ACCIDENT DETECTION USING MACHINE LEARNING METHOD

at

Sathyabama Institute Of Science and Technology

Submitted in partial fulfillment of the requirements for the award of bachelor of engineering degree in computer science and engineering

MANUKONDA TEJASREE - 38110307

MUCHUMARRI SAI JAHNAVI -38110330

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING

SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
JEEPIAAR NAGAR ,RAJIV GANDHI SALAI,
CHENNAI-600119, TAMILNADU
MAY - 2022

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600119

www.sathyabama.ac.in

BONAFIDE CERTIFICATE

Certified that this project report titled **"ACCIDENT DETECTION USING MACHINE LEARNING METHODS"** is the bonafide work of MANUKONDA TEJASREE , MUCHUMARRI SAI JAHNAVI who carried out the project work under my supervision during the academic year 2021-2022.

Internal Guide

Dr. A. Viji Amutha Mary, M.Tech., Ph.D

Head of the department

Dr. S. Vigneshwari M.E., Ph.D., and Dr. L. Lakshmanan M.E., Ph.D.,

Submitted for viva voce Examination held on :

Internal Examiner **External Examiner**

DECLARATION

I, **MANUKONDA TEJASREE , MUCHUMARRI SAI JAHNAV**I hearby declare that the project report entitled **ACCIDENT DETECTION USING MACHINE LEARNING METHO**DS done by us under the guidance of **Dr. A. Viji Amutha Mary, M.Tech., Ph.Df**   is submitted in partial fulfillment of the requirements for the award of Bachelor Of Engineering Degree in Computer Science and Engineering.

DATE : 4/04/2022 M.Tejasree, M.Sai Jahnavi

PLACE : CHENNAI**.** **Signature of candidate**

ABSTRACT

There are lots of studies about preventing or detecting the car accidents. Most of them includes sensing objects which might cause accident or statistics about accidents. In this study, a system which detects happening accidents will be studied. The system will collect necessary information from neighbor vehicles and process that information using machine learning tools to detect possible accidents. Machine learning algorithms have shown success on distinguishing abnormal behaviors than normal behaviors. This study aims to analyze traffic behavior and consider vehicles which move different than current traffic behavior as a possible accident. Results showed that clustering algorithms can successfully detect accidents.The problem of deaths and injuries as a result of accidents is to be a global phenomenon. Traffic safety has been a serious concern since the start of the automobile age, almost one hundred years ago.It has been estimated that over 300,000 persons die and 10 to 15 million persons are injured every year in road accidents throughout the world.Statistics have also shown that mortality in road accidents is very high among young adults that constitute the major part of the work force. In order to overcome this problem, there is need of various has road safety strategies and measure.Losses in road accidents are unbearable, to the society as well as a developing country like us. So, it has become an essential requirement to control and arrange traffic with an advanced system to decrease the number of road accidents in our country. By taking simple precautions, based on prediction of a sophisticated system may prevent traffic accidents. Moreover, to tackle this situation where every day so many people were killed in a traffic accident. and day by day this rate is getting increased. The implementation of machine learning is a functional and a great approach to take an accurate decision with the experience to manage the current situation and the findings of the analysis part which we are suggested to traffic authorities.

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D, Dean,** School of Computing, **Dr. S. Vigneshwari, M.E., Ph.D. and Dr. L. Lakshmanan, M.E., Ph.D., Heads of the Department of Computer Science and Engineering** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. A. Viji Amutha Mary, M.Tech., Ph.D** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

The problem of deaths and injuries as a result of accidents is to be a global phenomenon. Traffic safety has been a serious concern since the start of the automobile age, almost one hundred years ago.It has been estimated that over 300,000 persons die and 10 to 15 million persons are injured every year in road accidents throughout the world.Statistics have also shown that mortality in road accidents is very high among young adults that constitute the major part of the work force. In order to overcome this problem, there is need of various has road safety strategies and measure.Losses in road accidents are unbearable, to the society as well as a developing country like us. So, it has become an essential requirement to control and arrange traffic with an advanced system to decrease the number of road accidents in our country. By taking simple precautions, based on prediction of a sophisticated system may prevent traffic accidents. Moreover, to tackle this situation where every day so many people were killed in a traffic accident. and day by day this rate is getting increased.

1.1 TECHNOLOGIES USED:

To implement this project we have used Machine learning

1.1.1 Machine Learning

Machine Learning (ML) is coming into its own, with a growing recognition that ML can play a key role in a wide range of critical applications, such as data mining, natural language processing, image recognition, and expert systems.

" A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.”

1.2 Importance of Machine Learning

Machine learning is important because it gives enterprises a view of trends in customer behaviour and business operational patterns, as well as supports the development of new products. Many of today's leading companies, such as Facebook, Google and Uber, make machine learning a central part of their operations.

1.3 Machine Learning Process

Machine learning workflow refers to the series of stages or steps involved in the process of building a successful machine learning system.

The various stages involved in the machine learning workflow are-

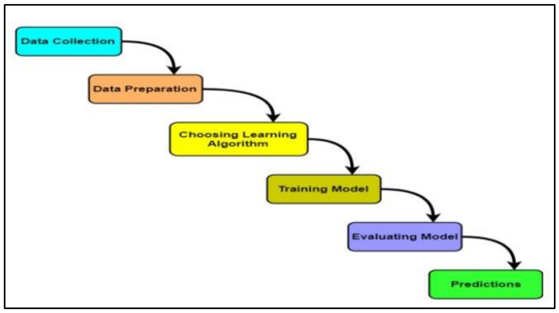


Figure 1.1 Machine Learning Process

Figure 1.2 Machine Learning Process

Data Collection- In this stage,

Data is collected from different sources.The type of data collected depends upon the type of desired project.Data may be collected from various sources such as files, databases etc.The quality and quantity of gathered data directly affects the accuracy of the desired system.

Data Preparation- In this stage,

Data preparation is done to clean the raw data.Data collected from the real world is transformed to a clean dataset.

Raw data may contain missing values, inconsistent values, duplicate instances etc.So, raw data cannot be directly used for building a model.

Choosing Learning Algorithm- In this stage,

The best performing learning algorithm is researched.It depends upon the type of problem that needs to solved and the type of data we have.If the problem is to classify and the data is labeled, classification algorithms are used.

If the problem is to perform a regression task and the data is labeled, regression algorithms are used.

If the problem is to create clusters and the data is unlabeled, clustering algorithms are used.

Training Model- In this stage,

The model is trained to improve its ability.The dataset is divided into training dataset and testing dataset.

The training and testing split is order of 80/20 or 70/30.It also depends upon the size of the dataset.Training dataset is used for training purpose.Testing dataset is used for the testing purpose.

1.4Types of Machine Learning

There are 3 types of Machine Learning

1. Supervised Machine Learning
2. Unsupervised Machine Learning
3. Reinforcement Machine Learning

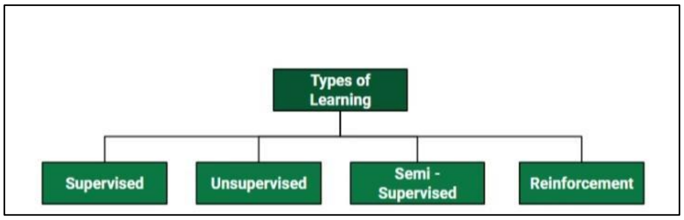


Figure 1.2 Types of Machine Learning

1.5 Supervised Machine Learning :

Supervised machine learning is based on supervision. It means in the supervised learning technique, we train the machines using the "labelled" dataset, and based on the training, the machine predicts the output. Here, the labelled data specifies that some of the inputs are already mapped to the output. More preciously, we can say; first, we train the machine with the input and corresponding output, and then we ask the machine to predict the output using the test dataset.

Supervised machine learning can be classified into two types of problems, which are given below:

a) Classification : Classification algorithms are used to solve the classification problems in which the output variable is categorical, such as "Yes" or No, Male or Female, Red or Blue, etc

b) Regression : Regression algorithms are used to solve regression problems in which there is a linear relationship between input and output variables. These are used to predict continuous output variables, such as market trends, weather prediction, etc.

1.6 Unsupervised Machine Learning :

In unsupervised learning, the models are trained with the data that is neither classified nor labelled, and the model acts on that data without any supervision.

The main aim of the unsupervised learning algorithm is to group or categories the unsorted dataset according to the similarities, patterns, and differences.

Unsupervised Learning can be further classified into two types, which are given below:

a) Clustering : The clustering technique is used when we want to find the inherent groups from the data. It is a way to group the objects into a cluster such that the objects with the most similarities remain in one group and have fewer or no similarities with the objects of other groups.

b) Association : Association rule learning is an unsupervised learning technique, which finds interesting relations among variables within a large dataset. The main aim of this learning algorithm is to find the dependency of one data item on another data item and map those variables accordingly so that it can generate maximum profit.

1.7 Reinforcement Machine Learning :

Reinforcement learning works on a feedback-based process, in which an AI agent (A software component) automatically explore its surrounding by hitting & trail, taking action, learning from experiences, and improving its performance.

Reinforcement learning is categorized mainly into two types of methods/algorithms:

a) Positive Reinforcement Learning: Positive reinforcement learning specifies increasing the tendency that the required behaviour would occur again by adding something. It enhances the strength of the behaviour of the agent and positively impacts it.

b) Negative Reinforcement Learning: Negative reinforcement learning works exactly opposite to the positive RL. It increases the tendency that the specific behaviour would occur again by avoiding the negative condition

CHAPTER -2 LITERATURE

REVIEW

2 LITERATURE SURVEY:

2.1 Angshuman, G., 2004. An Incident Detection Algorithm Based On a Discrete State Propagation Model of Traffic Flow.

Proposed work

Incidents are defined as random and nonrecurring events such as vehicular crashes,

disabled vehicles, spilled loads, temporary maintenance and construction activities, and other unusual events that disrupt the normal flow of traffic. Incident-related congestion is a common occurrence on busy roadways. The number of incidents per million-vehicle-miles has been reported to be between 20 and 200 and lane-blocking incidents lasting more than 45 minutes per 100 million-vehicle-miles have been reported to be 1.09. It has been estimated that 52 to 58 percent of the traffic congestion in urban areas is due to incidents, amounting to 2 billion vehicle-hours of delay and a cost of $40 billion in terms of hours of delay and excess fuel consumption in 2001, as reported in the 2003 Annual emergency services, lost time and reduction in productivity, increased cost of goods and services, reduced air quality, increased vehicle maintenance costs and reduced quality of life.

2.2 : Kukshya, V., Krishnan, H., Kellum, C., n.d. Design of a system solution for relative positioning of vehicles using vehicle-to-vehicle radio communications during GPS outages.

Proposed work

Active safety applications for vehicles have been at the forefront of the automotive safety community for a number of years. Cooperative collision warning based on vehicle-to-vehicle radio communications and GPS systems is one such promising active safety application that has evoked considerable interest among automobile manufacturers and research communities worldwide. In this paper, we address one of the key functional components of the cooperative collision warning application, which is, accurate estimation of relative positions of all the neighboring vehicles based on real-time exchange of their individual GPS position coordinates, and then propose a novel system solution for achieving the same (relative positioning functionality) during persistent GPS outages. Based on survey results, we also qualitatively assess various radio based ranging and relative positioning techniques, experimentally evaluate the received signal strength based ranging technique, and comment on their suitability for our proposed solution.

2.3 Xu, Q., Mak, T., Ko, J., Sengupta, R., 2004. Vehicle-to-vehicle safety messaging in DSRC. AC

The design of layer-2protocols for a vehicle to send safety messages to other vehicles. The target is to send vehicle safety messages with high reliability and low delay. The communication is one-to-many, local, and geo-significant. The vehicular communication network is ad-hoc, highly mobile, and with large numbers of contending nodes. We design several random access protocols for medium access control. The protocols fit in the DSRC multi-channel architecture.Analytical bounds on performance of the addressed protocols are derived. Simulations are conducted to assess the reception reliability and channel usage of the protocols. The sensitivity of the protocol performance is evaluated under various offered traffic and vehicular traffic flows. The results show our approach is feasible for vehicle safety messages in DSRC.

2.4 Krakiwsky, E.J., Harris, C.B., Wong, R.V.., 1988. A Kalman filter for integrating dead reckoning, map matching and GPS positioning, in: , IEEE Position Location and Navigation Symposium, 1988. Record. Navigation into the 21st Century. IEEE PLANS ’88. Presented at the , IEEE Position Location and Navigation Symposium, 1988. Record. Navigation into the 21st Century. IEEE PLANS ’88, IEEE, pp. 39-46.

To make driving easier and safer, modern vehicles are equipped with driver support systems. Some of these systems, for example navigation or curvature warning systems, need the global position of the vehicle. To determine this position, the Global Positioning System (GPS) or a Dead Reckoning (DR) system can be used. However, these systems have often certain drawbacks. For example, DR systems suffer from error growth with time and GPS signal masking can occur. By integrating the DR position and the GPS position, the complementary characteristics of these two systems can be used advantageously. In this thesis, low cost in-vehicle sensors (gyroscope and speedometer) are used to perform DR and the GPS receiver used has a low update frequency. The two systems are integrated with an extended Kalman filter in order to estimate a position. The evaluation of the implemented positioning algorithm shows that the system is able to give an estimated position in the horizontal plane with a relatively high update frequency and with the accuracy of the GPS receiver used. Furthermore, it is shown that the system can handle GPS signal masking for a period of time. In order to increase the performance of a positioning system, map matching can be added. The idea with map matching is to compare the estimated trajectory of a vehicle with roads stored in a map data base, and the best match is chosen as the position of the vehicle. In this thesis, a simple off-line map matching algorithm is implemented and added to the positioning system. The evaluation shows that the algorithm is able to distinguish roads with different direction of travel from each other and handle off-road driving.

2.5 Boukerche, A., Oliveira, H., Nakamura, E., Loureiro, A., 2008. Vehicular Ad Hoc Networks: A New Challenge for Localization-Based Systems. Computer Communications 31, 2838–2849.

The new kind of ad hoc network is hitting the streets: Vehicular Ad Hoc Networks. (VANets). In these networks, vehicles communicate with each other and possibly with a roadside infrastructure to provide a long list of applications varying from transit safety to driver assistance and Internet access. In these networks, knowledge of the real-time position of nodes is an assumption made by most protocols, algorithms, and applications. This is a very reasonable assumption, since GPS receivers can be installed easily in vehicles, a number of which already comes with this technology. But as VANets advance into critical areas and become more dependent on localization systems, GPS is starting to show some undesired problems such as not always being available or not being robust enough for some applications. For this reason, a number of other localization techniques such as Dead Reckoning, Cellular Localization, and Image/Video Localization has been used in VANets to overcome GPS limitations. A common procedure in all these cases is to use Data Fusion techniques to compute the accurate position of vehicles, creating a new paradigm for localization in which several known localization techniques are combined into a single solution that is more robust and precise than the individual approaches. In this paper, we further discuss this subject by studying and analyzing the localization requirements of the main VANet applications.

CHAPTER - 3 PROJECT DESCRIPTION

3.1 EXISTING SYSTEM:

In existing method Data Mining techniques are used to identify the locations where high frequency accidents are occurred and analyze them to identify the factors that have an effect on road accidents at that locations. The first task is to divide the accident location into k groups using the k-means clustering algorithm based on road accident frequency counts. Then, association rule mining algorithm applied in order to find out the relationship between distinct attributes which are in accident data set and according to that know the characteristics of locations.

3.2 PROPOSED SYSTEM:

Now in this method classification techniques will be using for identifying the accident prone area's. The accident data records which can help to understand the characteristics of many features like drivers behavior, roadway conditions, light condition, weather conditions and so on. This can help the users to compute the safety measures which is useful to avoid accidents. The data set can be analyzing based on Yolo(You only look once) algorithm will gives the accurate dataset. The models are performed to identify statistically significant factors which can be able to predict the probabilities of crashes and injury that can be used to perform a risk factor.

3.3 WORKING

In this project, we will be developing a system which will be constantly monitor vehicles to check whether accident occurred or not. first we need to collect the data from the website.In this project we have collect the data from kaggle.after we need topreprocess the data, because when we collect the data from. the website there is some missing data or some aspects will occur, so to avoid this we need to preprocess the data ,this is called as data preprocessing Then apply the Machine learning algorithms to predict the output . after prediction we will save the data .Thus, helps to effectively monitor accidents among the general public .

3.4 MODULE DESCRIPTION

- Data collection and Data Preprocessing
- Data cleaning
- Visualization

3.4.1 Data collection and Data Preprocessing :

Collecting data for training the ML model is the basic step in the machine learning pipeline. The predictions made by ML systems can only be as good as the data on which they have been trained. Following are some of the problems that can arise in data collection:

Inaccurate data. The collected data could be unrelated to the problem statement.

Missing data. Sub-data could be missing. That could take the form of empty values in columns or missing images for some class of prediction.

Data imbalance. Some classes or categories in the data may have a disproportionately high or low number of corresponding samples. As a result, they risk being under-represented in the model.

Data bias. Depending on how the data, subjects and labels themselves are chosen, the model could propagate inherent biases on gender, politics, age or region, for example. Data bias is difficult to detect and remove.

Data Preprocessing : Real-world raw data and images are often incomplete, inconsistent and lacking in certain behaviors or trends. They are also likely to contain many errors. So, once collected, they are pre-processed into a format the machine learning algorithm can use for the model.

Pre-processing includes a number of techniques and actions:

Data cleaning. These techniques, manual and automated, remove data incorrectly added or classified.

Data imputations. Most ML frameworks include methods and APIs for balancing or filling in missing data. Techniques generally include imputing missing values with standard deviation, mean, median and k-nearest neighbors (k-NN) of the data in the given field.

3.4.2 Data cleaning :

Data cleaning is one of the important parts of machine learning. It plays a significant part in building a model. It surely isn’t the fanciest part of machine learning and at the same time, there aren’t any hidden tricks or secrets to uncover. However, the success or failure of a project relies on proper data cleaning. Professional data scientists usually invest a very large portion of their time in this step because of the belief that “Better data beats fancier algorithms”.

3.4.3 Data visualization :

Data visualization is the graphical representation of information and data in a pictorial or graphical format(Example: charts, graphs, and maps). Data visualization tools provide an accessible way to see and understand trends, patterns in data, and outliers. Data visualization tools and technologies are essential to analyzing massive amounts of information and making data-driven decisions. The concept of using pictures is to understand data that has been used for centuries. General types of data visualization are Charts, Tables, Graphs, Maps

CHAPTER -4
METHODOLOGY FLOW

This chapter describes the overall design. It also describes each module that is to be implemented.

4.1 SYSTEM ARCHITECTURE

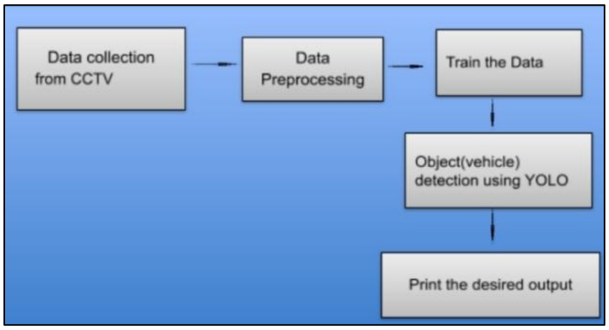

Data collection from CCTV
Data Preprocessing
Train the Data
Object(vehicle) detection using YOLO
Print the desired output

Figure: 4.1: System Architecture

4.2 Input images:

Our Accident detection data set

This dataset consists of **1,376 images** belonging to two classes:

- Accident : 690 images
- Non - Accident : 686 images

Our goal is to train a custom deep learning model to detect whether a accident has occurred or not .

4.2.1 Accident dataset creation :

To create this dataset, Then creating a custom computer vision Python script to add accidents to them, thereby creating an *artificial* (but still real-world applicable) datasetThis method is actually a lot easier than it sounds once you apply landmarks to the problem.

4.2.2 Working :

First we collect the data from CCTV cameras.Then we apply data preprocessing technique.after preprocessing the data we train the system by using data set(accident) & (non accident).after completion of training we will apply YOLO algorithm to detect the object.By using YOLO algorithm it will divide into frames then the frames will detect whether the accident is happened or not.

4.3 DISADVANTAGES OF EXISTING SYSTEM:

- Time consuming Process
- More prone to damages

4.4 ADVANTAGES OF PROPOSED SYSTEM:

- Less prone to damages
- Faster
- Less hassle

4.5 APPLICATIONS:

- Used in traffic .
- Used in Highways .
- We can also use this at driving schools and vehicle exhibitions

CHAPTER-5
TOOLS AND TECHNIQUES

SOFTWARE REQUIREMENTS

- Python
- Pycharm

5.1. Python

In this project, python is used as the programming language. In technical terms, Python is an object-oriented, high-level programming language with integrated dynamic semantics primarily for web and app development. It is extremely attractive in the field of Rapid Application Development because it offers the dynamic typing and binding options. Python is relatively simple, so it's easy to learn since it requires a unique syntax that focuses on readability. Developers can read and translate Python code much easier than other languages. In turn, this reduces the cost of program maintenance and development because it allows teams to work collaboratively without significant language and experience barriers.

Python supports the use of modules and packages, which means that programs can be designed in a modular style and code can be reused across a variety of projects. Once you've developed a module or package you need, it can be scaled for use in other projects, and it's easy to import or export these modules. One of the most promising benefits of Python is that both the standard library and the interpreter are available free of charge, in both binary and source form. There is no exclusivity either, as Python and all the necessary tools are available on all major platforms. Therefore, it is an enticing option for developers who don't want to worry about paying high development costs.

If this description of Python over your head, don't worry. You will understand it soon enough. What you need to take away from this section is that Python is a programming language used to develop software on the web and in app form, including mobile. It's relatively easy to learn, and the necessary tools are available to all free of cost.

5.2 PYCHARM

PyCharm is the Python IDE by JetBrains, designed for professional Python developers. Industry-leading code completion, code navigation, safe refactoring, and smart debugging are just a few important features that contribute to make professional software development a more productive and enjoyable experience. PyCharm Professional Edition comes with wide support for Python web frameworks, modern JavaScript development, as well as with advanced database tools and scientific tools integrations.

Pros:

- "I have been using PyCharm since over 4 years now and It is one of the best IDEs out there for python developers. Great product with auto-complete features."

- "Also PyCharm have some tools to find errors and helps you to correct them. I also like the style and colors very nice for me”.

- "The best all in one IDE out there, the python supporting features are great and it has a many templates for different projects for ease of architecture."

- "PyCharm is probably the best IDE for Python projects as it has so many Python orientated features. Personally, I only use PyCharm for the occasional Python project and I am very satisfied with that."

5.3 Software Code Or Deployment:

```
import cv2
import os
def getFrame(sec, path, out_path):
    vidcap = cv2.VideoCapture(path)
    vidcap.set(cv2.CAP_PROP_POS_MSEC,sec*1000)
    hasFrames,image = vidcap.read()
    if hasFrames:
        st_path = os.path.join(out_path, "image"+str(count)+".jpg")
        cv2.imwrite(st_path, image)     # save frame as JPG file
    return hasFrames

temp_dir = 'test_input'
output_dir = 'output_frames'
```

```
for video_file in os.listdir(temp_dir):
    path = os.path.join(temp_dir, video_file)
    print(path)
    output_dir_folder = os.path.join(output_dir, video_file)
    os.mkdir(output_dir_folder)
    sec = 0
    frameRate = 2.0 #//it will capture image in each 0.5 second
    count=1
    success = getFrame(sec, path, output_dir_folder)
    while success:
        count = count + 1
        sec = sec + frameRate
        sec = round(sec, 2)
        success = getFrame(sec, path, output_dir_folder)

import cv2
import tensorflow as tf
import os
from skimage.transform import resize
import numpy as np
from PIL import Image
model = tf.keras.models.load_model('model_train')
categories = ["Accident", "Non-Accident"]
```

```
main_dir = "test_frames"
for dirs in os.listdir(main_dir):
    test_dir = os.path.join(main_dir, dirs)
    print(test_dir)
    a=0
    for img_t in os.listdir(test_dir):

       image = cv2.imread(os.path.join(test_dir, img_t), cv2.IMREAD_GRAYSCALE)
        image = resize(image, (128, 128))
        image = image / 255
        img_arr = image.reshape([1] + list(image.shape) + [1])
        predict_x= model.predict(img_arr)
        predict_class=np.argmax(predict_x,axis=1)
        if categories[predict_class[0]]=='Accident':
            print(img_t, "\t", categories[predict_class[0]])
            im=Image.open('test_frames/{}/{}'.format(dirs,img_t))
            im.show()
import cv2
import os
def getFrame(sec, path, out_path):
    vidcap = cv2.VideoCapture(path)
    vidcap.set(cv2.CAP_PROP_POS_MSEC,sec*1000)
    hasFrames,image = vidcap.read()
    if hasFrames:
        st_path = os.path.join(out_path, "image"+str(count)+".jpg")
        cv2.imwrite(st_path, image)     # save frame as JPG file
    return hasFrames

temp_dir = 'test_input'
output_dir = 'output_frames'

for video_file in os.listdir(temp_dir):
    path = os.path.join(temp_dir, video_file)
    print(path)
    output_dir_folder = os.path.join(output_dir, video_file)
    os.mkdir(output_dir_folder)
    sec = 0
    frameRate = 2.0 #//it will capture image in each 0.5 second
    count=1
    success = getFrame(sec, path, output_dir_folder)
    while success:
        count = count + 1
        sec = sec + frameRate
        sec = round(sec, 2)
        success = getFrame(sec, path, output_dir_folder)
import tensorflow as tf

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Activation, Dropout, Dense, Flatten
import numpy as np
import matplotlib.pyplot as plt
import cv2
from skimage.transform import resize

train = tf.keras.models.load_model('model_train')
categories = ["Accident", "Non-Accident"]

img_test = np.load("labels/test_features.npy")
label_test = np.load("labels/test_labels.npy")
img_test = img_test / 255
img_test = img_test.reshape(list(img_test.shape) + [1])
print(img_test.shape)
train.evaluate(img_test, label_test)[1]
print("index \t label  \t predicted")
#classes = train.predict_classes(img_test)
classes = train.predict(img_test)
np.argmax(classes)
for i in range(len(label_test)):
    print(i+1,"\t", categories[label_test[i]])
#, "\t", categories[classes[i]]
import tensorflow as tf

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Activation, Dropout, Dense, Flatten
import numpy as np
import matplotlib.pyplot as plt
import cv2
from skimage.transform import resize

X = np.load("labels/train_features.npy")
print("St 1")
Y = np.load("labels/train_labels.npy")
print("St 2")
X = X / 255
print("St 3")
X = X.reshape(list(X.shape) + [1])
print("St 4")
print(X.shape)

#creating my model
model = Sequential();

#adding first layer i.e convolutional layer, passing input_shape only in first layer
model.add(Conv2D(32, (5 , 5), activation="relu", input_shape=(128, 128, 1)))

#adding a pooling layer
model.add(MaxPooling2D(pool_size = (2 , 2)))

#adding another convolutional layer
model.add(Conv2D(32, (5 , 5), activation="relu"))

#adding another pooling layer
```

```
model.add(MaxPooling2D(pool_size = (2 , 2)))

#adding flatteing layer i.e. fully connected layer
model.add(Flatten())

#Embedding nuerons using dense layer
model.add(Dense(1000, activation="relu"))

#Adding a dropout with 50% droupout rate
model.add(Dropout(0.5))

#Embedding nuerons using dense layer
model.add(Dense(500, activation="relu"))

#Adding a dropout with 50% droupout rate
model.add(Dropout(0.5))

#Embedding nuerons using dense layer
model.add(Dense(250, activation="relu"))

#Embedding nuerons using dense layer
model.add(Dense(2, activation="softmax"))

#compiling the model
model.compile(loss = "sparse_categorical_crossentropy", optimizer = "adam",
metrics = ["accuracy"])
```

```
#training the model
train = model.fit(X, Y, batch_size = 256, epochs = 10, validation_split=0.2, shuffle=True)
model.save('model_train')

#plotting the model accuracy
plt.plot(train.history["accuracy"])
plt.plot(train.history["val_accuracy"])
plt.title("Model Accuracy Visualization")
plt.ylabel("Accuracy")
plt.xlabel("Epoch")
plt.legend(['train', 'val'], loc="upper right")
plt.show()
import numpy as np
import random
import matplotlib.pyplot as plt
import os
import cv2
from skimage.transform import resize

Train_DIR = r"/home/sparsh/Desktop/Accident_Detection/Accident-Dataset/train"
Test_DIR = r"/home/sparsh/Desktop/Accident_Detection/Accident-Dataset/test"
categories = ["Accident", "Non-Accident"]
test_data = []
train_data = []
```

```
def getTrainData():
    i = 0
    for index, category in enumerate(categories):
        path = os.path.join(Train_DIR , category)
        for img in os.listdir(path):
            i += 1
            print(i)
            try:
               image = cv2.imread(os.path.join(path, img),
cv2.IMREAD_GRAYSCALE)
                image = resize(image, (128, 128))
                train_data.append([image , index])
            except Exception as e:
                pass



def getTestData():
    i = 0
    for index, category in enumerate(categories):
        path = os.path.join(Test_DIR , category)
        for img in os.listdir(path):
            i += 1
            print(i)
            try:
                image = cv2.imread(os.path.join(path, img),
cv2.IMREAD_GRAYSCALE)
                image = resize(image, (128, 128))
                test_data.append([image , index])
            except Exception as e:
                pass
print("Getting Train Data")
getTrainData()

print("Getting Test Data")
getTestData()

random.shuffle(train_data)
random.shuffle(test_data)

train_features = []
train_labels = []
test_features = []
test_labels = []

for feature, label in train_data:
    train_features.append(feature)
    train_labels.append(label)

train_features = np.array(train_features)
train_labels = np.array(train_labels)
```

```
print(type(train_features))

for feature, label in test_data:
    test_features.append(feature)
    test_labels.append(label)

test_features = np.array(test_features)
test_labels = np.array(test_labels)
print(type(test_features))

print("Saving Numpy Arrays")

np.save("train_features", train_features)
np.save("train_labels", train_labels)
np.save("test_features", test_features)
np.save("test_labels", test_labels)
```

CHAPTER-6

FINAL RESULTS

We take the input video from a Source (CC TV camera □) and divide the video into severalframes.On each frame a Vehicle is detected whether Accident has occurred or not. If accident occurred it will print the image number with “accident” message.

6.1 FINAL RESULTS

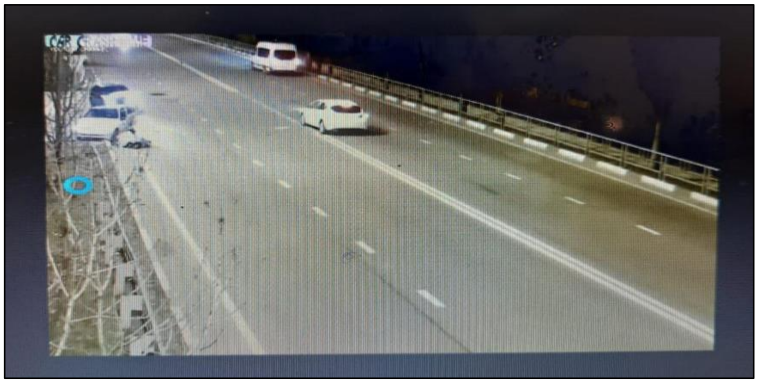


Fig 6.1 : Accident occurred

When we run the we got this screenshot because of accident has occurred , System will take the Video from CC TV cameras and cut the video in such frames then each frame will be detected using YOLO Algorithm . In this frame two vehicles got smashed to each other .

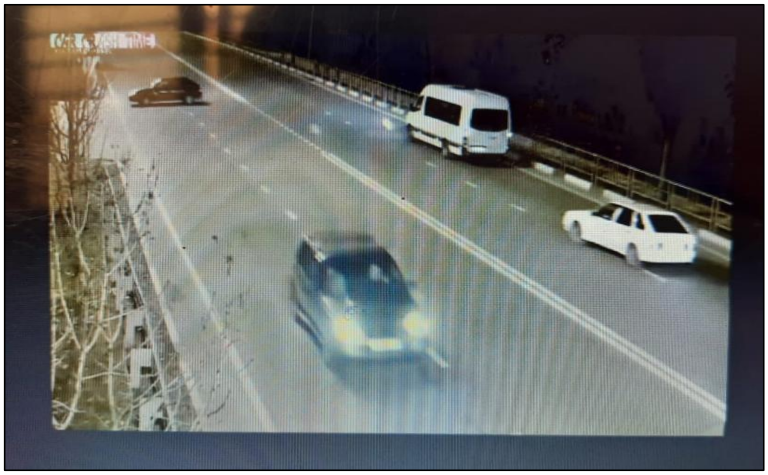


Fig 6.2 Accident will happen

When we run the we got this screenshot because of accident has occurred , System will take the Video from CC TV cameras and cut the video in such frames then each frame will be detected using YOLO Algorithm . In this frame vehicle is going in wrong direction soo Accident is going to happen so it will detected that also .

Comment [1]: Accident occurred

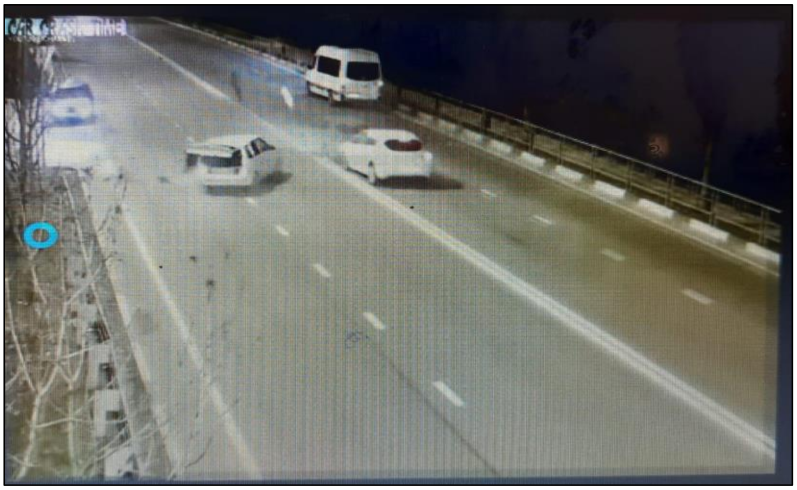
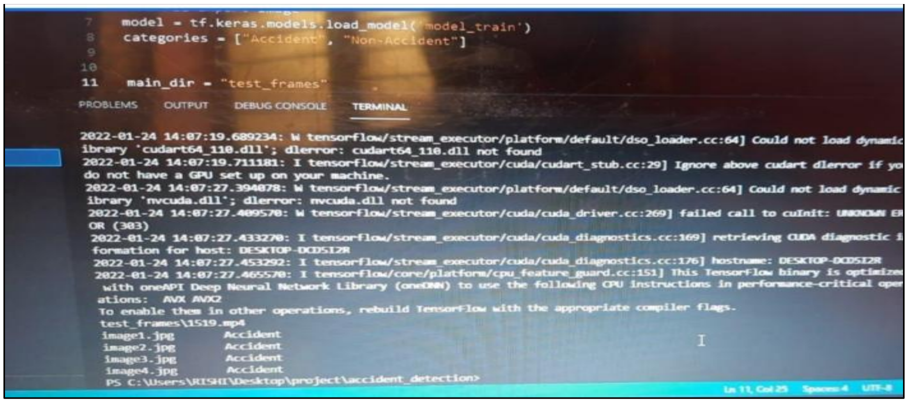


Fig : 6.3 Accident occurred.

When we run the we got this screenshot because of accident has occurred , System will take the Video from CC TV cameras and cut the video in such frames then each frame will be detected using YOLO Algorithm . In this frame two vehicles got smashed to each other .



```
model = tf.keras.models.load_model('model_train')
categories = ["Accident", "Non-Accident"]


main_dir = "test_frames"
```

```
2022-01-24 14:07:19.689234: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic
ibrary 'cudart64_110.dll'; dlerror: cudart64_110.dll not found
2022-01-24 14:07:19.711181: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dlerror if yo
do not have a GPU set up on your machine.
2022-01-24 14:07:27.394078: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic
ibrary 'nvcuda.dll'; dlerror: nvcuda.dll not found
2022-01-24 14:07:27.409570: W tensorflow/stream_executor/cuda/cuda_driver.cc:269] failed call to cuInit: UNKNOWN ER
OR (303)
2022-01-24 14:07:27.433270: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:169] retrieving CUDA diagnostic i
formation for host: DESKTOP-DCDSI2R
2022-01-24 14:07:27.453292: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:176] hostname: DESKTOP-DCDSI2R
2022-01-24 14:07:27.465570: I tensorflow/core/platform/cpu_feature_guard.cc:151] This TensorFlow binary is optimize
 with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical oper
ations:  AVX AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
test_frames\1519.mp4
image1.jpg        Accident
image2.jpg        Accident
image3.jpg        Accident
image4.jpg        Accident
PS C:\Users\RISHI\Desktop\project\accident_detection>
```

Fig 6.4 : No of Accident occurred

In this picture we got 4 images displayed with name accident , because in that video 4 accidents has happened so we got displayed that.

CHAPTER – 7

7.1 CONCLUSION

Accident detection operation is not an easy task to handle;it can be an extremely complicated process when it comes toreal time applications, which is the main reason why it is notimplemented yet on a large scale. The proposed system will help to improve the present scenarios.

7.2 FUTURE WORK

The proposed system deals with the detection of the accidents. But this can be extended by providing medication to the victims at the accident spot. By increasing the technology we can also avoid accidents by providing alerts systems that can stop the vehicle to overcome the accidents.

7.3 SCOPE OF FURTHER STUDY

The proposed system deals with the detection of the accidents. But this can be extended by providing medication to the victims at the accident spot. By increasing the technology we can also avoid accidents by providing alerts systems that can stop the vehicle to overcome the accidents.

CHAPTER - 8

REFERENCE

1. Kardas K, Cicekli NK. SVAS: surveillance video analysis system. Expert Syst Appl. 2017;89:343–61.

2. Wang Y, Shuai Y, Zhu Y, Zhang J. An P Jointly learning perceptually heterogeneous features for blind 3D video quality assessment. Neurocomputing. 2019;332:298–304 (ISSN 0925-2312).

3.Tzelepis C, Galanopoulos D, Mezaris V, Patras I. Learning to detect video events from zero or very few video examples. Image Vis Comput. 2016;53:35–44 (ISSN 0262-8856).

4. P.A. Viola and M.J. Jones, ―Rapid object detection using a boosted cascade of simple features,‖ in Proc. CVPR, 2001, no. 1, pp. 511–518.

5.“Global status report on road safety 2015”, World Health Organization,2016. [Online]. Available: http://www.who.int/violence_injury_prevention/road_safety_status/2015/en/. Accessed: 22- Mar- 2016

6. V. Goud, “Vehicle accident automatic detection and remote alarm device,” International Journal of Reconfigurable and Embedded Systems (IJRES), vol. 1, no. 2, Jul. 2012.

7. B. Prachi, D. Kasturi and C. Priyanka, ―Intelligent accident-detection and ambulance-rescue system,‖ PULSE, vol. 450, no. 16, pp. 2, Jun. 2014.

8. G. Liang, “Automatic traffic accident detection based on the Internet Things and Support Vector Machine”, International Journal of Smart Home (IJSH), vol. 9, no. 4, pp. 97-106, Apr. 2015.

9. C.Vidya Lakshmi, J.R.Balakrishnan, ―Automatic Accident Detection via Embedded GSM message interface with Sensor Technology ,International Journal of Scientific and Research Publications, vol. 2, no. 4, Apr 2012.

10. X. Tong, Z. Hong, S. Liu, X. Zhang, H. Xie, Z. Li, S. Yang, W. Wang and F. Bao, ―Building-damage detection using pre-and post-seismic high-resolution satellite stereo imagery: a case study of the May 2008 Wenchuan earthquake,‖ ISPRS Journal of Photogrammetry and Remote.Sensingg, vol. 68, pp.13-27, Mar. 2012.