detection method, but the IMU suffer accumulated error and it is complex.

## Disadvantages

* + - * The rapid growth of technology has made everything more facile and this advancement in technology additionally increased accidents. Due to this delayed medical attention, the accident victims might die as well.

## PROPOSED SYSTEM

Since there was no dataset available, a dataset was created that includes accident and non- accident images. If an accident occurs, an alert message will be sent to the nearby control unit. We trained the system with the created dataset. The trained system is then incorporated with the cameras so as to capture the video of the vehicles on the road. By calculating the probability, the system predicts whether an accident happened or not. In case of an accident, an alert is sent to the control rooms using the GSM module. Fig. 1 is the flowchart depicting working of the system. The camera module records the video of vehicles in the road. The camera is placed at fixed locations, mostly in accident-prone areas. Whenever an accident occurs, it is predicted using our deep learning model and followed by sending alert message to the nearby control rooms.

## Advantages:

* + - * The working of the system is based on deep learning techniques that use convolutional neural networks. By utilizing this system, many people can be saved from death.

## Software Requirements

For developing the application the following are the Software Requirements:

1. Python
2. Django
3. Mysql
4. Wampserver

### Operating Systems supported

1. Windows 7
2. Windows XP
3. Windows 8

### Technologies and Languages used to Develop

1. Python

### Debugger and Emulator

* Any Browser (Particularly Chrome)

### Hardware Requirements

For developing the application the following are the Hardware Requirements:

* Processor: Pentium IV or higher
* RAM: 256 MB
* Space on Hard Disk: minimum 512MB

## ALGORITHM

Algorithm Used : Convolutional nueral network. There are two main parts to a CNN architecture

* A convolution tool that separates and identifies the various features of the image for analysis in a process called as Feature Extraction
* A fully connected layer that utilizes the output from the convolution process and predicts the class of the image based on the features extracted in previous stages.

## Convolution Layers

There are three types of layers that make up the CNN which are the convolutional layers, pooling layers, and fully-connected (FC) layers. When these layers are stacked, a CNN architecture will be formed. In addition to these three layers, there are two more important parameters which are the dropout layer and the activation function which are defined below.

## Convolutional Layer

This layer is the first layer that is used to extract the various features from the input images. In this layer, the mathematical operation of convolution is performed between the input image and a filter of a particular size MxM. By sliding the filter over the input image, the dot product is taken between the filter and the parts of the input image with respect to the size of the filter (MxM).

The output is termed as the Feature map which gives us information about the image such as the corners and edges. Later, this feature map is fed to other layers to learn several other features of the input image.

## Pooling Layer

In most cases, a Convolutional Layer is followed by a Pooling Layer. The primary aim of this layer is to decrease the size of the convolved feature map to reduce the computational costs. This is performed by decreasing the connections between layers and independently operates on each feature map. Depending upon method used, there are several types of Pooling operations.

In Max Pooling, the largest element is taken from feature map. Average Pooling calculates the average of the elements in a predefined sized Image section. The total sum of the elements in the predefined section is computed in Sum Pooling. The Pooling Layer usually serves as a bridge between the Convolutional Layer and the FC Layer

## Fully Connected Layer

The Fully Connected (FC) layer consists of the weights and biases along with the neurons and is used to connect the neurons between two different layers. These layers are usually placed before the output layer and form the last few layers of a CNN Architecture.

In this, the input image from the previous layers are flattened and fed to the FC layer. The flattened vector then undergoes few more FC layers where the mathematical functions operations usually take place. In this stage, the classification process begins to take place.

## Dropout

Usually, when all the features are connected to the FC layer, it can cause overfitting in the training dataset. Overfitting occurs when a particular model works so well on the training data causing a negative impact in the model’s performance when used on a new data.

To overcome this problem, a dropout layer is utilised wherein a few neurons are dropped from the neural network during training process resulting in reduced size of the model. On passing a dropout of 0.3, 30% of the nodes are dropped out randomly from the neural network.

## Activation Functions

Finally, one of the most important parameters of the CNN model is the activation function. They are used to learn and approximate any kind of continuous and complex relationship between variables of the network. In simple words, it decides which information of the model should fire in the forward direction and which ones should not at the end of the network.

It adds non-linearity to the network. There are several commonly used activation functions such as the ReLU, Softmax, tanH and the Sigmoid functions. Each of these functions have a specific usage. For a binary classification CNN model, sigmoid and softmax functions are preferred an for a multi-class classification, generally softmax us used.

**3.5** . **PROBLEM STATEMENT**

Traditional traffic monitoring system in designed only to monitor traffic or to control the traffic, but it does not provide any solution to decrease the fatal accidental human damages rate which occur due to lack of medical aid in real time. Consider a scenario where an accident occurred but no one was there to report this accident, the victim is critical and every second counts, any delay can result in disability or death. We cannot root out accidents totally but we can improve in providing post accidental care just-in-time. There are lots of sensor based systems available in the market as well but that require vehicle owners to install those sensors in their vehicles. The working of these systems is based on any damage being sensed by the sensors installed; these signals from the sensors will trigger a system that will alert nearby medical assistance or an emergency contact number. But what if the accident happened of a vehicle which is not equipped with such sensor based system. We need an advance Artificial intelligence based surveillance system which not only can detect occurrence of accident but also can alert to nearby hospitals/ambulance or Traffic policemen in real-time. Our system is based on Neural Network and Deep Learning of object detection along computer vision technology and several methods and algorithms. Our approach will work on still images, recorded-videos, real-time live videos and will detect, classify, track and compute moving object velocity and direction using convolution neural network.

**Applications:**

Our system is based on Neural Network and Deep Learning of object detection along computer vision technology and several methods and algorithms. Our approach will work on still images, recorded-videos, real-time live videos and will detect, classify, track and compute moving object velocity and direction using convolution neural network.

**Limitations:**

Cost is high.

# 4. IMPLEMENTATION ON (PYTHON):

**Python** is a general purpose, dynamic, high level, and interpreted programming language. It supports Object Oriented programming approach to develop applications. It is simple and easy to learn and provides lots of high-level data structures.

Python is easy to learn yet powerful and versatile scripting language, which makes it attractive for Application Development.

Python's syntax and dynamic typing with its interpreted nature make it an ideal language for scripting and rapid application development.

Python supports multiple programming pattern, including object-oriented, imperative, and functional or procedural programming styles.

Python is not intended to work in a particular area, such as web programming. That is why it is known as multipurpose programming language because it can be used with web, enterprise, 3D CAD, etc.

We don't need to use data types to declare variable because it is *dynamically typed* so we can write a=10 to assign an integer value in an integer variable.

Python makes the development and debugging *fast* because there is no compilation step included in Python development , and edit-test-debug cycle is very fast.

**Python 2 vs. Python 3**

In most of the programming languages, whenever a new version releases, it supports the features and syntax of the existing version of the language, therefore, it is easier for the projects to switch in the newer version. However, in the case of Python, the two versions Python 2 and Python 3 are very much different from each other.

A list of differences between Python 2 and Python 3 are given below:

1. Python 2 uses **print** as a statement and used as print "something" to print some string on the console. On the other hand, Python 3 uses **print** as a function and used as print("something") to print something on the console.
2. Python 2 uses the function raw\_input() to accept the user's input. It returns the string representing the value, which is typed by the user. To convert it into the integer, we need to use the int() function in Python. On the other hand, Python 3 uses input() function which automatically interpreted the type of input entered by the user. However, we can cast this value to any type by using primitive functions (int(), str(), etc.).
3. In Python 2, the implicit string type is ASCII, whereas, in Python 3, the implicit string type is Unicode.
4. Python 3 doesn't contain the xrange() function of Python 2. The xrange() is the variant of range() function which returns a xrange object that works similar to Java iterator. The range() returns a list for example the function range(0,3) contains 0, 1, 2.
5. There is also a small change made in Exception handling in Python 3. It defines a keyword **as** which is necessary to be used. We will discuss it in Exception handling section of Python programming tutorial.

# Python Applications

Python is known for its general purpose nature that makes it applicable in almost each domain of software development. Python as a whole can be used in any sphere of development.

Here, we are specifing applications areas where python can be applied.

#### 1) Web Applications

We can use Python to develop web applications. It provides libraries to handle internet protocols such as HTML and XML, JSON, Email processing, request, beautifulSoup , Feed parser etc. It also provides Frameworks such as Django, Pyramid, Flask etc to design and delelop web based applications. Some important developments are: PythonWikiEngines, Pocoo, PythonBlogSoftware etc.

#### 2) Desktop GUI Applications

Python provides Tk GUI library to develop user interface in python based application. Some other useful toolkits wxWidgets, Kivy, pyqt that are useable on several platforms. The Kivy is popular for writing multitouch applications.

#### 3) Software Development

Python is helpful for software development process. It works as a support language and can be used for build control and management, testing etc.

#### 4) Scientific and Numeric

Python is popular and widely used in scientific and numeric computing. Some useful library and package are SciPy, Pandas, IPython etc. SciPy is group of packages of engineering, science and mathematics.

#### 5) Business Applications

Python is used to build Bussiness applications like ERP and e-commerce systems. Tryton is a high level application platform.

#### 6) Console Based Application

We can use Python to develop console based applications. For example: **IPython**.

#### 7) Audio or Video based Applications

Python is awesome to perform multiple tasks and can be used to develop multimedia applications. Some of real applications are: TimPlayer, cplay etc.

#### 8) 3D CAD Applications

To create CAD application Fandango is a real application which provides full features of CAD.

#### 9) Enterprise Applications

Python can be used to create applications which can be used within an Enterprise or an Organization. Some real time applications are: OpenErp, Tryton, Picalo etc.

#### 10) Applications for Images

Using Python several application can be developed for image. Applications developed are: VPython, Gogh, imgSeek etc.

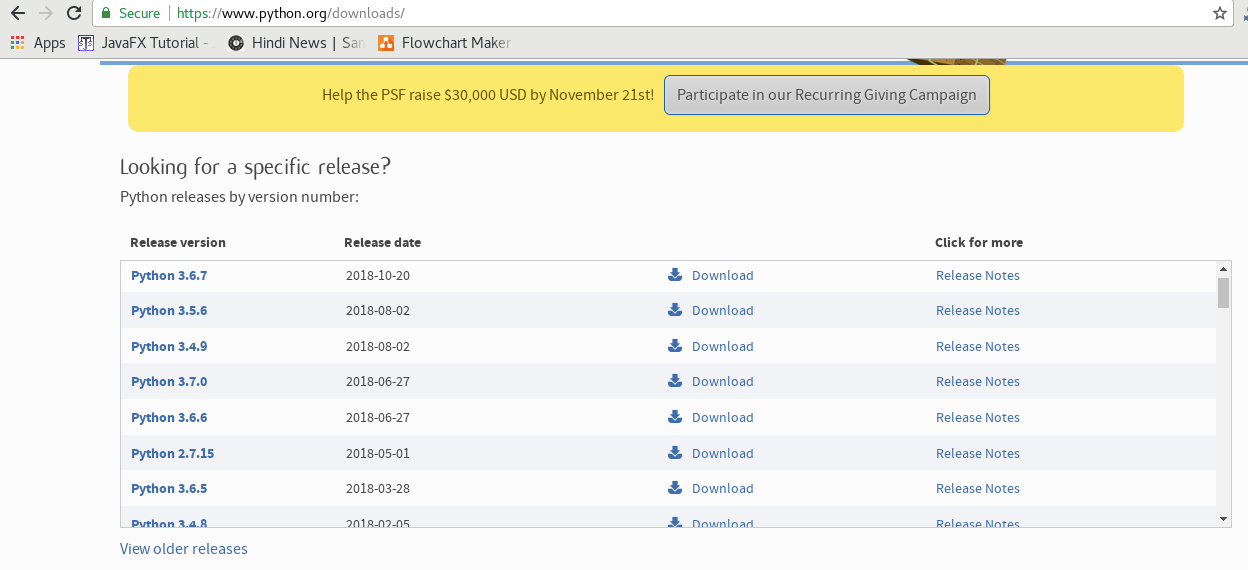
There are several such applications which can be developed using Python

# How to Install Python (Environment Set-up)

In this section of the tutorial, we will discuss the installation of python on various operating systems.

## Installation on Windows

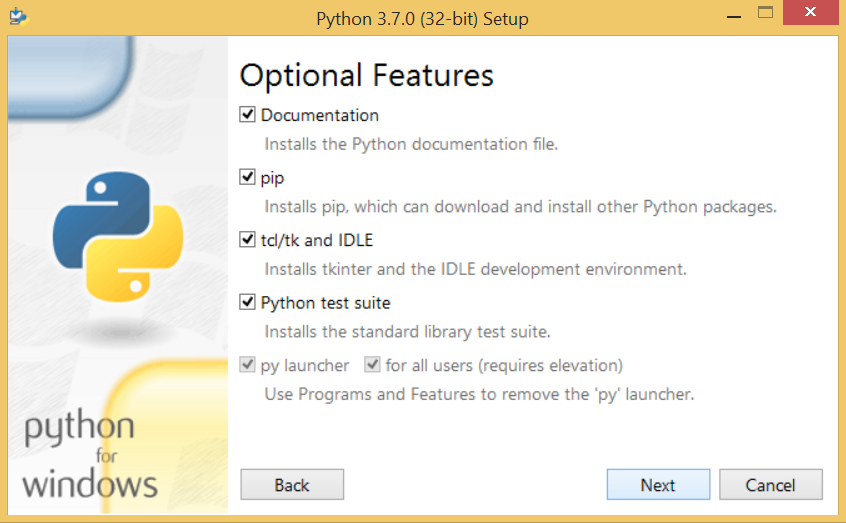
Visit the link <https://www.python.org/downloads/> to download the latest release of Python. In this process, we will install Python 3.6.7 on our Windows operating system.



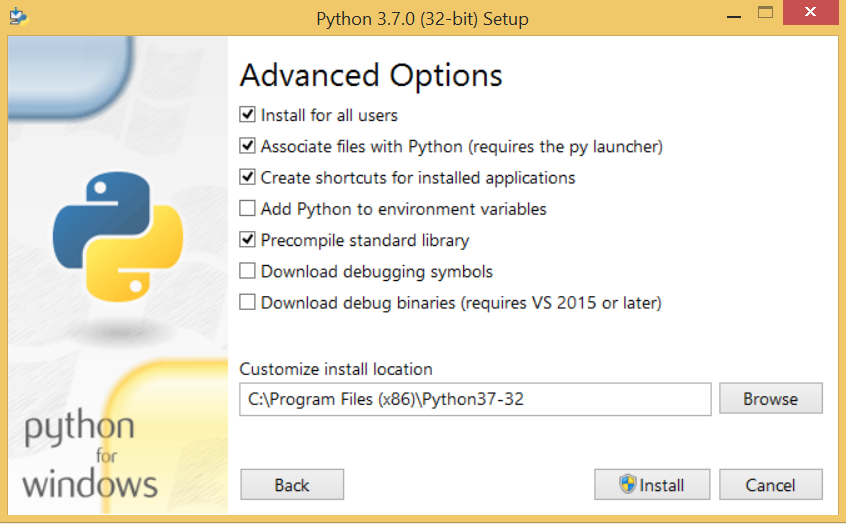
Double-click the executable file which is downloaded; the following window will open. Select Customize installation and proceed.



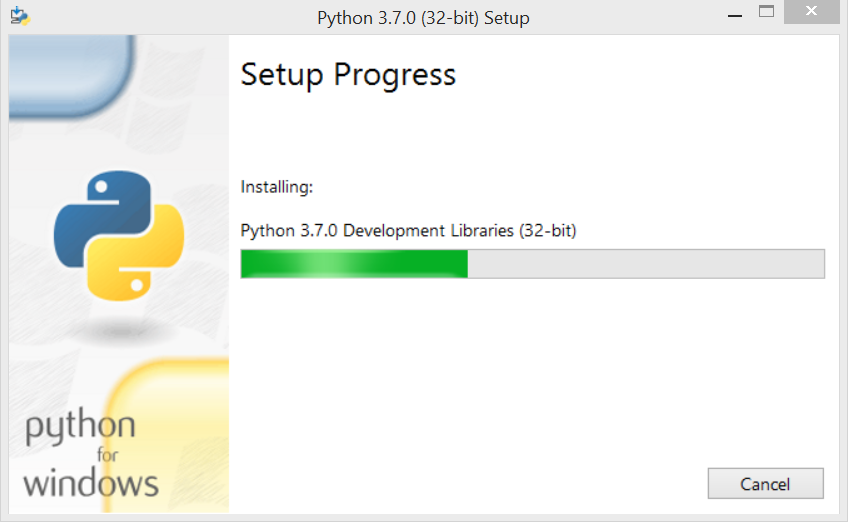
The following window shows all the optional features. All the features need to be installed and are checked by default; we need to click next to continue.



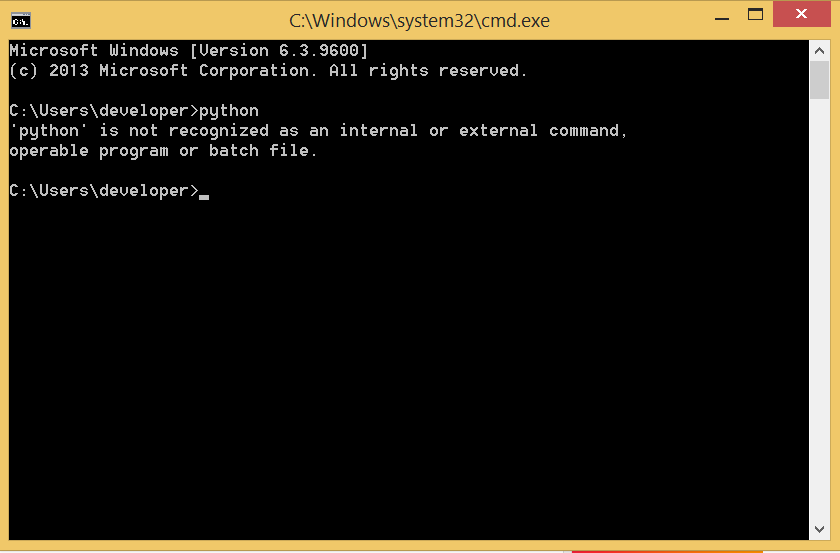
The following window shows a list of advanced options. Check all the options which you want to install and click next. Here, we must notice that the first check-box (install for all users) must be checked.



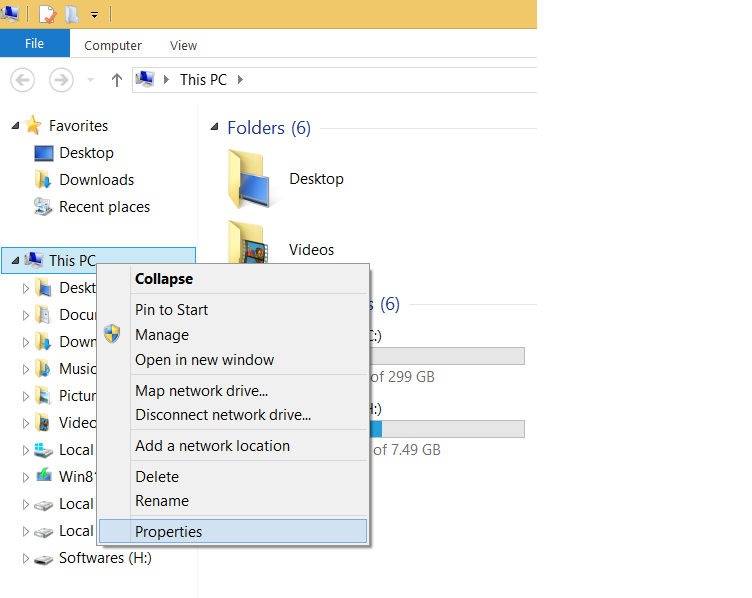
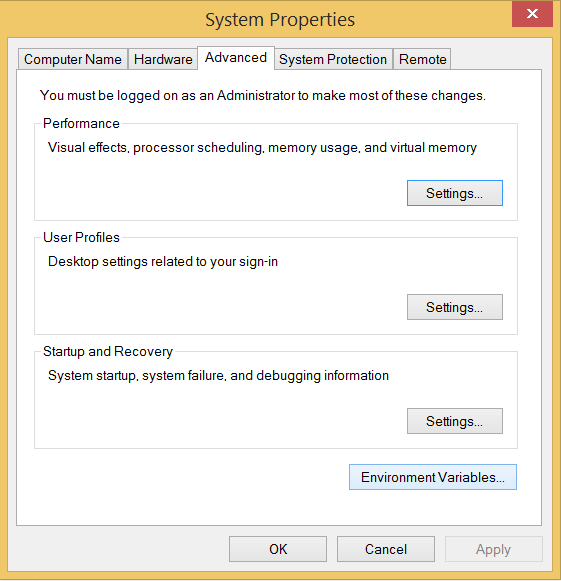
Now, we are ready to install python-3.6.7. Let's install it.



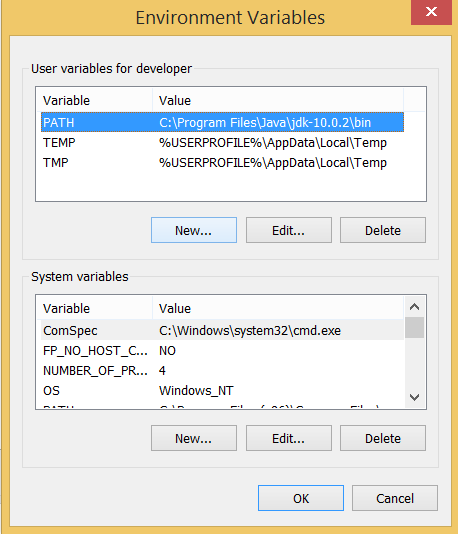
Now, try to run python on the command prompt. Type the command **python** in case of python2 or python3 in case of **python3**. It will show an error as given in the below image. It is because we haven't set the path.



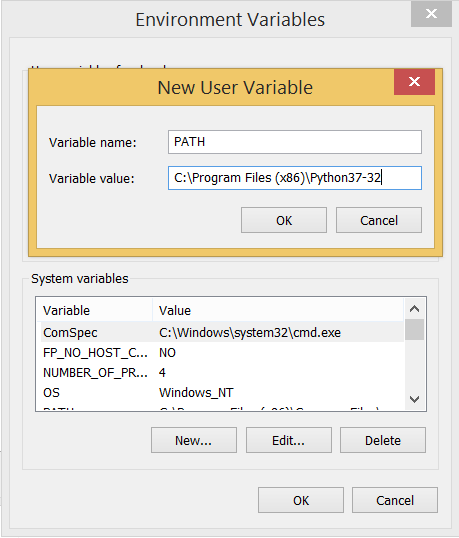
To set the path of python, we need to the right click on "my computer" and go to Properties → Advanced → Environment Variables.

Add the new path variable in the user variable section.



Type **PATH** as the variable name and set the path to the installation directory of the python shown in the below image.



Now, the path is set, we are ready to run python on our local system. Restart CMD, and type **python** again. It will open the python interpreter shell where we can execute the python statements.

**What Is A Script?**

Up to this point, I have concentrated on the interactive programming capability of Python.  This is a very useful capability that allows you to type in a program and to have it executed immediately in an interactive mode

**Scripts are reusable:**

Basically, a script is a text file containing the statements that comprise a Python program.  Once you have created the script, you can execute it over and over without having to retype it each time.

**Scripts are editable:**

Perhaps, more importantly, you can make  different versions of the script by modifying the statements from one file to the next using a text editor.  Then you can execute each of the individual versions.  In this way, it is easy to create different programs with a minimum amount of typing.

**You will need a text editor:**

Just about any text editor will suffice for creating Python script files.

You can use Microsoft Notepad, Microsoft WordPad, Microsoft Word, or just about any word processor if you want to.

**Difference between a script and a program**

**Script:** Scripts are distinct from the core code of the application, which is usually written in a different language, and are often created or at least modified by the end-user. Scripts are often interpreted from source code or byte code, where as the applications they control are traditionally compiled to native machine code.

**Program:** The program has an executable form that the computer can use directly to execute the instructions.

The same program in its human-readable source code form, from which executable programs are derived(e.g., compiled)

**Python**

what is Python? Chances you are asking yourself this. You may have found this book because you want to learn to program but don’t know anything about programming languages. Or you may have heard of programming languages like C, C++, C#, or Java and want to know what Python is and how it compares to “big name” languages. Hopefully I can explain it for you.

**Python concepts**

If you not interested in the how and whys of Python, feel free to skip to the next chapter. In this chapter I will try to explain to the reader why I think Python is one of the best languages available and why it’s a great one to start programming with.

• Open source general-purpose language.

• Object Oriented, Procedural, Functional

• Easy to interface with C/ObjC/Java/Fortran

• Easy-ish to interface with C++ (via SWIG)

• Great interactive environment

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

* **Python is Interpreted** − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
* **Python is Interactive** − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
* **Python is Object-Oriented** − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
* **Python is a Beginner's Language** − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

**History of Python**

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

**Python Features**

Python's features include −

* **Easy-to-learn** − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
* **Easy-to-read** − Python code is more clearly defined and visible to the eyes.
* **Easy-to-maintain** − Python's source code is fairly easy-to-maintain.
* **A broad standard library** − Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
* **Interactive Mode** − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
* **Portable** − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
* **Extendable** − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
* **Databases** − Python provides interfaces to all major commercial databases.
* **GUI Programming** − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
* **Scalable** − Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below −

* It supports functional and structured programming methods as well as OOP.
* It can be used as a scripting language or can be compiled to byte-code for building large applications.
* It provides very high-level dynamic data types and supports dynamic type checking.
* IT supports automatic garbage collection.
* It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

**Dynamic vs Static**

Types Python is a dynamic-typed language. Many other languages are static typed, such as C/C++ and Java. A static typed language requires the programmer to explicitly tell the computer what type of “thing” each data value is.

For example, in C if you had a variable that was to contain the price of something, you would have to declare the variable as a “float” type.

This tells the compiler that the only data that can be used for that variable must be a floating point number, i.e. a number with a decimal point.

If any other data value was assigned to that variable, the compiler would give an error when trying to compile the program.

Python, however, doesn’t require this. You simply give your variables names and assign values to them. The interpreter takes care of keeping track of what kinds of objects your program is using. This also means that you can change the size of the values as you develop the program. Say you have another decimal number (a.k.a. a floating point number) you need in your program.

With a static typed language, you have to decide the memory size the variable can take when you first initialize that variable. A double is a floating point value that can handle a much larger number than a normal float (the actual memory sizes depend on the operating environment).

If you declare a variable to be a float but later on assign a value that is too big to it, your program will fail; you will have to go back and change that variable to be a double.

With Python, it doesn’t matter. You simply give it whatever number you want and Python will take care of manipulating it as needed. It even works for derived values.

For example, say you are dividing two numbers. One is a floating point number and one is an integer. Python realizes that it’s more accurate to keep track of decimals so it automatically calculates the result as a floating point number

**Variables**

Variables are nothing but reserved memory locations to store values. This means that when you create a variable you reserve some space in memory.

Based on the data type of a variable, the interpreter allocates memory and decides what can be stored in the reserved memory. Therefore, by assigning different data types to variables, you can store integers, decimals or characters in these variables.

**Standard Data Types**

The data stored in memory can be of many types. For example, a person's age is stored as a numeric value and his or her address is stored as alphanumeric characters. Python has various standard data types that are used to define the operations possible on them and the storage method for each of them.

Python has five standard data types −

* Numbers
* String
* List
* Tuple
* Dictionary

## Python Numbers

Number data types store numeric values. Number objects are created when you assign a value to them

## Python Strings

Strings in Python are identified as a contiguous set of characters represented in the quotation marks. Python allows for either pairs of single or double quotes. Subsets of strings can be taken using the slice operator ([ ] and [:] ) with indexes starting at 0 in the beginning of the string and working their way from -1 at the end.

## Python Lists

Lists are the most versatile of Python's compound data types. A list contains items separated by commas and enclosed within square brackets ([]). To some extent, lists are similar to arrays in C. One difference between them is that all the items belonging to a list can be of different data type.

The values stored in a list can be accessed using the slice operator ([ ] and [:]) with indexes starting at 0 in the beginning of the list and working their way to end -1. The plus (+) sign is the list concatenation operator, and the asterisk (\*) is the repetition operator.

**Python Tuples**

A tuple is another sequence data type that is similar to the list. A tuple consists of a number of values separated by commas. Unlike lists, however, tuples are enclosed within parentheses.

The main differences between lists and tuples are: Lists are enclosed in brackets ( [ ] ) and their elements and size can be changed, while tuples are enclosed in parentheses ( ( ) ) and cannot be updated. Tuples can be thought of as **read-only** lists.

## Python Dictionary

Python's dictionaries are kind of hash table type. They work like associative arrays or hashes found in Perl and consist of key-value pairs. A dictionary key can be almost any Python type, but are usually numbers or strings. Values, on the other hand, can be any arbitrary Python object.

Dictionaries are enclosed by curly braces ({ }) and values can be assigned and accessed using square braces ([]).

**Different modes in python**

Python has two basic modes: normal and interactive.

The normal mode is the mode where the scripted and finished .py files are run in the Python interpreter.

Interactive mode is a command line shell which gives immediate feedback for each statement, while running previously fed statements in active memory. As new lines are fed into the interpreter, the fed program is evaluated both in part and in whole

# Python Functions

Functions are the most important aspect of an application. A function can be defined as the organized block of reusable code which can be called whenever required.

Python allows us to divide a large program into the basic building blocks known as function. The function contains the set of programming statements enclosed by {}. A function can be called multiple times to provide reusability and modularity to the python program.

In other words, we can say that the collection of functions creates a program. The function is also known as procedure or subroutine in other programming languages.

Python provide us various inbuilt functions like range() or print(). Although, the user can create its functions which can be called user-defined functions.

## Advantage of functions in python

There are the following advantages of C functions.

* By using functions, we can avoid rewriting same logic/code again and again in a program.
* We can call python functions any number of times in a program and from any place in a program.
* We can track a large python program easily when it is divided into multiple functions.
* Reusability is the main achievement of python functions.
* However, Function calling is always overhead in a python program.

## Creating a function

In python, we can use **def** keyword to define the function. The syntax to define a function in python is given below.

1. **def** my\_function():
2. function-suite
3. **return** <expression>

The function block is started with the colon (:) and all the same level block statements remain at the same indentation.

A function can accept any number of parameters that must be the same in the definition and function calling.

## Function calling

In python, a function must be defined before the function calling otherwise the python interpreter gives an error. Once the function is defined, we can call it from another function or the python prompt. To call the function, use the function name followed by the parentheses.

A simple function that prints the message "Hello Word" is given below.

1. **def** hello\_world():
2. **print**("hello world")
4. hello\_world()

**Output:**

hello world

## Parameters in function

The information into the functions can be passed as the parameters. The parameters are specified in the parentheses. We can give any number of parameters, but we have to separate them with a comma.

Consider the following example which contains a function that accepts a string as the parameter and prints it.

## Example 1

1. #defining the function
2. **def** func (name):
3. **print**("Hi ",name);
5. #calling the function
6. func("Ayush")

## Example 2

1. #python function to calculate the sum of two variables
2. #defining the function
3. **def** sum (a,b):
4. **return** a+b;
6. #taking values from the user
7. a = int(input("Enter a: "))
8. b = int(input("Enter b: "))
10. #printing the sum of a and b
11. **print**("Sum = ",sum(a,b))

**Output:**

Enter a: 10

Enter b: 20

Sum = 30

## Call by reference in Python

In python, all the functions are called by reference, i.e., all the changes made to the reference inside the function revert back to the original value referred by the reference.

However, there is an exception in the case of mutable objects since the changes made to the mutable objects like string do not revert to the original string rather, a new string object is made, and therefore the two different objects are printed.

## Example 1 Passing Immutable Object (List)

1. #defining the function
2. **def** change\_list(list1):
3. list1.append(20);
4. list1.append(30);
5. **print**("list inside function = ",list1)
7. #defining the list
8. list1 = [10,30,40,50]
10. #calling the function
11. change\_list(list1);
12. **print**("list outside function = ",list1);

**Output:**

list inside function = [10, 30, 40, 50, 20, 30]

list outside function = [10, 30, 40, 50, 20, 30]

## Example 2 Passing Mutable Object (String)

1. #defining the function
2. **def** change\_string (str):
3. str = str + " Hows you";
4. **print**("printing the string inside function :",str);
6. string1 = "Hi I am there"
8. #calling the function
9. change\_string(string1)
11. **print**("printing the string outside function :",string1)

**Output:**

printing the string inside function : Hi I am there Hows you

printing the string outside function : Hi I am there

## Types of arguments

There may be several types of arguments which can be passed at the time of function calling.

1. Required arguments
2. Keyword arguments
3. Default arguments
4. Variable-length arguments

## Required Arguments

Till now, we have learned about function calling in python. However, we can provide the arguments at the time of function calling. As far as the required arguments are concerned, these are the arguments which are required to be passed at the time of function calling with the exact match of their positions in the function call and function definition. If either of the arguments is not provided in the function call, or the position of the arguments is changed, then the python interpreter will show the error.

Consider the following example.

## Example 1

1. #the argument name is the required argument to the function func
2. **def** func(name):
3. message = "Hi "+name;
4. **return** message;
5. name = input("Enter the name?")
6. **print**(func(name))

**Output:**

Enter the name?John

Hi John

## Example 2

1. #the function simple\_interest accepts three arguments and returns the simple interest accordingly
2. **def** simple\_interest(p,t,r):
3. **return** (p\*t\*r)/100
4. p = float(input("Enter the principle amount? "))
5. r = float(input("Enter the rate of interest? "))
6. t = float(input("Enter the time in years? "))
7. **print**("Simple Interest: ",simple\_interest(p,r,t))

**Output:**

Enter the principle amount? 10000

Enter the rate of interest? 5

Enter the time in years? 2

Simple Interest: 1000.0

## Example 3

1. #the function calculate returns the sum of two arguments a and b
2. **def** calculate(a,b):
3. **return** a+b
4. calculate(10) # this causes an error as we are missing a required arguments b.

**Output:**

TypeError: calculate() missing 1 required positional argument: 'b'

## Keyword arguments

Python allows us to call the function with the keyword arguments. This kind of function call will enable us to pass the arguments in the random order.

The name of the arguments is treated as the keywords and matched in the function calling and definition. If the same match is found, the values of the arguments are copied in the function definition.

Consider the following example.

## Example 1

1. #function func is called with the name and message as the keyword arguments
2. **def** func(name,message):
3. **print**("printing the message with",name,"and ",message)
4. func(name = "John",message="hello") #name and message is copied with the values John and hello respectively

**Output:**

printing the message with John and hello

# 20 Python libraries

**1.** Requests. The most famous http library written by kennethreitz. It’s a must have for every python developer.

**2.** Scrapy. If you are involved in webscraping then this is a must have library for you. After using this library you won’t use any other.

**3.** wxPython. A gui toolkit for python. I have primarily used it in place of tkinter. You will really love it.

**4.** Pillow. A friendly fork of PIL (Python Imaging Library). It is more user friendly than PIL and is a must have for anyone who works with images.

**5.** SQLAlchemy. A database library. Many love it and many hate it. The choice is yours.

**6.** BeautifulSoup. I know it’s slow but this xml and html parsing library is very useful for beginners.

**7.** Twisted. The most important tool for any network application developer. It has a very beautiful api and is used by a lot of famous python developers.

**8.** NumPy. How can we leave this very important library ? It provides some advance math functionalities to python.

**9.** SciPy. When we talk about NumPy then we have to talk about scipy. It is a library of algorithms and mathematical tools for python and has caused many scientists to switch from ruby to python.

**10.** matplotlib. A numerical plotting library. It is very useful for any data scientist or any data analyzer.

**11.** Pygame. Which developer does not like to play games and develop them ? This library will help you achieve your goal of 2d game development.

**12.** Pyglet. A 3d animation and game creation engine. This is the engine in which the famous [python port](https://github.com/fogleman/Minecraft) of minecraft was made

**13.** pyQT. A GUI toolkit for python. It is my second choice after wxpython for developing GUI’s for my python scripts.

**14.** pyGtk. Another python GUI library. It is the same library in which the famous Bittorrent client is created.

**15.** Scapy. A packet sniffer and analyzer for python made in python.

**16.** pywin32. A python library which provides some useful methods and classes for interacting with windows.

**17.** nltk. Natural Language Toolkit – I realize most people won’t be using this one, but it’s generic enough. It is a very useful library if you want to manipulate strings. But it’s capacity is beyond that. Do check it out.

**18.** nose. A testing framework for python. It is used by millions of python developers. It is a must have if you do test driven development.

**19.** SymPy. SymPy can do algebraic evaluation, differentiation, expansion, complex numbers, etc. It is contained in a pure Python distribution.

**20.** IPython. I just can’t stress enough how useful this tool is. It is a python prompt on steroids. It has completion, history, shell capabilities, and a lot more. Make sure that you take a look at it.

**Numpy**

NumPy’s main object is the homogeneous multidimensional array. It is a table of elements (usually numbers), all of the same type, indexed by a tuple of positive integers. In NumPy dimensions are called axes. The number of axes is rank.

• Offers Matlab-ish capabilities within Python

• Fast array operations

• 2D arrays, multi-D arrays, linear algebra etc.

**Matplotlib**

• High quality plotting library.

**Python class and objects**

These are the building blocks of OOP class creates a new object. This object can be anything, whether an abstract data concept or a model of a physical object, e.g. a chair. Each class has individual characteristics unique to that class, including variables and methods. Classes are very powerful and currently “the big thing” in most programming languages. Hence, there are several chapters dedicated to OOP later in the book.

The class is the most basic component of object-oriented programming. Previously, you learned how to use functions to make your program do something.

Now will move into the big, scary world of Object-Oriented Programming (OOP). To be honest, it took me several months to get a handle on objects.

When I first learned C and C++, I did great; functions just made sense for me.

Having messed around with BASIC in the early ’90s, I realized functions were just like subroutines so there wasn’t much new to learn.

However, when my C++ course started talking about objects, classes, and all the new features of OOP, my grades definitely suffered.

Once you learn OOP, you’ll realize that it’s actually a pretty powerful tool. Plus many Python libraries and APIs use classes, so you should at least be able to understand what the code is doing.

One thing to note about Python and OOP: it’s not mandatory to use objects in your code in a way that works best; maybe you don’t need to have a full-blown class with initialization code and methods to just return a calculation. With Python, you can get as technical as you want.

As you’ve already seen, Python can do just fine with functions. Unlike languages such as Java, you aren’t tied down to a single way of doing things; you can mix functions and classes as necessary in the same program. This lets you build the code

Objects are an encapsulation of variables and functions into a single entity. Objects get their variables and functions from classes. Classes are essentially a template to create your objects.

Here’s a brief list of Python OOP ideas:

• The class statement creates a class object and gives it a name. This creates a new namespace.

• Assignments within the class create class attributes. These attributes are accessed by qualifying the name using dot syntax: ClassName.Attribute.

• Class attributes export the state of an object and its associated behavior. These attributes are shared by all instances of a class.

• Calling a class (just like a function) creates a new instance of the class.

This is where the multiple copies part comes in.

• Each instance gets ("inherits") the default class attributes and gets its own namespace. This prevents instance objects from overlapping and confusing the program.

• Using the term self identifies a particular instance, allowing for per-instance attributes. This allows items such as variables to be associated with a particular instance.

**Inheritance**

First off, classes allow you to modify a program without really making changes to it.

To elaborate, by subclassing a class, you can change the behavior of the program by simply adding new components to it rather than rewriting the existing components.

As we’ve seen, an instance of a class inherits the attributes of that class.

However, classes can also inherit attributes from other classes. Hence, a subclass inherits from a superclass allowing you to make a generic superclass that is specialized via subclasses.

The subclasses can override the logic in a superclass, allowing you to change the behavior of your classes without changing the superclass at all.

Operator Overloads

Operator overloading simply means that objects that you create from classes can respond to actions (operations) that are already defined within Python, such as addition, slicing, printing, etc.

Even though these actions can be implemented via class methods, using overloading ties the behavior closer to Python’s object model and the object interfaces are more consistent to Python’s built-in objects, hence overloading is easier to learn and use.

User-made classes can override nearly all of Python’s built-in operation methods.

**Exceptions**

I’ve talked about exceptions before but now I will talk about them in depth. Essentially, exceptions are events that modify program’s flow, either intentionally or due to errors.

They are special events that can occur due to an error, e.g. trying to open a file that doesn’t exist, or when the program reaches a marker, such as the completion of a loop.

Exceptions, by definition, don’t occur very often; hence, they are the "exception to the rule" and a special class has been created for them. Exceptions are everywhere in Python.

Virtually every module in the standard Python library uses them, and Python itself will raise them in a lot of different circumstances.

Here are just a few examples:

• Accessing a non−existent dictionary key will raise a KeyError exception.

• Searching a list for a non−existent value will raise a ValueError exception

• Calling a non−existent method will raise an AttributeError exception.

• Referencing a non−existent variable will raise a NameError exception.

• Mixing datatypes without coercion will raise a TypeError exception.

One use of exceptions is to catch a fault and allow the program to continue working; we have seen this before when we talked about files.

This is the most common way to use exceptions. When programming with the Python command line interpreter, you don’t need to worry about catching exceptions.

Your program is usually short enough to not be hurt too much if an exception occurs.

Plus, having the exception occur at the command line is a quick and easy way to tell if your code logic has a problem.

However, if the same error occurred in your real program, it will fail and stop working. Exceptions can be created manually in the code by raising an exception.

It operates exactly as a system-caused exceptions, except that the programmer is doing it on purpose. This can be for a number of reasons. One of the benefits of using exceptions is that, by their nature, they don’t put any overhead on the code processing.

Because exceptions aren’t supposed to happen very often, they aren’t processed until they occur.

Exceptions can be thought of as a special form of the if/elif statements. You can realistically do the same thing with if blocks as you can with exceptions.

However, as already mentioned, exceptions aren’t processed until they occur; if blocks are processed all the time.

Proper use of exceptions can help the performance of your program.

The more infrequent the error might occur, the better off you are to use exceptions; using if blocks requires Python to always test extra conditions before continuing.

Exceptions also make code management easier: if your programming logic is mixed in with error-handling if statements, it can be difficult to read, modify, and debug your program.

User-Defined Exceptions

I won’t spend too much time talking about this, but Python does allow for a programmer to create his own exceptions.

You probably won’t have to do this very often but it’s nice to have the option when necessary.

However, before making your own exceptions, make sure there isn’t one of the built-in exceptions that will work for you.

They have been "tested by fire" over the years and not only work effectively, they have been optimized for performance and are bug-free.

Making your own exceptions involves object-oriented programming, which will be covered in the next chapter

. To make a custom exception, the programmer determines which base exception to use as the class to inherit from, e.g. making an exception for negative numbers or one for imaginary numbers would probably fall under the Arithmetic Error exception class.

To make a custom exception, simply inherit the base exception and define what it will do.

**Python modules**

Python allows us to store our code in files (also called modules). This is very useful for more serious programming, where we do not want to retype a long function definition from the very beginning just to change one mistake. In doing this, we are essentially defining our own modules, just like the modules defined already in the Python library.

To support this, Python has a way to put definitions in a file and use them in a script or in an interactive instance of the interpreter. Such a file is called a module; definitions from a module can be imported into other modules or into the main module.

**Testing code**

As indicated above, code is usually developed in a file using an editor.

To test the code, import it into a Python session and try to run it.

Usually there is an error, so you go back to the file, make a correction, and test again.

This process is repeated until you are satisfied that the code works. T

he entire process is known as the development cycle.

There are two types of errors that you will encounter. Syntax errors occur when the form of some command is invalid.

This happens when you make typing errors such as misspellings, or call something by the wrong name, and for many other reasons. Python will always give an error message for a syntax error.

## Functions in Python

## It is possible, and very useful, to define our own functions in Python. Generally speaking, if you need to do a calculation only once, then use the interpreter. But when you or others have need to perform a certain type of calculation many times, then define a function.

## You use functions in programming to bundle a set of instructions that you want to use repeatedly or that, because of their complexity, are better self-contained in a sub-program and called when needed. That means that a function is a piece of code written to carry out a specified task.

## To carry out that specific task, the function might or might not need multiple inputs. When the task is carred out, the function can or can not return one or more values.There are three types of functions in python:

## help(),min(),print().

## Python Namespace

Generally speaking, a **namespace** (sometimes also called a context) is a naming system for making names unique to avoid ambiguity. Everybody knows a namespacing system from daily life, i.e. the naming of people in firstname and familiy name (surname).

An example is a network: each network device (workstation, server, printer, ...)needs a unique name and address. Yet another example is the directory structure of file systems.

The same file name can be used in different directories, the files can be uniquely accessed via the pathnames.   
Many programming languages use namespaces or contexts for identifiers. An identifier defined in a namespace is associated with that namespace.

This way, the same identifier can be independently defined in multiple namespaces. (Like the same file names in different directories) Programming languages, which support namespaces, may have different rules that determine to which namespace an identifier belongs.

Namespaces in Python are implemented as Python dictionaries, this means it is a mapping from names (keys) to objects (values). The user doesn't have to know this to write a Python program and when using namespaces.

Some namespaces in Python:

* **global names** of a module
* **local names** in a function or method invocation
* **built-in names**: this namespace contains built-in functions (e.g. abs(), cmp(), ...) and built-in exception names

**Garbage Collection**

Garbage Collector exposes the underlying memory management mechanism of Python, the automatic garbage collector. The module includes functions for controlling how the collector operates and to examine the objects known to the system, either pending collection or stuck in reference cycles and unable to be freed.

# 4.1 CODE

from tkinter import messagebox from tkinter import \*

from tkinter import simpledialog import tkinter

from tkinter import filedialog

from tkinter.filedialog import askopenfilename import time

import cv2

import tensorflow as tf

from collections import namedtuple from collections import defaultdict from io import StringIO

from PIL import Image import numpy as np import winsound

main = tkinter.Tk() main.title("Accident Detection") main.geometry("1300x1200")

global filename global detectionGraph

global msg

def loadModel():

global detectionGraph detectionGraph = tf.Graph()

with detectionGraph.as\_default(): od\_graphDef = tf.GraphDef()

with tf.gfile.GFile('model/frozen\_inference\_graph.pb', 'rb') as file: serializedGraph = file.read() od\_graphDef.ParseFromString(serializedGraph) tf.import\_graph\_def(od\_graphDef, name='')

messagebox.showinfo("Training model loaded","Training model loaded")

def beep():

frequency = 2500 # Set Frequency To 2500 Hertz duration = 1000 # Set Duration To 1000 ms == 1 second winsound.Beep(frequency, duration)

def uploadVideo(): global filename

filename = filedialog.askopenfilename(initialdir="videos")

pathlabel.config(text=filename) text.delete('1.0', END) text.insert(END,filename+" loaded\n");

def calculateCollision(boxes,classes,scores,image\_np): global msg

#cv2.putText(image\_np, "NORMAL!", (230, 50), cv2.FONT\_HERSHEY\_SIMPLEX, 1.0,

(255, 255, 255), 2, cv2.LINE\_AA)

for i, b in enumerate(boxes[0]):

if classes[0][i] == 3 or classes[0][i] == 6 or classes[0][i] == 8: if scores[0][i] > 0.5:

for j, c in enumerate(boxes[0]):

if (i != j) and (classes[0][j] == 3 or classes[0][j] == 6 or classes[0][j] == 8) and scores[0][j]> 0.5:

Rectangle = namedtuple('Rectangle', 'xmin ymin xmax ymax')

ra = Rectangle(boxes[0][i][3], boxes[0][i][2], boxes[0][i][1], boxes[0][i][3])

rb = Rectangle(boxes[0][j][3], boxes[0][j][2], boxes[0][j][1], boxes[0][j][3])

ar = rectArea(boxes[0][i][3], boxes[0][i][1],boxes[0][i][2],boxes[0][i][3]) col\_threshold = 0.6\*np.sqrt(ar)

area(ra, rb)

if (area(ra,rb)<col\_threshold) : print('accident')

msg = 'ACCIDENT!'

beep() return True

else:

return False

def rectArea(xmax, ymax, xmin, ymin): x = np.abs(xmax-xmin)

y = np.abs(ymax-ymin) return x\*y

def load\_image\_into\_numpy\_array(image):

(im\_width, im\_height) = image.size

return np.array(image.getdata()).reshape((im\_height, im\_width, 3)).astype(np.uint8)

def area(a, b): # returns None if rectangles don't intersect dx = min(a.xmax, b.xmax) - max(a.xmin, b.xmin)

dy = min(a.ymax, b.ymax) - max(a.ymin, b.ymin) return dx\*dy

def detector(): global msg

msg = ''

cap = cv2.VideoCapture(filename) with detectionGraph.as\_default():

with tf.Session(graph=detectionGraph) as sess: while True:

ret, image\_np = cap.read()

image\_np\_expanded = np.expand\_dims(image\_np, axis=0) image\_tensor = detectionGraph.get\_tensor\_by\_name('image\_tensor:0') boxes = detectionGraph.get\_tensor\_by\_name('detection\_boxes:0') scores = detectionGraph.get\_tensor\_by\_name('detection\_scores:0') classes = detectionGraph.get\_tensor\_by\_name('detection\_classes:0')

num\_detections = detectionGraph.get\_tensor\_by\_name('num\_detections:0')

(boxes, scores, classes, num\_detections) = sess.run([boxes, scores, classes, num\_detections], feed\_dict={image\_tensor: image\_np\_expanded})

calculateCollision(boxes, classes, scores, image\_np)

cv2.putText(image\_np, msg, (230, 50), cv2.FONT\_HERSHEY\_SIMPLEX, 1.0,

(255, 0, 0), 2, cv2.LINE\_AA)

cv2.imshow('Accident Detection', image\_np) if cv2.waitKey(25) & 0xFF == ord('q'):

cv2.destroyAllWindows() break

def exit(): main.destroy()

font = ('times', 16, 'bold')

title = Label(main, text='Accident Detection') title.config(bg='light cyan', fg='pale violet red') title.config(font=font)

title.config(height=3, width=120) title.place(x=0,y=5)

font1 = ('times', 13, 'bold')

uploadButton = Button(main, text="Load & Generate CNN Model", command=loadModel) uploadButton.place(x=50,y=100)

uploadButton.config(font=font1)

pathlabel = Label(main)

pathlabel.config(bg='light cyan', fg='pale violet red') pathlabel.config(font=font1) pathlabel.place(x=460,y=100)

webcamButton = Button(main, text="Browse System Videos", command=uploadVideo)

webcamButton.place(x=50,y=150) webcamButton.config(font=font1)

webcamButton = Button(main, text="Start Accident Detector", command=detector) webcamButton.place(x=50,y=200)

webcamButton.config(font=font1)

exitButton = Button(main, text="Exit", command=exit) exitButton.place(x=330,y=250) exitButton.config(font=font1)

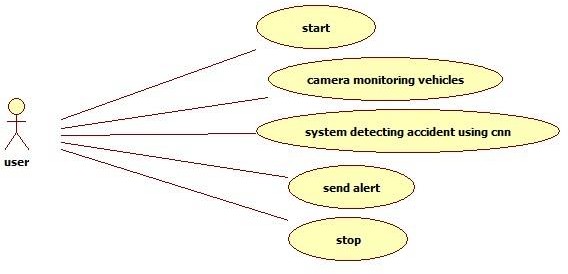
font1 = ('times', 12, 'bold') text=Text(main,height=20,width=150) scroll=Scrollbar(text) text.configure(yscrollcommand=scroll.set) text.place(x=10,y=250) text.config(font=font1)

main.config(bg='snow3') main.mainloop()

# 5.UML DIAGRAMS

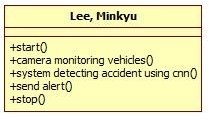
### 5.1. USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



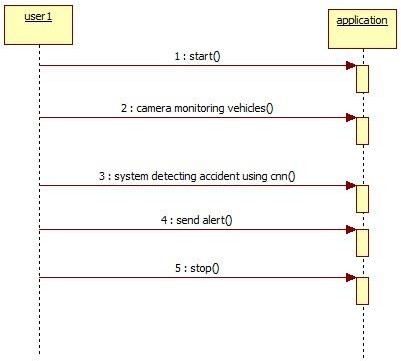
### 5.2. CLASS DIAGRAM:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



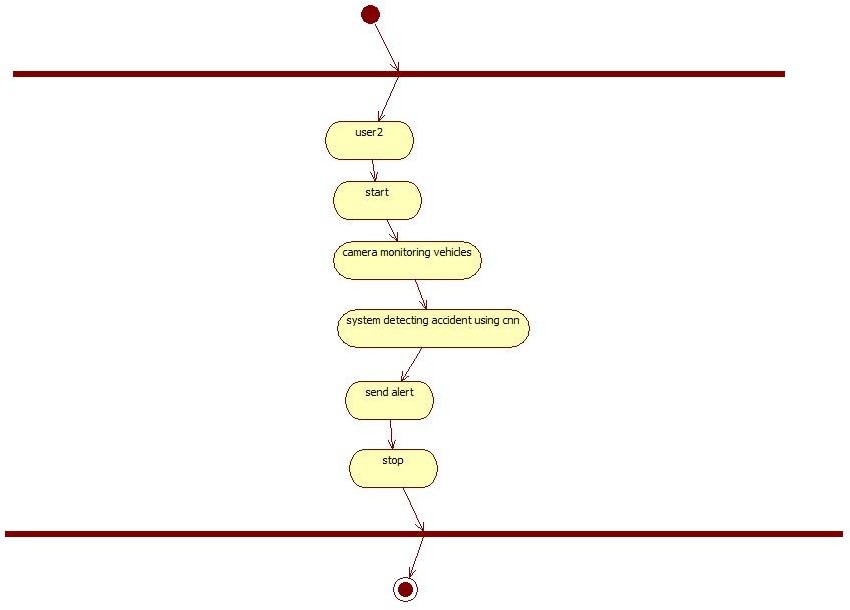
### 5.3. SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



**5.4. ACTIVITY DIAGRAM**

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



# 6.TESTING

## Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

## Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

## Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted. Invalid Input : identified classes of invalid input must be rejected. Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised. Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

## System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

## White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

## Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

## Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

## Test strategy and approach

Field testing will be performed manually and functional tests will be written

in detail.

## Test objectives

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

## Features to be tested

* Verify that the entries are of the correct format
* No duplicate entries should be allowed
* All links should take the user to the correct page.

## Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level

– interact without error.

**Test Results:**All the test cases mentioned above passed successfully. No defects encountered.

## Acceptance Testing

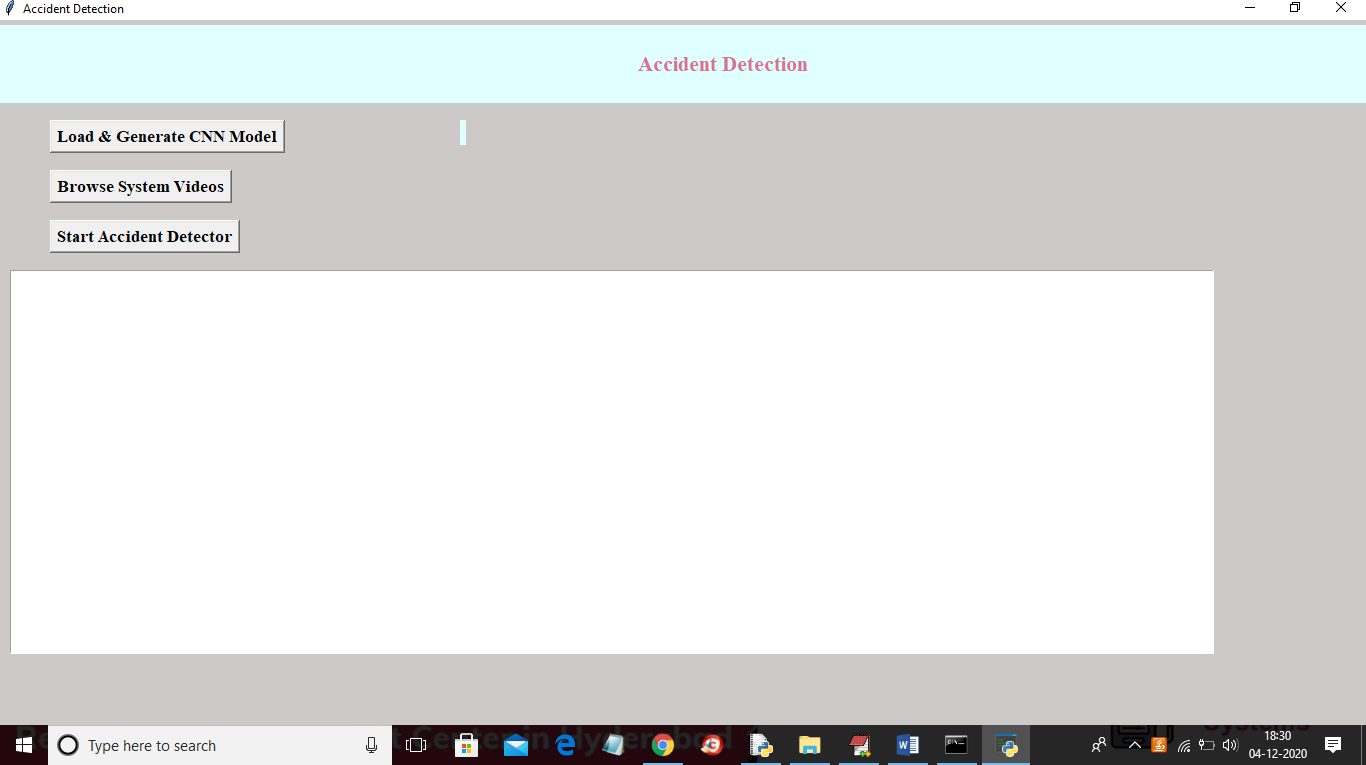
User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:**All the test cases mentioned above passed successfully. No defects encountered.

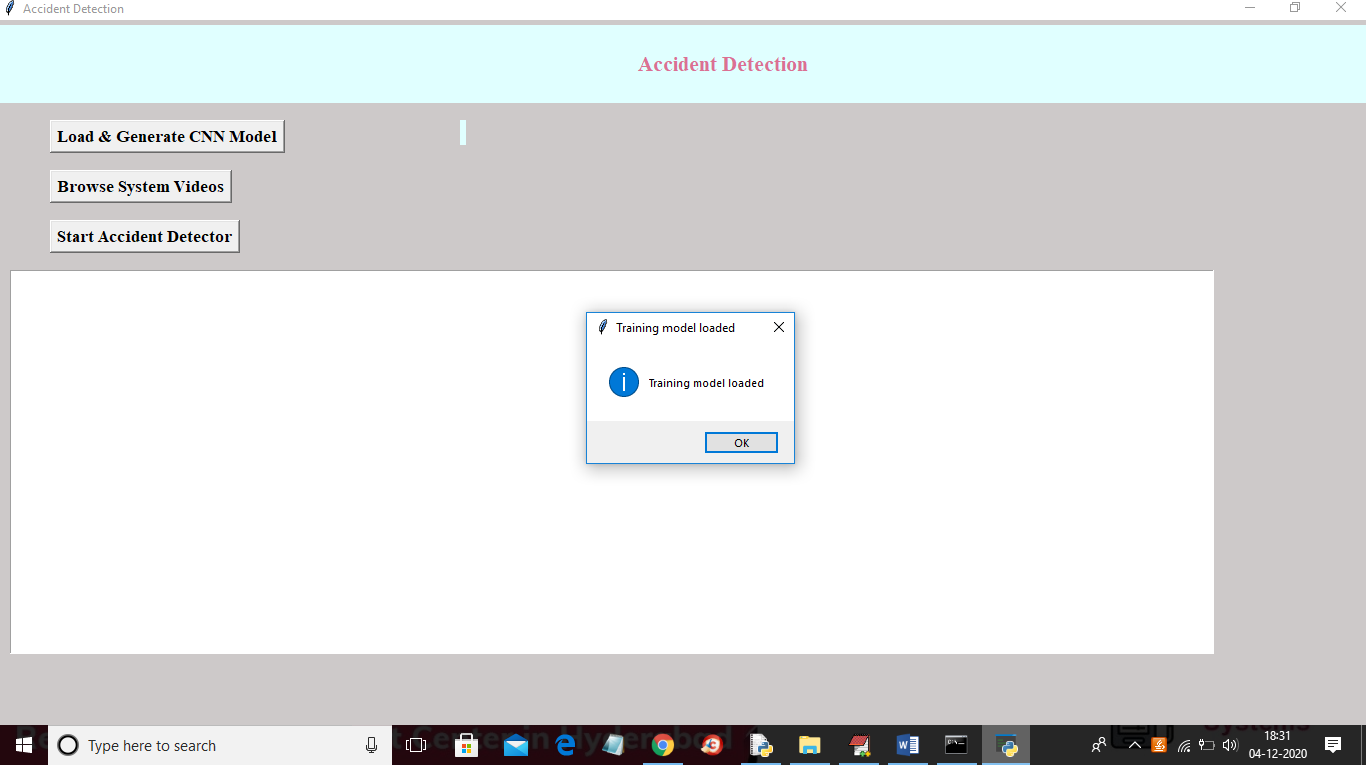
# 7.RESULTS

This project is trained with images where vehicles collided and accident occurred and in test video if anything such collision happens between vehicles then application detect as accident. Training is done with tensorflow and CNN Algorithm.

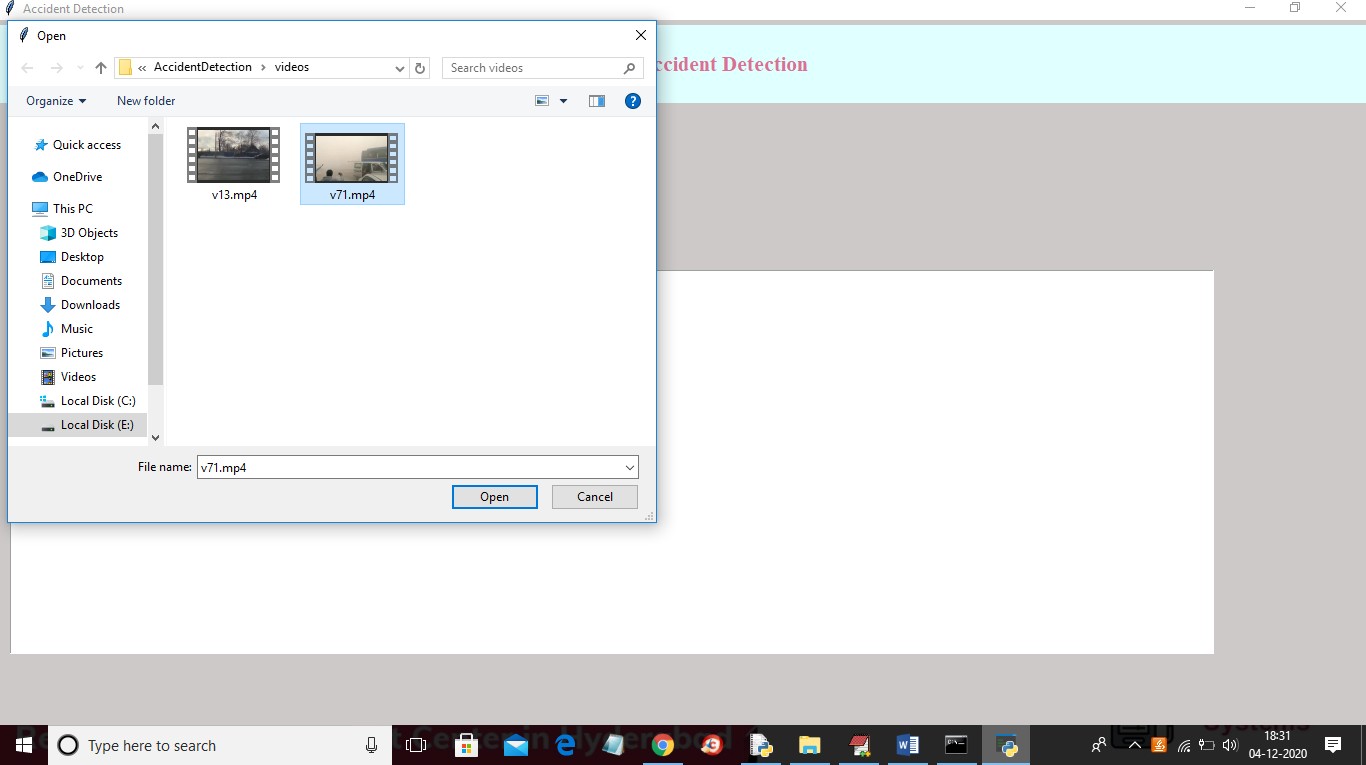
To run project double click on run.bat file to get below screen



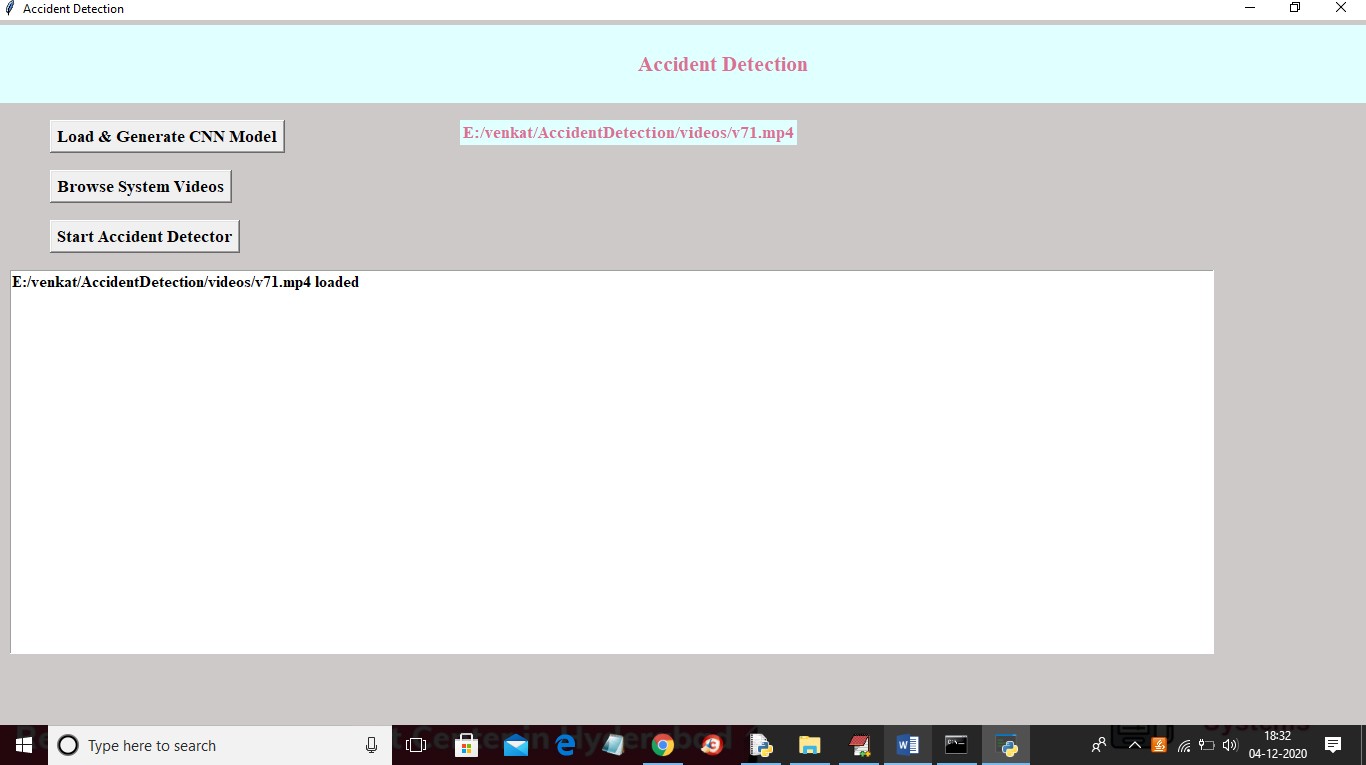
In above screen click on ‘Load & Generate CNN Model’ button to trained CNN with dataset and to load CNN model using tensorflow



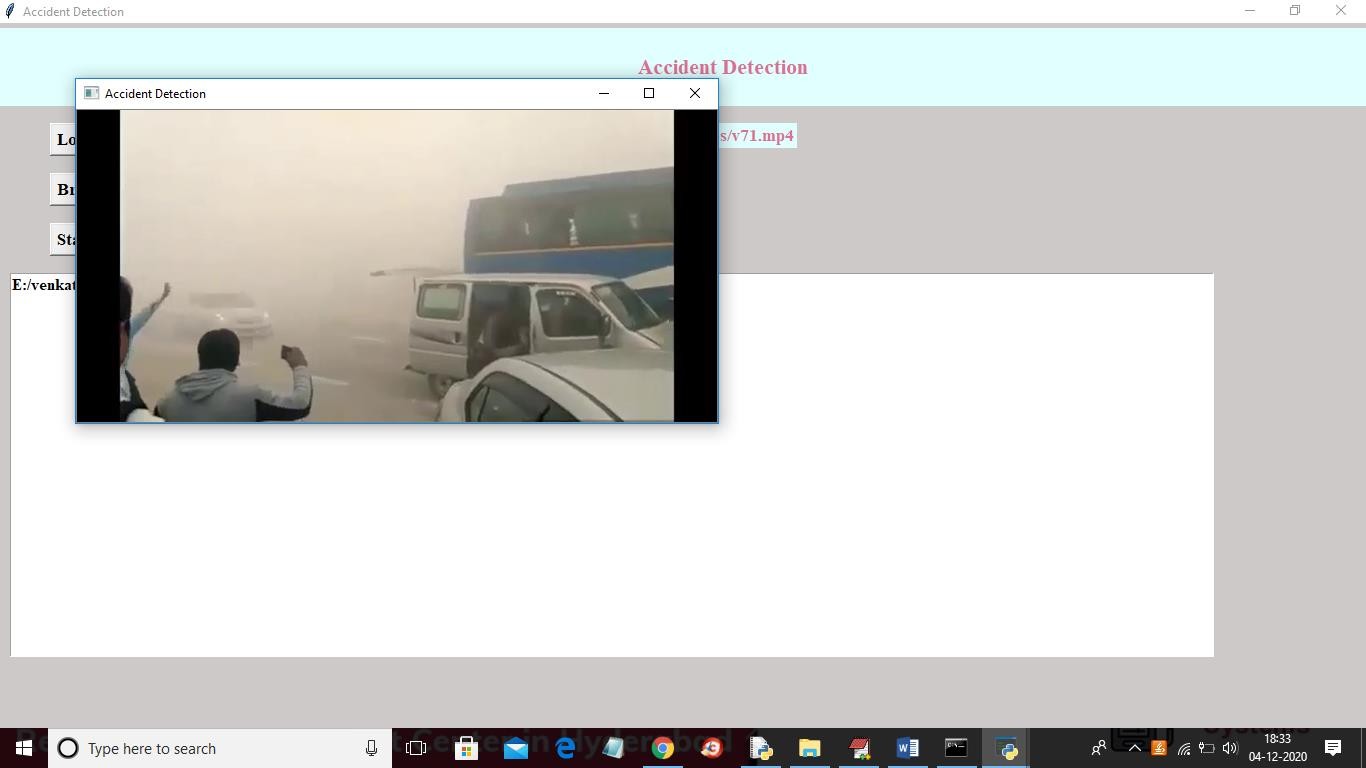
In above screen tensorflow model is loaded and now click on ‘Browse System Video’ button to upload video



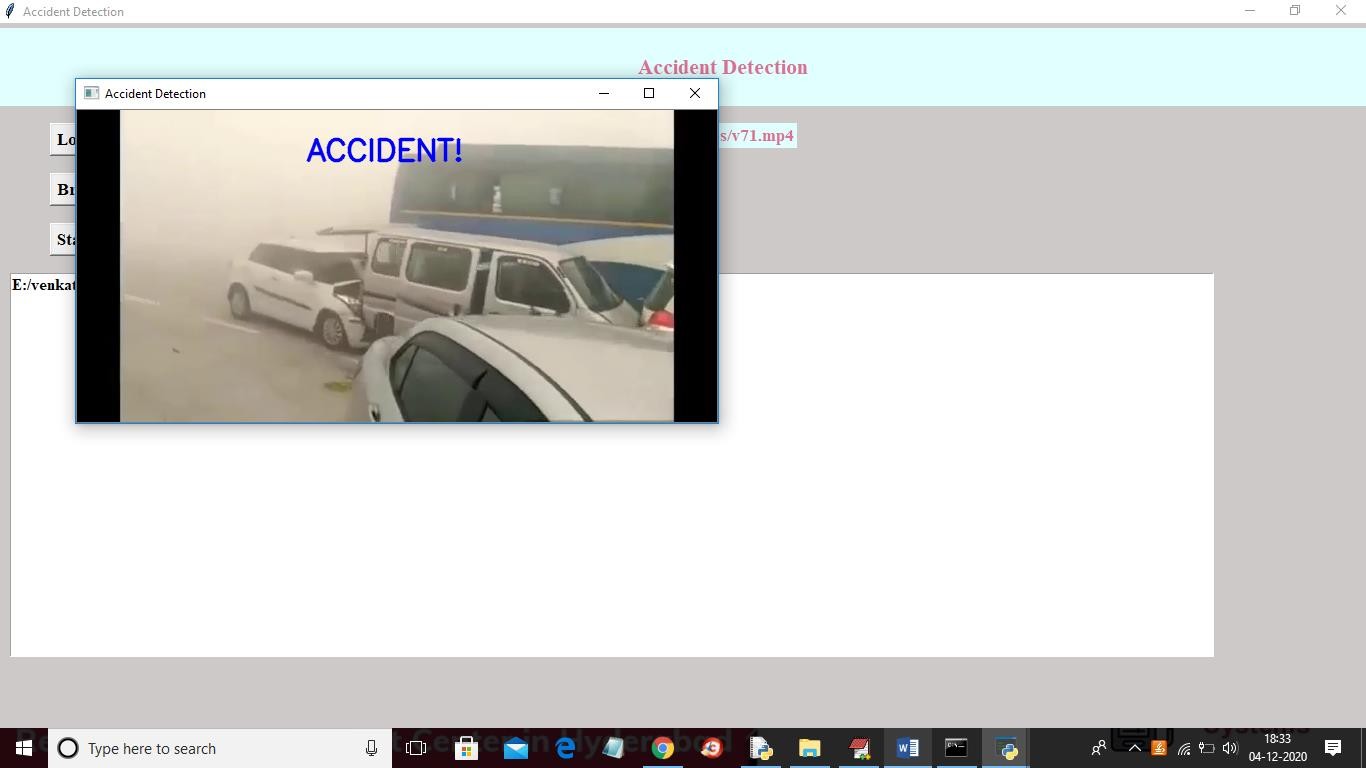
In above screen selecting and uploading video and then click on ‘Open’ button to load video



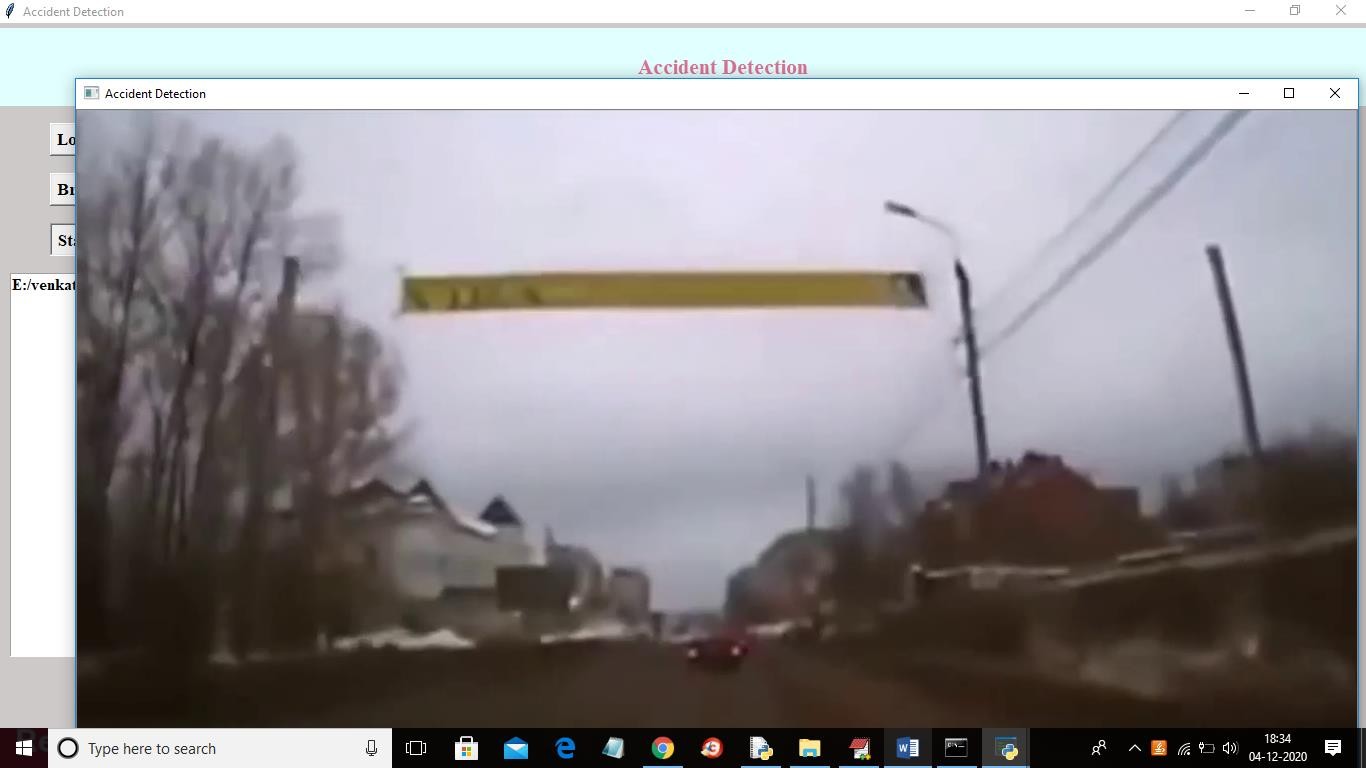
In above screen video is loaded and now click on ‘Start Accident Detector’ button to play video and detect accident

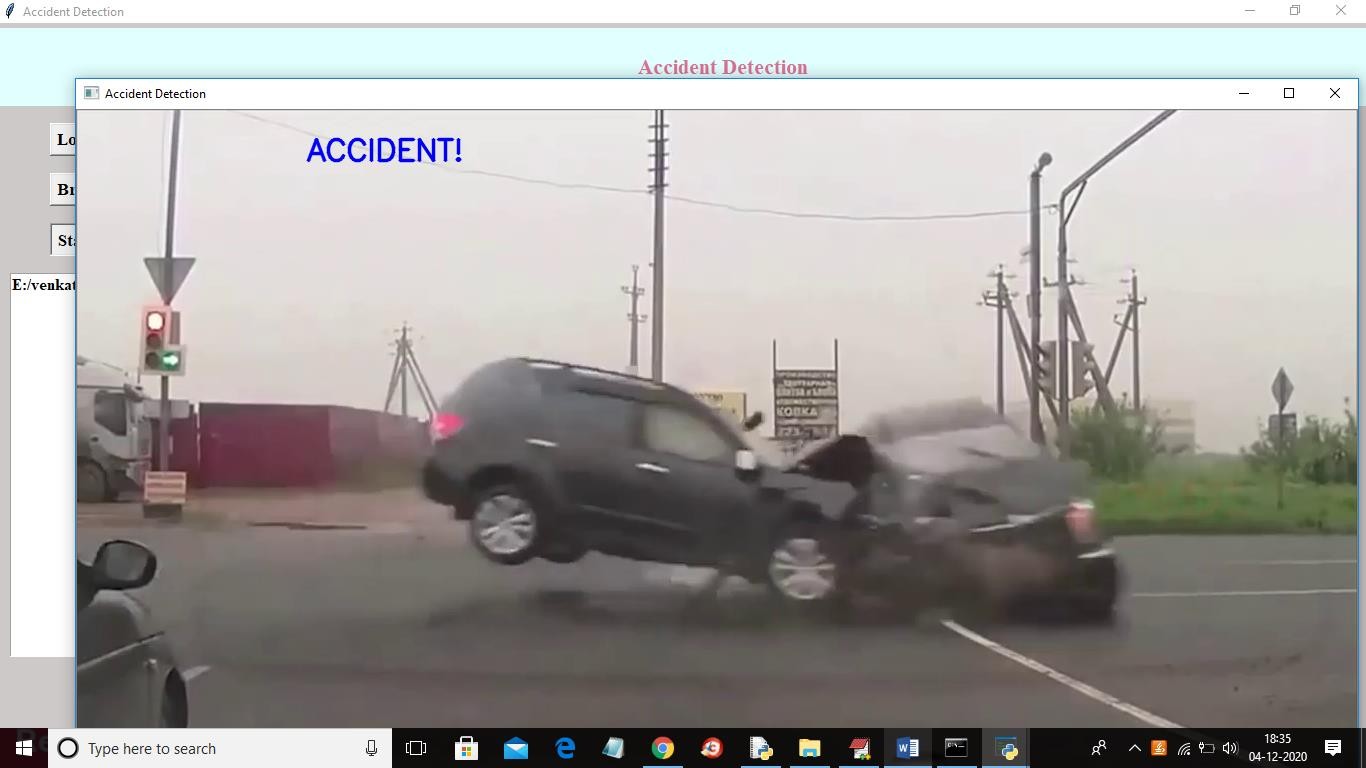


In above screen video start playing and upon accident detection will get below screen with beep sound

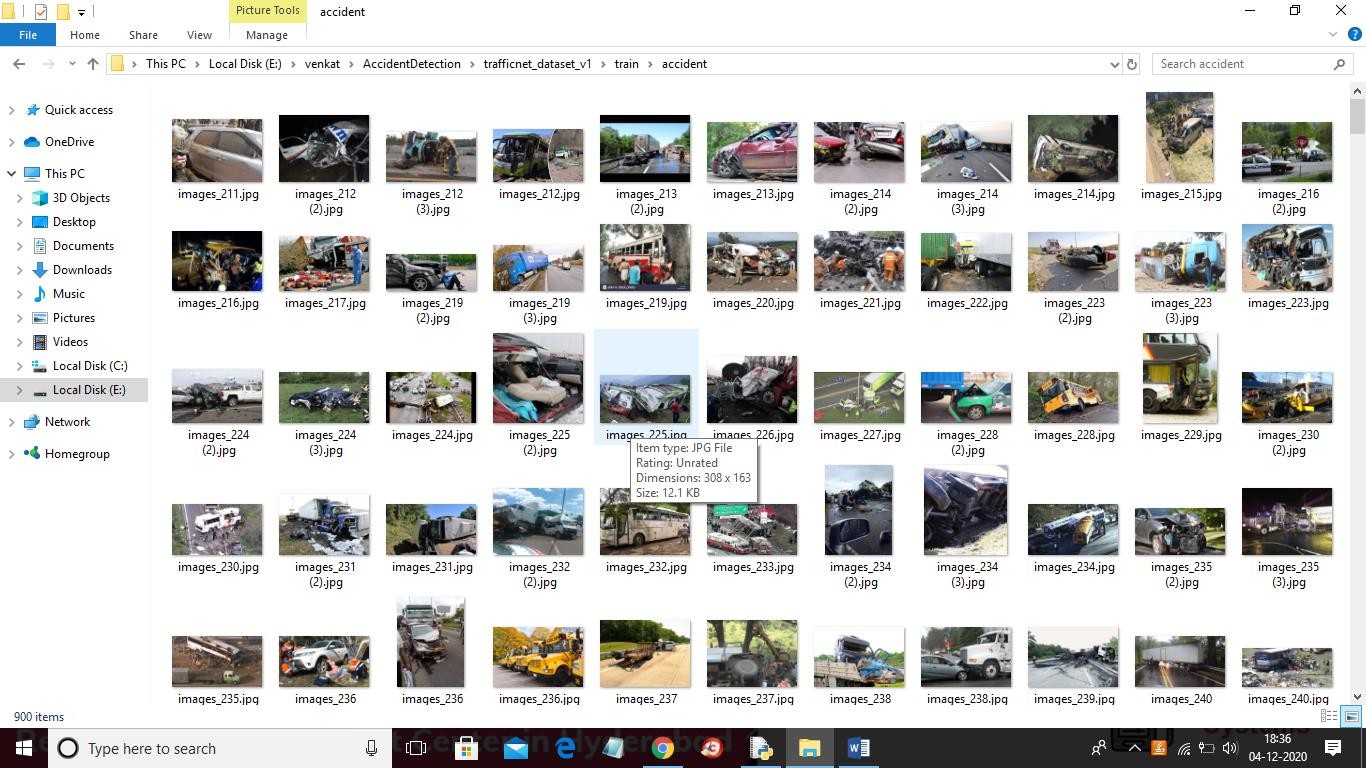


In below screen playing another video without message if normal driving appear





In above screen upon collision then accident display message will appear with beep sound In below screen application is trained with below images



# 8.CONCLUSION

The proposed system is used to detect road accidents. When an accident is detected, an alert message is sent to nearby control rooms using the GSM module. This system is more reliable and economical when compared to existing systems. It can detect accidents with high level of accuracy as the model architecture is trained using the created dataset. Our preliminary evaluation shows that the system works in a perfect manner and can be deployed over a large area. With the help of this system, immediate action can be taken by sending alert to the officials and will help the medical teams to reach the accident spot in time and save the valuable human lives. Thus, the proposed system will play an important role in the society where road accidents have nowadays become a major threat.

# 9.REFERENCES

1. "Global status report on road safety 2015", World Health Organization, 2019. [Online]. Available: <http://www.who.int/violence_injury_prevention/road_safety_status/20> 15/en/. [Accessed: 07- Mar- 2019].
2. Prabakar, S., et al. "An enhanced accident detection and victim status indicating system: Prototype." India Conference (INDICON), 2012 Annual IEEE. IEEE, 2012.
3. "Lexus Enform", Lexus, 2019. [Online]. Available: https:/[/www.lexus.com/](http://www.lexus.com/enform)e[nform.](http://www.lexus.com/enform) [Accessed: 07- Mar- 2019].
4. "OnStar Safety and Security Services", Onstar.com, 2019. [Online]. Available: https:/[/www.onst](http://www.onstar.com/us/en/services/safety-security/)a[r.com/us/en/services/safety-security/.](http://www.onstar.com/us/en/services/safety-security/) [Accessed: 07- Mar- 2019].
5. "SOSmart automatic car crash detection and notification app", SOSmart automatic car crash detection app, 2019. [Online]. Available: [http://www.sosmartapp.com.](http://www.sosmartapp.com/) [Accessed: 07- Mar- 2019].
6. C. Kockan, “Communication between vehicles” PhD thesis, Istanbul Technical University, 2008
7. Zeng, Yuanyuan, Deshi Li, and Athanasios V. Vasilakos. "Opportunistic fleets for road event detection in vehicular sensor networks." Wireless Networks 22.2 (2016): 503-521.