A MAJOR PROJECT REPORT

on

# "VEHICLE DETECTION AND SPEED ESTIMATION USING DEEP LEARNING"

*Submitted in partial fulfillment of the requirements for the award of*

**BACHELOR OF ENGINEERING**

***in***

**ELECTRICAL AND ELECTRONICS ENGINEERING**

***of***

**Visvesvaraya Technological University, Belagavi**

**Submitted by**


| | |
|---|---|
| **Anusha** | 1SI18EE005 |
| **Nidhi T** | 1SI18EE028 |
| **Saniya Kouser** | 1SI18EE046 |
| **Vyshnavi G L** | 1SI18EE059 |

Carried out at

**Siddaganga Institute of Technology**

**Tumakuru**

Under the Guidance of

**B.A. Sridhara M.E**

**Associate Professor**

**Department of Electrical and Electronics**

**Siddaganga Institute of Technology**

**Tumakuru-572103**



**DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING**

**SIDDAGANGA INSTITUTE OF TECHNOLOGY**

***(An Autonomous Institute Affiliated to Visvesvaraya Technological University, Belagavi, Recognized by AICTE and Accredited by NBA, New Delhi)***

**TUMAKURU-572103, KARNATAKA, INDIA**


**2021-2022**

# SIDDAGANGA INSTITUTE OF TECHNOLOGY

***(An Autonomous Institute Affiliated to Visvesvaraya Technological University, Belagavi, Recognized by AICTE and Accredited by NBA, New Delhi)***

**DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING**

# CERTIFICATE

Certified that the project work entitled **“VEHICLE DETECTION AND SPEED ESTIMATION USING DEEP LEARNING”** is a bonafide work carried out by **Anusha USN:1SI18EE005, Nidhi.T USN:1SI18EE028, Saniya Kouser USN:1SI18EE046, Vyshnavi G L USN:1SI18EE059** in partial fulfillment of the requirement for the award of the degree of **Bachelor of Engineering** in **Electrical and Electronics Engineering** of **Siddaganga Institute of Technology, Tumakuru,** an autonomous institute under Visvesvaraya Technological University, Belagavi during the academic year **2021-2022**. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said Degree.

| | | |
|---|---|---|
| **B.A. Sridhara M.E** | **Dr. Rashmi** | **Dr. S.V. Dinesh** |
| Associate Professor<br>Dept of E&E<br>SIT,Tumakuru-572103 | Professor&Head<br>Dept of E&E<br>SIT,Tumakuru-572103 | Principal<br>SIT,Tumakuru-572103 |

**External Viva-Voce:**

| **Name of the Examiners** | **Signature with date** |
|---|---|
| 1. ____________________ | ______________________ |
| 2. ____________________ | ______________________ |

# SIDDAGANGA INSTITUTE OF TECHNOLOGY

***(An Autonomous Institute Affiliated to Visvesvaraya Technological University, Belagavi, Recognized by AICTE and Accredited by NBA, New Delhi)***

**DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING**

## UNDERTAKING

We **Anusha, Nidhi T, Saniya kouser and Vyshnavi G L** hereby declare that the project work entitled **"VEHICLE DETECTION AND SPEED ESTIMATION USING DEEP LEARNING"** has been independently carried out by us under the supervision of **B.A Sridhara ,** Designation, Dept. of Electrical and Electronics, Siddaganga Institute of Technology, Tumakuru and submitted in partial fulfilment of the requirements for the award of the degree of **Bachelor of Engineering** in **Electrical and Electronics Engineering** of **Siddaganga Institute of Technology, Tumakuru,** an autonomous institute under **Visvesvaraya Technological University, Belagavi** during the academic year **2021-2022**.

This work has not been submitted in part or full for the award of any other Degree/Diploma in this University or any other university.

**Place: Tumakuru**
**Date: 28-07-2022**

| | |
|---|---|
| **Anusha** | 1SI18EE005 |
| **Nidhi T** | 1SI18EE028 |
| **Saniya Kouser** | 1SI18EE046 |
| **Vyshnavi G L** | 1SI18EE059 |

# ACKNOWLEDGMENT

"The satisfaction euphoric that accompany the success of any work would be incomplete unless we mention the name of the people who made it possible, whose constant suggestions, guidance, encouragement helped us in completing our project work and served our effort with success".

We express my deep sense of gratitude and respectful 'Pranams' at the lotus feet of **Dr. Sree Sree Sivakumara Swamigalu,** Revered Founder President and **Sree Sree Siddalinga Swamigalu,** Revered President, Siddaganga Education Society, for their blessings and for providing us with an opportunity to fulfill our most cherishable desires of reaching our goal.

We would like to express our profound gratitude to **Dr. M. N. Channabasappa**, Director, SIT, Tumakuru and **Dr. S.V.Dinesh,** Principal, SIT, Tumakuru, for giving all the facilities to pursue the Bachelor's degree successfully.

We would like to thank **Dr. Rashmi,** Professor and Head, Department of E&EE, SIT, Tumakuru for his kind consent and whole hearted co-operation.

We take this opportunity to express our sincere gratitude to our guide **B.A. Sridhara ,** Designation, Department of E&EE, SIT, Tumakuru for his/her kind support, guidance and encouragement throughout the course of this work. It was a great privilege and honour to work and study under his/her guidance.

We sincerely thank all the faculty members of E & E Engineering department for their support and help in enriching my knowledge. Also, we record our gratitude to the non-teaching staff of the department.

Finally, we would like to present our deepest gratitude to our parents and friends for their moral support and their encouragement, which motivated towards successful completion of our project.

**Anusha**
**Nidhi T**
**Saniya Kouser**
**Vyshnavi G L**

# ABSTRACT

Intelligent traffic control and surveillance are the basic requirements for developing a smart city in India. As urban traffic problems increase, video-based traffic surveillance systems have become popular over the last few decades. These systems are extremely useful for monitoring and managing a variety of traffic conditions, including traffic management, accident prevention and safe transportation. Within that range, one of the purposes is to measure vehicle speed on roads. Determining the speed of the vehicle thus becomes a challenging task. In determining the speed of vehicle detection and vehicle tracking are key steps. To overcome disadvantages of the traditional RADAR/LIDAR technology here we are using the method of determining vehicle speed using image processing. This includes detecting moving vehicles, estimating their speed and detecting them about exceeding the speed limit. This project detects and tracks vehicle passing through the tracking area, vehicle tracking is based on relative positions of vehicles in consecutive frames and keeps records of vehicle positions and red marks vehicles that exceed the speed limit and alerts the traffic manager.

The fundamental requirement for the development of India's smart cities is an intelligent system of traffic control and surveillance. Traffic monitoring systems based on video have gained popularity in recent years as urban traffic issues worsen. These systems are particularly helpful for managing various traffic conditions, such as traffic management, accident prevention, and safe transportation, with one of the goals being to determine vehicle speed on the road. Therefore, determining a vehicle's speed has become a difficult process. Vehicle identification and vehicle tracking are the crucial steps in speed determination. Here, image processing is used to determine vehicle speed in order to overcome the drawbacks of the old approach. Detecting moving cars, determining their speed, and identifying speed limit violations are all included in this. The method described in this study uses a single camera in a well-lit environment to detect moving vehicles and estimate their speeds. It is both effective and innovative. The suggested method identifies and tracks any vehicle that passes through the monitoring area. The tracking of the vehicles is based on their relative positions in successive frames, and a record of their positions is kept. Additionally, it warns the traffic manager and highlights in red any vehicles travelling at excessive speeds.

# INDEX

## List of figures:

## Chapter-1

# INTRODUCTION

### 1.1 General Overview

In India an intelligent traffic management and surveillance is the basic need for the smart city. In the current years we are able to see there's a huge growth in the range of vehicles all around the globe. Along with the growth in range of vehicles will increase the number of accidents. Therefore, it's far critical to restrict the speed of the vehicles at certain zones or areas. Initially a Doppler radar was accustomed capture the speed of Vehicles on roads. Cameras were used to capture rushing vehicles. A Doppler radar uses the modification in frequency of a passing vehicle to see the speed of the vehicle. This development is thought as Doppler effect. Later Traffic Radars started using sensors to find speeds. At present, traffic radars/lidars integrate the utilization of sensors and image processing to catch traffic violators. The traffic violation needn't be rushing, it can even be reserved/wrong lane detection, passing red-light detection, tailgating, wrong facet reordering etc. In the future, traffic radars could even observe seatbelts and texting whereas driving offences. Such work is presently below analysis and will commence shortly. modern speed camera has advanced tracking analysis with video recording and online streaming capabilities, tracking of multiple vehicles at same time, auto number plate detection, it can detect tailgating, and vehicle overtaking on the wrong side, with real time data transmission by using sensor. This project mainly focuses on speed determination of vehicle with pre-recorded video

### 1.2 Objective

The objective of this project is to detect the speed using Image Processing in Python by using OpenCV and TensorFlow to provide an efficient system to detect and track vehicles using image processing and vehicle speed estimation methods to make roads safer with fast-growing technology and to build intelligent traffic systems. is to do Basic requirements for smart city improvement and traffic accident prevention.

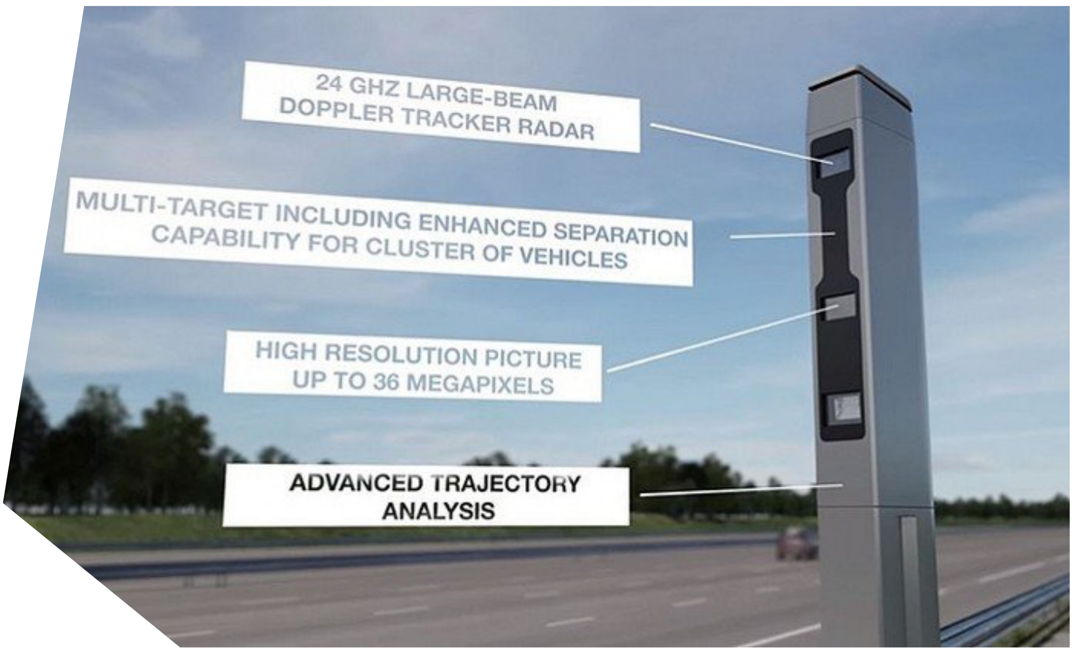



Figure 1: Modern Speed Camera

# CHAPTER-2

## 2.1 Problem statement

Increased car use has increased traffic on highways. With the increasing speed of automobiles today, speed sensing has become a major issue in preventing fatal accidents. Radar technology is a modern and productive way to determine speed. Radar speed measurements are commonly used for this purpose and may not be accurate in some cases, such as detecting small vehicles with weak echoes. In addition, these tools are difficult to detect vehicles that change speed too often or too quickly. In this project, we came up with an alternative image and video processing method that could overcome the shortcomings of radar guns.


Figure 2.1: Traffic and accidents caused by over speed

## 2.2 Solution

Vehicle detection, tracking and speed estimation using Image Processing Technique and Deep Learning gives appropriate solution to overcome the advantages of traditional methods. These include detecting moving vehicles, estimating speed, and detecting speed limit violation. The proposed approach captures and tracks vehicles passing through the surveillance area. Vehicle tracking is based on the relative position of the vehicle within a continuous frame and saves the vehicle position. And it indicates those over speed vehicles in red colour and alerts the traffic manager.

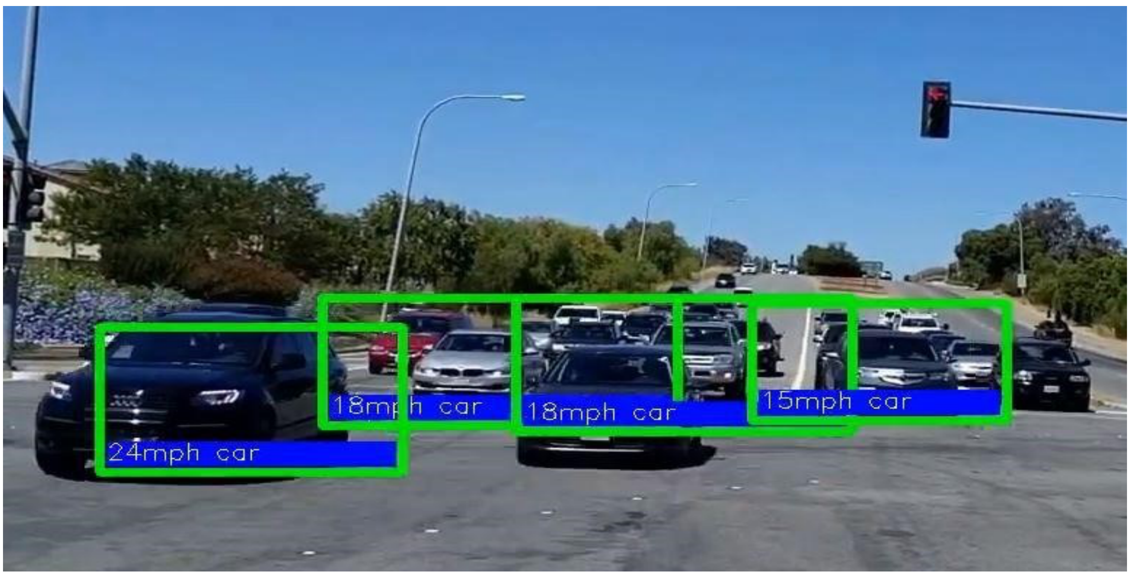



Figure 2.2: Estimating speed of the vehicle

## Chapter-3

# LITERATURE SURVEY

1. **Alexander Grents, Vitalii Varkentin, Nikolay Goryaev** in the year 2020 proposed Determining vehicle speed based on video using convolutional Neural Network with datasheet considered 750 images from 52,000 objects. The images were collected from traffic cameras of Intersvyaz in the city named Chelyabinsk. Categories like cars, buses, trolleybus, trucks, , ambulances, fire-trucks, vans, uncategorized Mask R-CNN architecture is used for vehicle detection. Masking was used on the road video so as to recognise vehicle accessible regions. A rectangular region is selected, and speed of a vehicle crossing the region is calculated with the help of equations. It is a straightforward method but The accuracy of this radar depends of the video camera resolution and frame rate.

2. **Alexander A S , Deasy Aprilia T Fergyantoo E. Gunawan** in year 2019 proposed Detection of Vehicle Position and Speed using Camera Calibration and Image Projection Methods Direct Linear Transformation is used for estimating the length of a segment of road. Vehicle position is detected by usage of Background Subtraction and speed is determined by Mixture of Gaussian (MoG) ,Camera Calibration is used to calculate the target position of vehicle in 3D space. were used.it is High Accuracy (96% claimed) Length of segment of road is calculated through image. No need for manual measurements. But Best result is when camera is on top of the road in future it can be used for multiple detection and tailgating can be monitored

3. **Saleh Jawadi, Mattias Dahl, and Mats I. Pettersson** in year 2019 proposed Vehicle speed measurement model for a video-based systems Intrusion detection is used when an object passes through a virtual line and enter in a region of interest(ROI). The detection distance is directly proportional to the vehicle’s speed and it is inversely proportional to the camera’s frame rate. It is a simple idea to implement. but Does not classify vehicles Best result is only when camera is on top of the road In future tailgating can be monitored and wrong lane can be detected

4. **Tarun Kumar and Dharmendaar** in the year 2016 proposed An Efficient Approach for Detection and Speed Estimation of Moving Vehicles , Firstly Image acquisition and preprocessing is done. then background is modelled. By using Camera calibration to map the relationship with real world.Dark shaded area under vehicle is known as the region of interest. It claims to have a high accuracy (87 %--98%).but the Accuracy varies during time of day .Mostly it would be ineffective at time of night S = d*f/n.IN future Classification of vehicles and the Recording of number-plates details can be done.

5. **Manne Sai Sravan, Sudha Natarajan, sai Krishna, Binsu J Kailath** in 2018 proposed Fast and accurate on-road vehicle detection based on colour intensity segregation This uses lane tracking and ROI extraction. A custom vehicle filter consisting of erosion and dilation is made .It does not need a dataset of vehicles to detect but No classification of vehicles .It cannot be used for multiple vehicles. In future it can be used for detecting wrong lane

6. **Alexander A S Gunawana** in the year 2019 implemented Direct Linear Transformation (DLT)approach for projection of road image , and object position recognition by using Background subtraction and tracking with the help of Mixture of Gaussian (MoG) to assess the object speed. Here Euclidean distance is used. Obiect position detection with ground fact is 12.07 pixels with an angle of 40Â° as camera point. But it can't be carried out for detecting multi- object .

7. **Danang Wahyuu Wicaksono** (2017) applied the Euclidean distance along with extensive variety of camera plots for estimation of vehicle speed. This match on calculating speed of the moving automobile . But the very best in moving automobile speed estimation in this approach relies upon on vehicle speed determination on high-speed cars and objects are near each other might be recognized as one vehicle

8. **Suresh** (2016) carried out a way at the Estimation of speed for vehicle in a motion using Image Processing . This method is applied for estimating car speed by tracking motion of the automobile. The speed dedication is performed by considering length traveled by the vehicle over the rate of frame and the number of frames. But its now no longer deals in detecting multi-vehicle at a time; the angle of camera isnot determined.

9. **Benjamin Penchas, Tobin Bell, Marco Monteiro** in the year 2017 implemented a method of determaining the speed of vehicle in real time given a mounted dashboard video stream by using a deep neural network. They explained how neural networks are implemented and work in speed estimation.

# CHAPTER-4

# TECHNOLOGIES USED

## 4.1 MACHINE LEARNING

Machine learning is a subdivision of Artificial Intelligence (AI) and computer technological way which is used to make specialty of using facts and algorithm to copy the method of people learn and increasing the accuracy using step by step procedure.
Machine Learning is the fundamental feature to present the way of developing area of facts technologically. Algorithms are made to classify or to predict using statistical methods by not covering key insights inside project. These key insights power the selection making inside the packages and business. Here, impacting key is commonly used to increase the advantages. As massive facts keep extending and grow, the marketplace region for fact's scientists get increase, using them to help with inside the identity of the maximum applicable enterprise questions. Finally, the facts are used to reply to them.

## How machine learning works

1. Decision process: Generally, Device mastering algorithms are used for the prediction or classification. Based on few records which may be labelled or may not be labelled, our set of rules will produce an estimation of approximately a sample inside the records.
2. Error Function: A blunders characteristic helps to assess the prediction of the version. The acknowledged examples of this version and the blunders characteristic are involved in assessment to evaluate the accuracy of this version.
3. Model Optimizing Process: If the version suits the factors of the schooling set at maximum, then the parameters ore adjusted to decrease the difference among the acknowledge instance and estimation of the version. The set of rules will continue their tasks of examine and optimization process and the parameters has been updated autonomously till they reach the threshold accuracy.

## Machine learning methods

Machine learning has mainly three primary categories.

### 1.Supervised system studying

Supervised studying is described through its application of categorized datasets to educate algorithms for the purpose of categorize facts or expect effects accurately. This system adjusts its parameters till the version has been configured appropriately by entering the facts to fed into this version. This process undergoes validation manner to make certain work to avoid overfitting and underfitting. Supervised studying facilitates corporation clear up for quite a few real-international issues at scale, including classifying uninvited mail in a distinct folder out of inbox. In Supervised studying encompass neural network encompass neural networks, some strategies are utilized namely linear regression, logistic regression, naïve Bayes random forest, guide vector system (SVM) and more.

**2.Unsupervised system studying**

Unsupervised studying is used to investigate and collect the not labelled datasets by using system studying algorithms. These algorithms find out hidden styles or facts groupings without the need of human involvement. Its function is to find the similarities and differences. It is best answer for exploratory facts analysis, photograph, go-promoting strategies, sample recognition and client segmentation. It is importantly utilized to lessen the range of proficiency in a version through the manner of reduction in dimensionality, Predominant factor analysis (PFA) and Singular Price Decomposition (SPD) are not usual place strategies for this system studying. Other algorithms are utilized in unsupervised studying encompass neural networks namely K-approach clustering, probabilistic clustering strategies and more.

**3.Semi-supervised studying**

This gives the better performance and most satisfied method among supervised and unsupervised studying. This uses a smaller categorized facts set to manual type and function extraction from a larger and not labelled facts set during the training. Semi-supervised studying can clear up the trouble of getting now no longer sufficient categorized facts (or now no longer being capable of come up with the money for to label sufficient facts) to educate a supervised studying algorithm.

**Reinforcement learning**:

Reinforcement learning: is the most used technology by Data scientists to train the system to finish a multistep technique for which there are truly described rules. The algorithm finishes a mission and deliver programmed by Data Scientists which is High quality or poor. Because it works out how to finish a task. But for the maximum part, algorithm decides on its very own what steps to take alongside the way.

## 4.2 Deep learning

Deep learning is an improved version of Machine learning. Basically, Deep learning is a neural network which contains three or more layers. The neural networks in deep learning try to reproduce the characteristic of the human brain to do a specific task. This will be possible from comparing the ability permitting it to learn and adopt them from large quantities of data. Even though, a neural network with a single layer can still produce approximate predictions, the extra adding layers can assist to optimize and helps to improve the quality and accuracy of the predictions.
Deep learning is using in many manmade Artificial intelligence (AI) applications and offences which enhance the ability of automation and appearing in to do the physical and analytical tasks: without human involvement. Deep learning span situated at the back of ordinary merchandise and all springs (which includes voice enabled TV remotes, virtual assistants, and creditors card fraud detection) in addition to rising the technologies (Which includes self-riding cars).

### Working:

Artificial neural networks or Deep learning neural networks tries to imitate the human mind via a mixture of bias, parameters, and data inputs. These factors work collectively to correctly identify, organize, and describe objects inside the data.

Deep learning neural networks encompass more than one layers which are interconnected as required, every developing upon the already existing layer to verify, refine and advance the prediction or categorization. Ahead propagation is referred by this development of computations via network. The visible layer in the deep neural network is enter and output layers of it. The input layer is located at the ingestion of data in the deep learning model for processing. And the guput layer is located at the very last classification or prediction is made.

Backpropagation is another process which is used to make use of algorithms. For example, like gradient descent us used to estimate the errors in the prediction. After that, it balances the biases and parameters of the feature with the help of backward shifting via the layers with the intention to upskill the version. Simultaneously, Backpropagation and ahead propagation permit a Neural community to make conclusions or estimations and accurate for mistakes with respect. By taking the extra time to do the work efficiently, the accuracy get increases in step by step.

The above-mentioned points describe the only type of deep Neural network. Since deep learning algorithms are tremendously complex, there are special kinds of neural networks to deal with datasets or issues. Ex: CNNs and RNNs.

**Convolutional neural networks (CNNs**) are used in most cases in image classification applications and computer vision to detect patterns and features within the picture which allow tasks like object recognition and detection. The record was created that a CNN performed highly efficiently by challenging the human in an object detection or recognition challenge in 2015.
**Recurrent neural network (RNNs)** is generally utilized in speech recognition applications and natural language because it supports times series or sequential data.

## 4.3 Image processing:

Image processing is the process of turning an image into a digital shape and applying specific operations to it in order to extract some useful information. The image processing system typically uses a set of predetermined signal processing techniques and treats all images as 2D signals.

The following 5 categories of image processing exist:
1. Visualization: Look for objects inside the image that can't be seen.
2. Recognition: Detect or distinguish objects inside the image.
3. Sharpening and restoration: From the original image, produce an improved version.
4. Recognizing patterns - Count the many patterns on the objects in the image.
5. Retrieval - Browse and search through a vast database of digital photos that may be identical to the original image

**Fundamental Image Processing Steps**

1. **Image Acquisition**
   The initial stage of image processing is image acquisition. In image processing, this is often referred to as pre-processing. The image must be retrieved from a source, which is typically a hardware-based source.

2. **Image Enhancement**
   Image enhancement is the process of bringing out and emphasising particular elements of a hobby in a blurry image. This might involve changing the contrast or brightness, for example.

3. **Image Restoration**
   The process of improving an image's appearance is called image restoration. Picture restoration, as opposed to image augmentation, is carried out utilising specific mathematical or probabilistic models.

4. **Colour Image Processing**
   Some digital colour modelling techniques are used in colour picture processing. Because digital photos are widely used on the internet, this step has become more important.

5. **Wavelets and Multiresolution Processing**
   Wavelets are used to represent images at different resolution levels. For the purposes of pyramidal representation and data compression, the images are separated into wavelets or smaller sections.

6. **Compression**
   Compression is a technique used to reduce the amount of space needed to store or the bandwidth needed to transmit an image. This is done mostly when the photograph will be used online.

7. **Morphological Processing**
   A collection of processing methods known as "morphological processing" are used to transform images only based on their forms.

8. **Segmentation**
   One of the most challenging aspects of image processing is segmentation. It involves dividing an image into its individual objects or components.

9. **Representation and Description**
   Every area is represented and specified in a way that is suitable for subsequent computer processing once an image has been divided into areas through the segmentation process. The attributes of the image and the properties close are offered as representation. Extracting quantitative data to distinguish between distinct object classes is the focus of description.

10. **Recognition**
    Recognition gives an object a label only based on its description.

## Working of Machine Learning Image Processing

Machine learning algorithms frequently follow a specific pipeline or set of steps when learning from data. An operational algorithm for an Image Processing use case using a generic example of the same. First and foremost, ML algorithms need a lot of high-quality data to analyse and forecast outcomes with a fair amount of accuracy. Because of this, we must ensure that the photos are correctly edited, tagged, and generic for ML image processing. This is where Computer Vision (CV), a field concerned with machines being able to understand the image data, enters the picture. With the help of CV, we manipulate, process, load, transform, and manage photos to provide the best dataset possible for the machine learning algorithm. A typical machine learning workflow for processing images is shown.

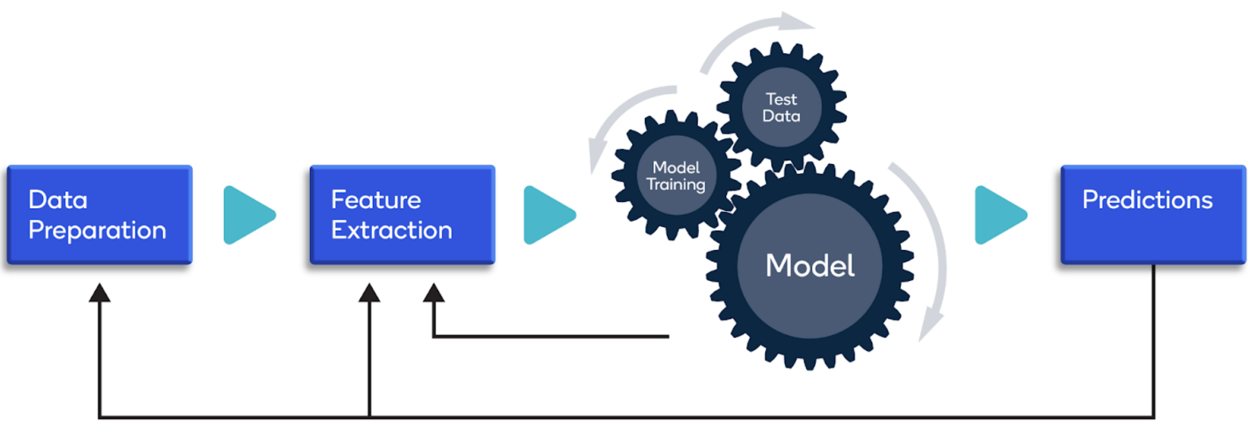



Figure 4.1 : Image processing workflow

Example: we need to develop an algorithm that can determine whether a particular image contains a dog or a cat. We'll need to collect photographs of cats and dogs for this and pre-process them with CV. The pre-processing actions consist of:

1. Transforming all the photographs to the same format.
2. Removing the useless portions of image crops.
3. Changing them to numbers so that algorithms can use them to learn from (array of numbers)

Computers rely on the image resolution and interpret input images as arrays of pixels. It will see (height * width * dimension) depending on the image resolution. For instance, a picture of an RGB matrix with a 6 × 6 x 3 array (where 3 stands for the RGB values) and a picture of a grayscale matrix with a 4 x 4 x 1 array. The next phase uses these features (the facts that have been processed) to choose and build a machine-learning algorithm to categorise novel feature vectors given a large library of feature vectors with established classifications. We need to choose the best method for this, and some of the most well-known ones are Bayesian Nets, Decision Trees, Genetic Algorithms, Nearest Neighbors, and Neural Nets, among others.

## 4.4 Python language

Python is a popular computer programming language for building websites and software, automating processes, and performing data analysis. Since Python is a general-purpose language, it may be utilised to develop a wide range of unique applications and is not specialised for any issues. Its versatility, together with its beginner-friendliness, has elevated it to the top of today's most popular programming languages. According to a survey conducted by industry research firm red Monk, it rose to the second-most popular programming language among developers in 2021.


Figure 4.2 : pip Installation

**PYTHON is used in:**

- Data analysis and machine learning
- Web development
- Automation or scripting
- Software testing and prototyping
- Routine activities

**Installation**

- **Step 1:** From python.org download Python binaries
- **Step 2:** Install binaries
- **Step 3:** Add Python to system env variables
- **Step 4:** Install pip
- **Step 5:** By using pip install virtual env

## Chapter-5

# SOFTWARE DESCRIPTION

## 5.1 VISUAL STUDIO

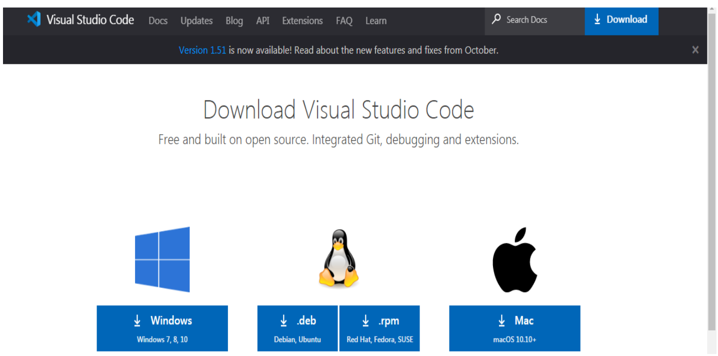
In layman's terms, Visual Studio Code is a code editor. It is "a free editor that allows programmers to write code, allows for debugging, and allows for the use of the IntelliJ experience to correct the code." In ordinary terms, it enables users to write down the code easily. Many human beings say that it's far 1/2 of an IDE and an editor, however the choice is as much as to the coders. Any program/software program that we see, or use works at the code that runs withinside the background. Traditionally coding become used to do withinside the conventional editors or maybe withinside the simple editors like notepad! These editors used to offer simple help to the coders.

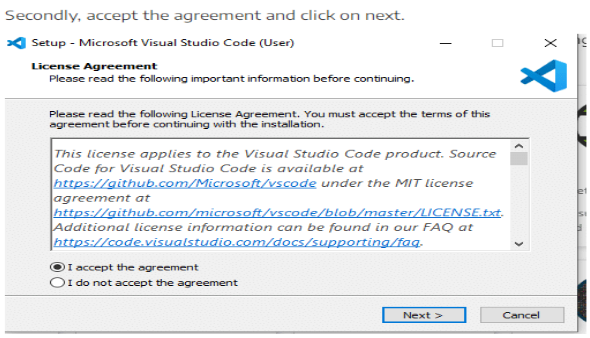
Some of them have been so simple that it become very hard in writing simple English stage applications in them. As time went by, a few programming languages wished a selected framework and help for in addition coding and improvement it, which become now no longer feasible the use of them. VI Editor, Sublime Text Editor, is one of the many varieties of editors that got here into existence. The maximum distinguished and which helps nearly each coding language is VISUAL STUDIO CODE. Its functions permit the consumer adjust the editor as consistent with the usage, because of this that the consumer is capable of download the libraries from the net and combine it with the code as consistent with his requirements.


Figure 5.3 : Visual studio code

**Installation**

Step 1: By using Link download VS code.
Step 2: Download the Windows setup for Visual Studio Code. After downloading, launch the installer. (VS Code User Setup-{version}.exe). and, run the file.



Accept and click 'Next'

After accepting all the requests click on finish button. By default, VS Code installs under: **"C:\users{username}\AppData\Local\Programs\Microsoft VS Code."** after installation is successful, we see the following.

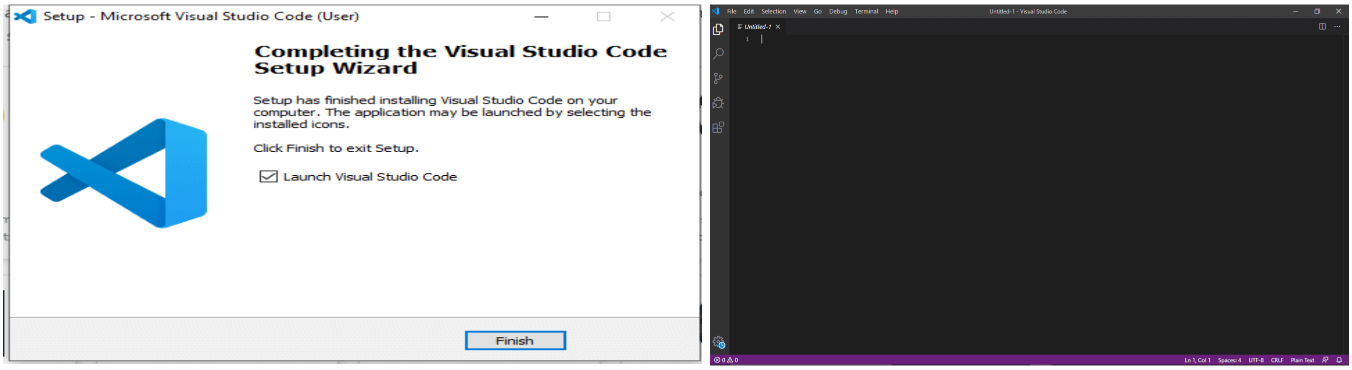



Figure 5.2 : Visual studio installation

## 5.2 **Libraries**

### 5.2.1 Open cv

OpenCV is an open-source library which could be very beneficial for pc imaginative and prescient packages along with video evaluation, CCTV photos evaluation and picture evaluation. More than 2,500 efficient algorithms make up OpenCV, which is written in C++. We can utilise this library as a starting point for developing computer vision applications so that we don't have to start from scratch. These applications will focus on real-world situations. These days, numerous companies, including Google, Amazon, Microsoft, and Toyota, use this library. Many academics and architects participate. We can easily deploy it on any OS, including Windows, Ubuntu, and MacOS.


Figure 5.3 : OpenCV

Step 1- Download vs code
Step 2- Install pre-built OpenCV package by using OpenCV-MinGW-Build.

**OpenCV 4.1.1-x64 | zip | tar.gz**

```
git clone -b OpenCV-4.1.1-x64 git@github.com:huihut/OpenCV-MinGW-Build.git
```

▼ Configuration

- MinGW-x86_64-8.1.0-posix-seh-rt_v6-rev0
- Windows-10-64bit
- CMake-3.14.1

Step 3: Add the Pre-built OpenCV Packages and Compiler Path to **Path** Variable

Set the Path variable of your account with the binaries for the compiler and OpenCV package to let other programmes to find the names of the packages without having to specify the absolute path.
Here is an illustration of how environment variables are set:

Step 4: Configure the Code Editor files to the project.

### 5.2.2- NumPy

A Python library for working with arrays is called NumPy.
Additionally, it contains capabilities for working with matrices, the Fourier transform, and linear algebra.
The development of NumPy began in 2005 with the help of Travis Oliphant. It is available to use freely and is an open supply challenge. Numerical Python is referred to as NumPy.

**Installation :**

Figure 5.4 : NumPy

If we have already installed on a system, then installation of NumPy is easy.
Install it using the command below
NumPy is simple to install if Python and PIP are already set up on a machine.
Install it by executing the following command.

```
C:\Users\name>pip install NumPy
```

After installation, import it into programmes by including the phrase import:

```
import NumPy
```

## Chapter-6

# METHODOLOGY

A video stream is recorded using a camera. Here, we are utilising a sample-based background subtraction technique that simulates the background and continuously updates the background model to detect moving cars. The model parameters are better suited to describe the scene in real time when the backdrop model is updated. The samples used for training are updated using trackers. Because existing trackers can only track one object at a time, we use multi-threading technology to create an effective multi-object management module. For analysing and determining the tracked vehicle's status, this module can allocate numerous objects tracking jobs to parallel blocks. Finally, to determine the total number of vehicles, we use the tracked vehicle's state.

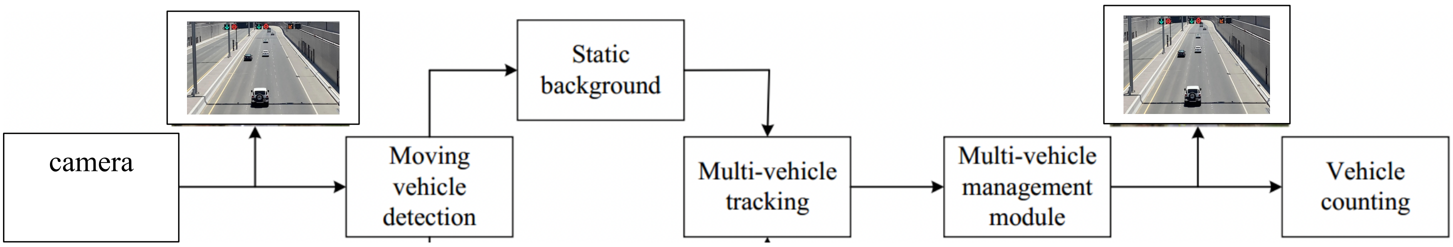



Figure 6.1 : Framework of the model

### 6.1 Vehicle Detection :

Vehicle detection is a crucial step in the vehicle counting process. Background modelling can be used to extract moving objects from a static background. The usage of ViBe for vehicle detection provides the following benefits.

1. The backdrop model can be updated via the ViBe foreground detection technique. We can successfully suppress noisy points in the image caused by minute brightness variations by changing the backdrop model.

2. ViBe chooses a specified area in the image for background modelling before modelling the complete image. The computational load is considerably reduced as a result.

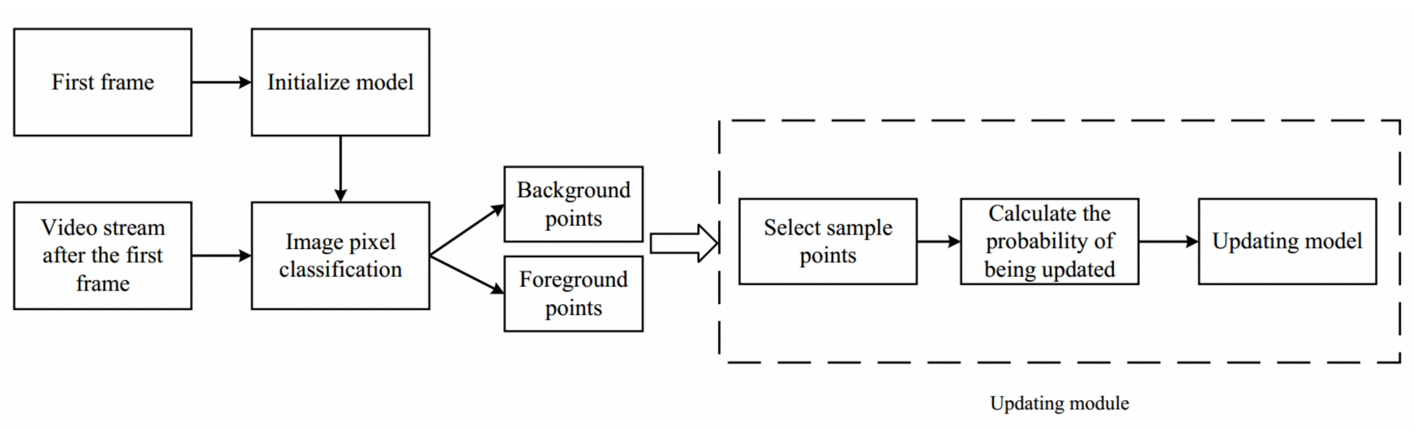



Figure 6.2: The overview of ViBe algorithm.

Initializing the background is the first stage in the ViBe process. N background samples [v1, v2, ..vN] are used to model each backdrop pixel. Choose a sample of nearby pixel values at random for modelling. The difference D between the pixel values in the field cantered on the point v (x) is defined to categorise the pixel v (x).

$$D = |v(x) - vi| \quad (1)$$

D for an RGB visual,

$$D = |vr(x)| + |vg(x)| + |vg(x)| + |vb(x)| \quad (2)$$

Value for the background sample in equation (1), vi. The centre pixels in equation (2) are vr(x), vg(x), and vb (x). Three channels' worth of background sample values are denoted as vri, vgi, and vbi. Pixel difference minimum threshold is Dt. S and St - number of points above the minimum threshold for pixel difference and minimum value of S

To analyse the algorithm's fundamentals, we consider looking at a grayscale image as an example. Three different classification parameters for pixels are defined here. The point vx is labelled as background if S > St. The first stage in ViBe is to initialise the background using a video stream as input. The method starts extracting foreground at the second frame after initialising the background model in the first frame. The chance of updating the model is calculated after randomly selecting sample points.An updated model method is used to enhance the detection performance of moving objects in changing backgrounds. The likelihood of updating each background point is 1ϕ. 1ϕ is the likelihood of updating neighbouring points.
Utilizing the spatial propagation properties of the image pixels, the neighbor's sample image values are updated. The backdrop model then progressively spreads outside. The likelihood of updating the background model is 1 if a pixel is determined to be a background point. Three phases make up the changing process in general: randomly choosing the sample replace, randomly choosing whether to update the backdrop model, and randomly choosing whether to update the field pixels.

To enhance the performance of detecting moving objects against changing backdrops, model updating techniques are applied. The likelihood that each background point will be updated using this method is 1. There is a 1004 percent chance of updating a neighbor's points. The spatial diffusion attribute of the pixel value is used to update the sample pixel value of a neighbour. The background model then gradually expands outward. The likelihood of updating the background model is 1 if it is found that the pixel is the reference point. The updating procedure generally consists of three steps:

a. Pick a sample update at random
b. Pick at random whether or not to update the backdrop model.
c. Choose at random whether to modify the field pixels.

## 6.2 Vehicle tracking:

KCF moves along much more quickly over other detection tracking techniques like TLD & STRUCK. Tracker behaviour is impacted by vehicle multiscale and form changes because of the complicated ground conditions in UAV recordings. Take on this issue with the online learning tracker. It trains the detector in the tracking process and views the tracking process as a regression based problem. The detector is used to determine where an object will be in the following frame. Examples of inputs used during training include samples and labels like (x1, y1), (x2, y2), .. (xn, yn). The label value yi, which is the number [0, 1], can be found by computing the distance between the centre of the object and the centre of the sample. Yi tends to be 1 if the sample is near an item, but 0 otherwise. The training's objective is to become familiar with the function f (z) = w T z, z = [z1,.., Zn]. By doing this, the sample's root-mean-squared error is reduced.

$$\min w \sum i\ (f(xi) - yi)\ 2 + \lambda kwk\ 2 \qquad (6)$$

where λ – controls over fitting known as regularization parameter

The following are the main steps that make up the KCF tracking procedure. The response of the tiny window sample is determined before the classifier is trained using the tracker sample that was chosen close to the prior location Pt for frame t. The response of the each sample is then determined by the classifier learned at frame t by taking samples close to the position Pt of the previous frame at frame t + 1. The sample reacts most strongly to the expected location Pt + 1. Figure 6.3 demonstrates that during the tracking process, we train the regression by locating a sample close to the object at frame t and using regression to predict the displacement of the object monitored at frame t + 1. The dashed red box in frame t is the first tracking box. (A) The object's state at frame t. (B) The object's condition at frame t + 1. A checkmark indicates that box #1 receives the most feedback.
This is 2.5 times larger as a prediction box (blue). The pattern box obtained after circling the blue box is the black box surrounding the object. To train your classifier, are using these sample boxes. We first receive a sample in the prediction area at frame t + 1. H. It is in the vicinity of the solid blue box. The responses to these boxes can then be calculated using the classifier. Of course, field #1 receives the most feedback. As a result, you can anticipate the object's displacement. The Gray feature and the HOG feature are the two choices for extracting an object's features, respectively. The HOG feature is employed here.

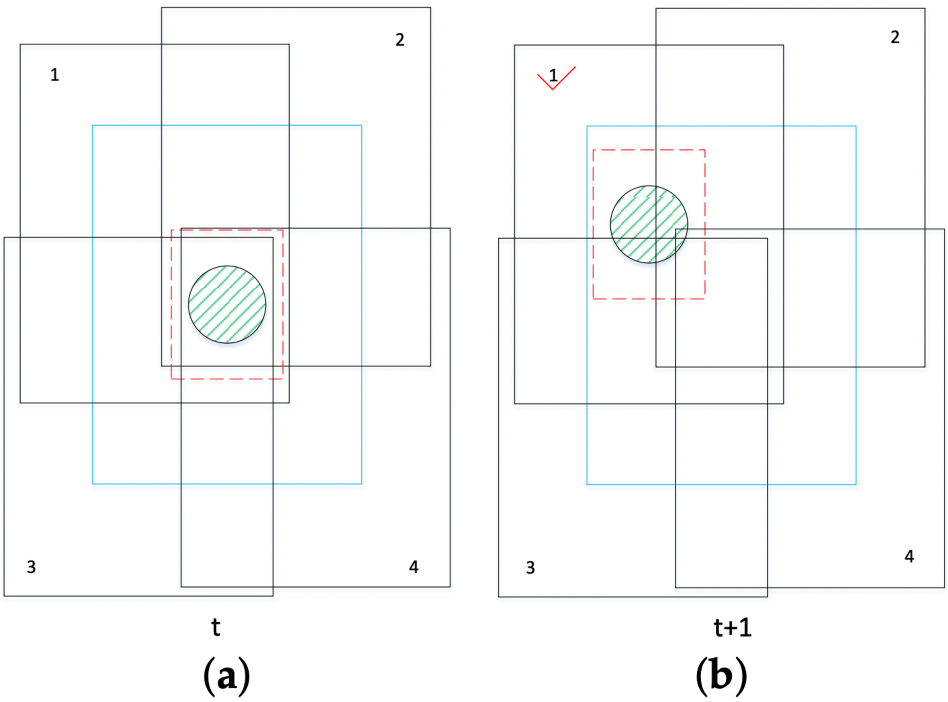



Figure 6.3 : Figure of tracking process.

### 6.3 Multi-Vehicle Management Module

The multi-vehicle management module was created using multithreading technology because the original KCF tracker could only track one object. This module distributes multi-object tracking duties to parallel blocks in order to quickly analyse and confirm the tracking state of the tracked vehicle. Assume that the detector has picked up O1, O2,..., On as the objects. All detection data are first provided to the tracker for initialization, as seen in Figure 5. The initialised objects will be displayed as Oi 1, Oi 2,..., Oin. In order to determine the vehicles O1, O2,..., On at each frame, a new object module receives them. the new object module with n = 2 is described. As shown in fig 6.4 two additional blobs, O01d and O02d, are depicted in frame t as a green ellipse. The two tracked blobs O01t and O02t are represented in frame t + 1 by two yellow ovals. By examining the overlap between the detection box and the tracking box at frame t+2, the new object module may identify the new blob new.In our experiment, the rate of overlap between the tracking box and the detection box is denoted by the symbol. A new blob will be added to the tracker if 0.1. Ot 1, Ot 2,..., and Ot m are the last tracked objects. In fact, processing several objects by the algorithm can take some time. Create an iterative multithreaded multipurpose processing mode to solve this problem. It is possible to automatically assign each item to a different thread for processing. The system simultaneously assigns each thing its own storage space. The algorithm automatically obtains the thread and the corresponding memory space when the target vanishes. These will ultimately be utilised with new products. This enables concurrent thread allocation and reclamation, which effectively handles handling numerous objects. In Figure 7, various trackers that are handled by various threads process the data of detectors O1, O2,..., On. One thread block contains S threads, while the entire thread network is made up of several thread blocks. Multi-threading technology is used to significantly lessen the computational strain.

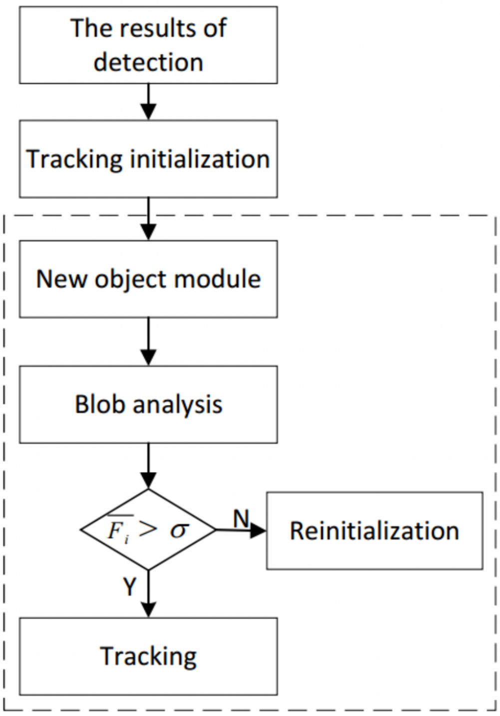



Figure 6.4: KCF tracker

Module for managing multiple vehicles. The tracking module that connects the tracker and detector is located inside the dashed box. It comprises of multi-object tracking, blob analysis, and new object module. To prevent these errors in the future, all tracker faults are analysed according to the regression response in the multi-vehicle management module. Regression's answer is denoted by the acronym Fti, where I stands for the blob number and t for the frame number. The average regression response of the blob I during tracking can be stated as follows:

$$Fi = N \sum n=1 \ Fni \qquad (7)$$

where N is the total number of frames as of the moment. The confidence level for the blob is defined as. Blob I will be continually tracked if $Fi > \sigma$. Blob I will be reinitialized by the detector if $Fi \leq \sigma$. Vehicle counts are based on the final tracked findings. In the section after this, we mostly talk about the vehicle counting module.

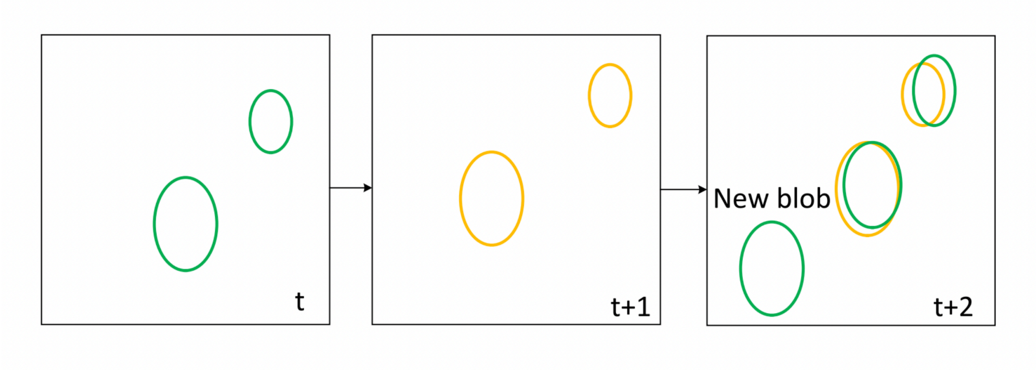



Figure 6.5: New-object identification.

In fig. 6.6 above The processing is done in the order of 1 to n. In frame t, there are two green blobs shown. Two blobs (yellow) are tracked in frame t + 1. After that, a blob (green) in frame t + 2 is labelled as a new blob and will be added to the tracker.

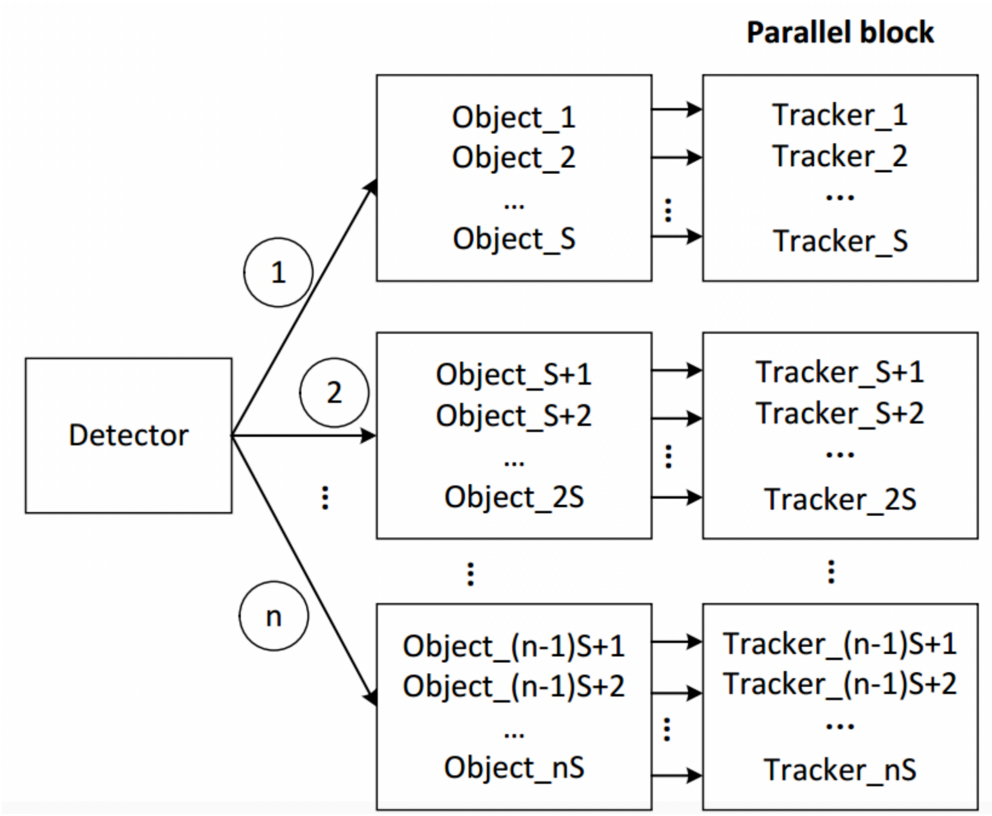



Figure 6.6: The multi-object tracker

As depicted in fig. 6.5 The detector's output is processed in parallel blocks with various numbered trackers in a sequential manner. S stands for the CPU kernel number. In conclusion, the multi-object management module serves as the foundation for the suggested vehicle counting technique.

### 6.4 Vehicle Counting Module

Branding and virtual test lines are the foundation of common car counting techniques. The first method involves counting the number of connecting points, while the second creates a fictitious test line on the highway. establishes the area for car counting. Vehicles in the space below the red line should be counted. As seen in Figure 6.6, the car is divided into two directions when travelling down the highway. It is unnecessary to draw many lines in the vicinity to detect whether a vehicle is passing thanks to the numerous vehicle tracking and management modules in our system. The vehicle ID and direction data are stored in the multi-vehicle management module. This can be used to precisely count the number of automobiles nearby. Consider the scenario when m vehicles are being tracked in frame t. Determine a count plus one if the car is tracked in frame t + 1 with a different ID.

Assemble the tracker and detector and collaborate within a single framework. Each tracker follows separate objects without interfering with each other. The status information for each object won't be muddled by this. The detector will re-initialize the appropriate tracker if the tracker results are inaccurate. We use multithreaded technology, which can greatly lessen the computational load, to process many trackers.

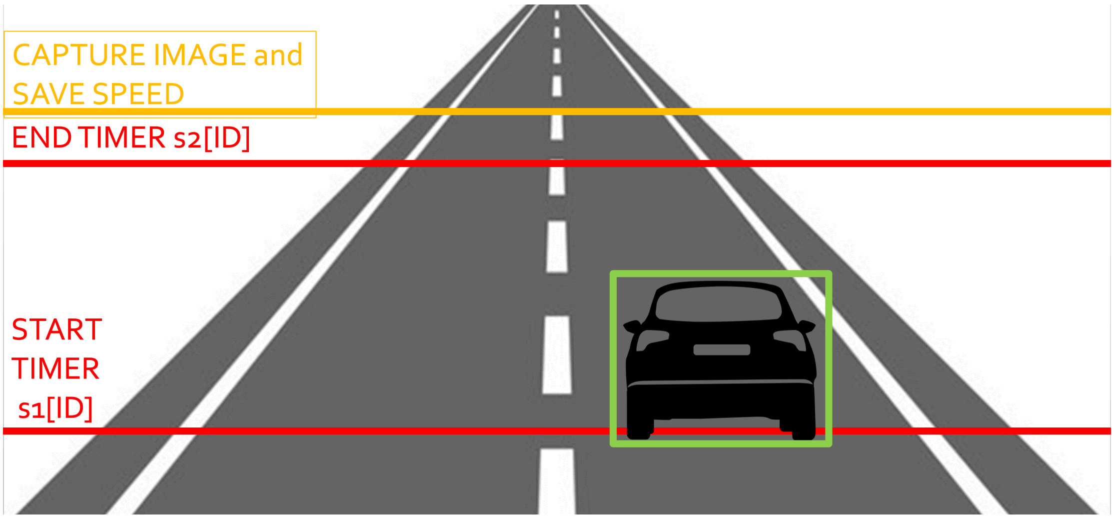
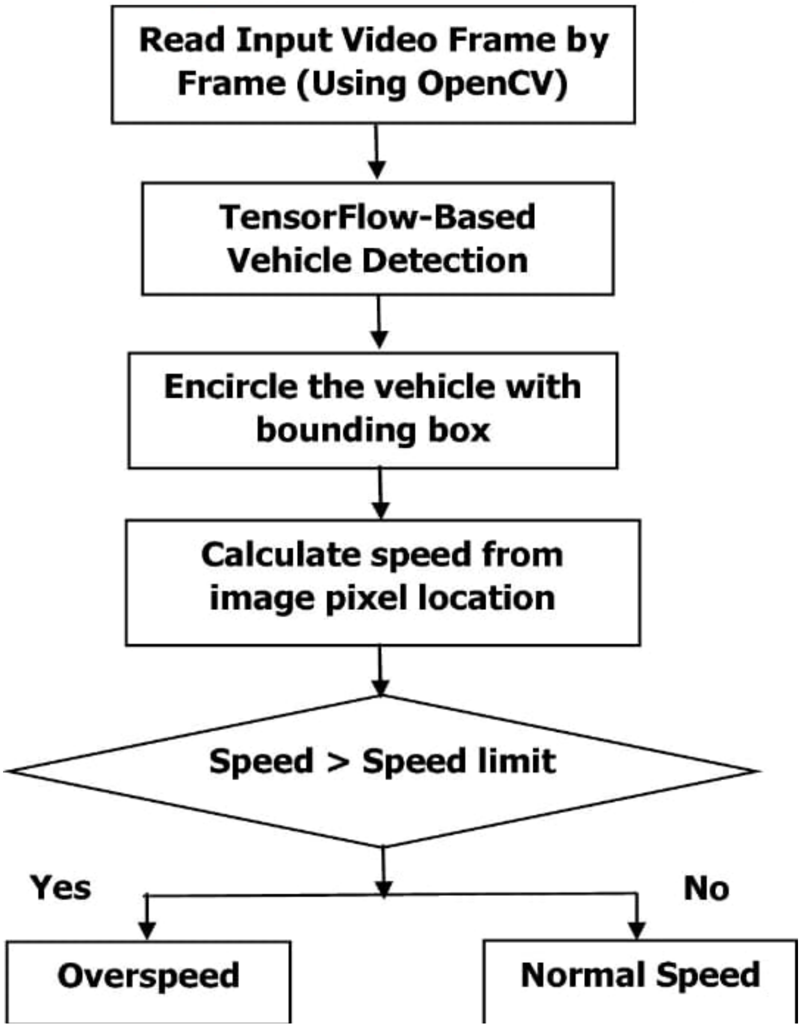



Figure 6.7: Vehicle with timer and lines defined.

# Chapter 7

## 7.1 Flowchart




## 7.2 Algorithm

Step 1- The recorded video is given as input frame by frame using OpenCV.

Step 2 - Object is detected by using Image subtraction.

Step 3 -The object is tracked and bounding box encircle the vehicle according to the area threshold

Step 4 - Speed is calculated from the Time difference between the position of a vehicle and displayed on top of bounding box

Step 5- speed limit is checked i.e. 80kmph

Step 6 - if current speed of vehicle is above speed limit then data of violators is pointed out saved with the summary

## Chapter 8

# PROJECT MODEL

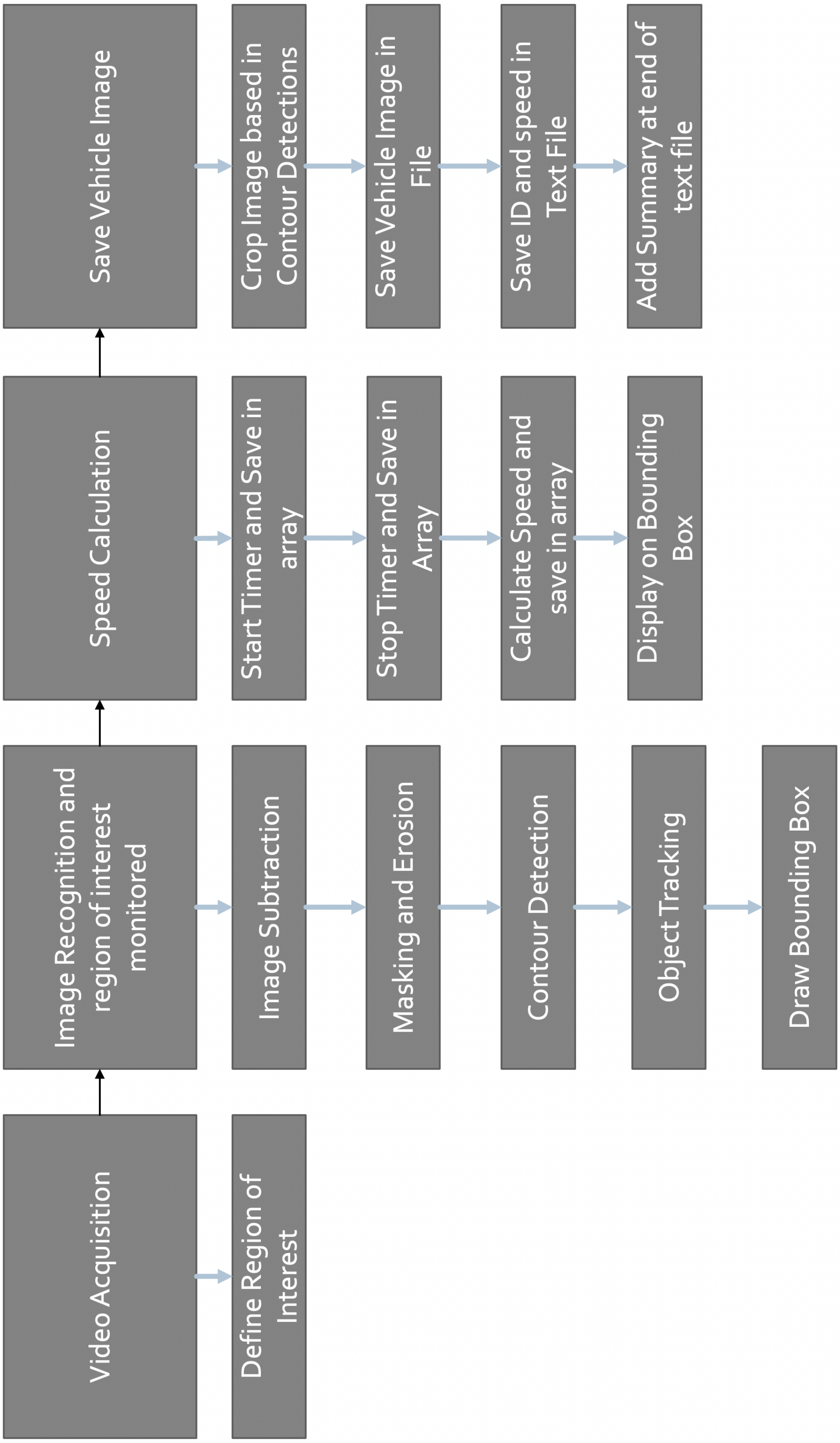



Figure 8.1 : Project Model Block Diagram

## 8.1 Video Acquisition

A digitized version of an external video stream is considered as a video input. It typically requires encoding and post-production of the software in addition to the equipment used to transfer the original stream to a digital file format (including tape recorders, digital storage, or camcorders).

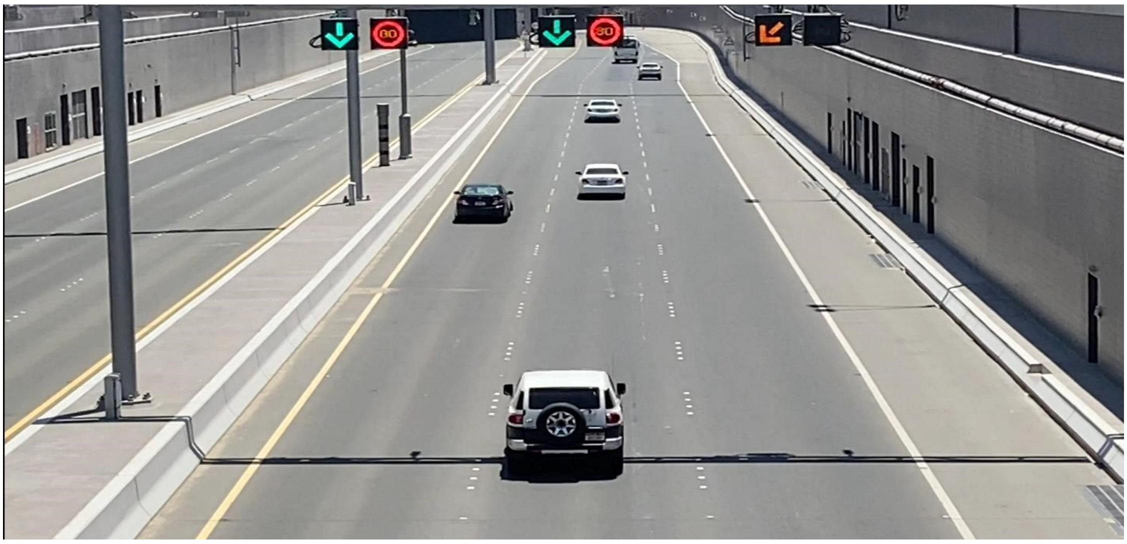


Figure 8.2 : Sample Video Screenshot

## 8.2 Region of Interest (ROI) and Masking

The region of interest (ROI) occupies a smaller portion of the original video.

1. Image Subtraction helps find the difference between two frames. Image subtraction is performed to detect a moving vehicle.
2. Masking is done so that the moving vehicle appears white, and the remaining images appear black.


Figure 8.3 : Masking Image Screenshot

## 8.3 Contour Detection and Object Tracking

Contours are detected according to the area threshold of the number of pixels. Threshold is used to prevent detection of contours of smaller moving objects other than vehicles. The object tracks the distance between the two contours between the frames based on. Each contour is assigned to an ID.

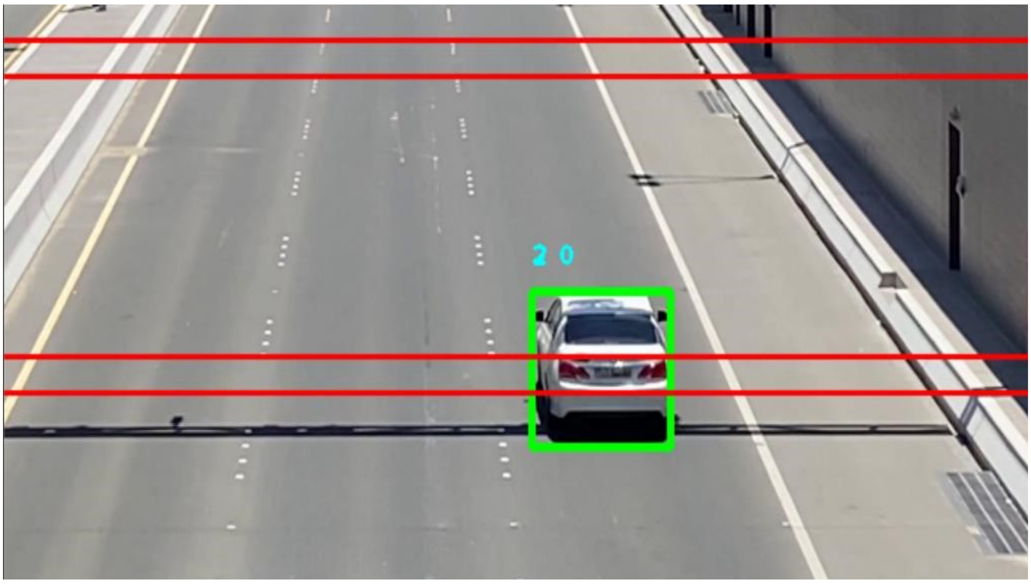



Figure 8.4 : Contour Detection

## 8.4 Speed Estimation

Calculate the time difference between the positions of the vehicles and estimate the speed according to the formula. The timer starts when the vehicle reaches the line 1 and ends when the vehicle crosses the line 2. The speed is displayed above the bounding box only when the vehicle crosses both lines.

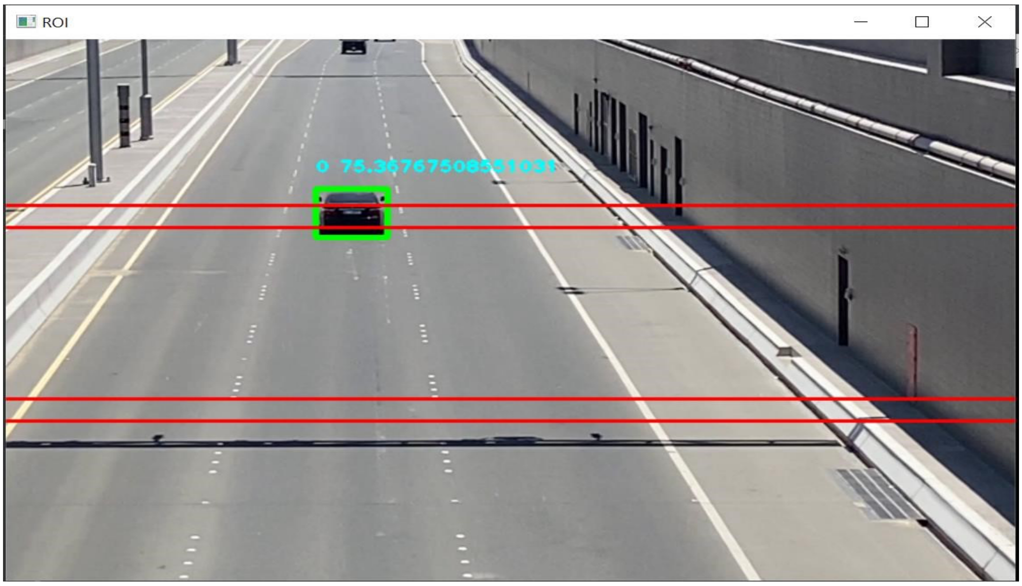



Figure 8.6 : Speed Estimation

## 8.5 Save Vehicle Data

An image of the bounding box (of the car) is saved to a file along with its speed. Vehicles exceeding the speed limit are assigned to a separate folder.

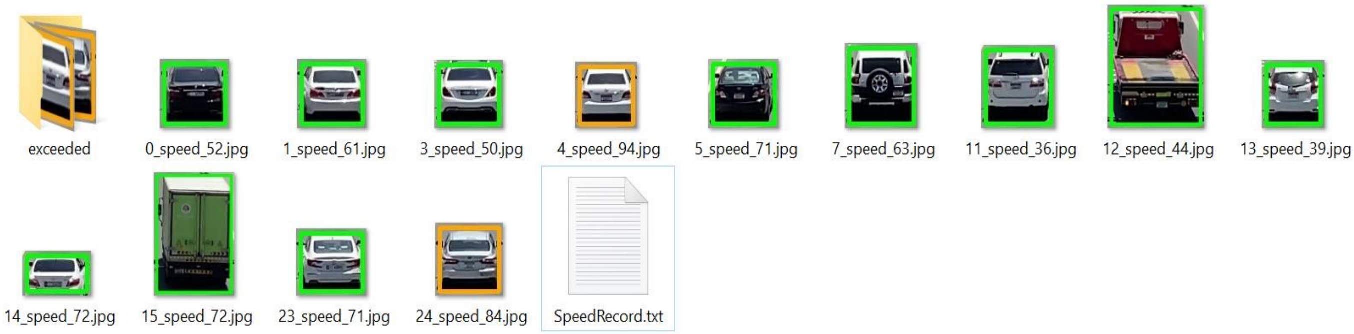
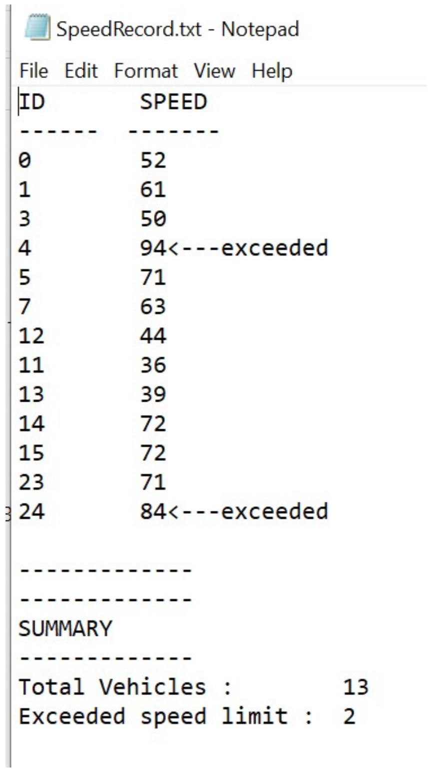



Figure 8.6 : Save Vehicle Data

## 8.6 Create Summary

Vehicle data is stored in a format of text file. Vehicles that exceed the speed limit are flagged. Displays a summary of the counting of vehicles and the speed violators.


SpeedRecord.txt - Notepad

File Edit Format View Help

```
ID        SPEED
------  -------
0         52
1         61
3         50
4         94<---exceeded
5         71
7         63
12        44
11        36
13        39
14        72
15        72
23        71
24        84<---exceeded

-------------
-------------
SUMMARY
-------------
Total Vehicles :        13
Exceeded speed limit :  2
```

Figure 8.7 : Saved Data with ID, Speed and Summary

# Chapter-9

# IMPLEMENTATION

## 9.1 Video Acquisition

```
cap = cv2.VideoCapture("Resources/traffic3.mp4")
```

## 9.2 Region of Interest and Masking

```
#KERNALS
kernalOp = np.ones((3,3),np.uint8)
kernalOp2 = np.ones((5,5),np.uint8)
kernalCl = np.ones((11,11),np.uint8)
fgbg=cv2.createBackgroundSubtractorMOG2(detectShadows=True)
#MASKING
fgmask = fgbg.apply(roi)
ret, imBin = cv2.threshold(fgmask, 200, 255,
cv2.THRESH_BINARY) mask1 = cv2.morphologyEx(imBin,
cv2.MORPH_OPEN, kernalOp) mask2 = cv2.morphologyEx(mask1,
cv2.MORPH_CLOSE, kernalCl)
```

## 9.3 Contour Detection

```
contours,_ = cv2.findContours(mask2,cv2.RETR_TREE,cv2.CHAIN_APPROX_SIMPLE)
detections = []
 for cnt in
contours:
#print(cnt)
    area = cv2.contourArea(cnt)
if area > 1000:
        x,y,w,h = cv2.boundingRect(cnt)

cv2.rectangle(roi,(x,y),(x+w,y+h),(0,255,0),3)
detections.append([x,y,w,h])
```

## 9.4 Object Tracking

```
for rect in
objects_rect:     x, y,
w, h = rect     cx = (x
+ x + w) // 2     cy =
(y + y + h) // 2

    # Find out if that object was detected already
same_object_detected = False
     for id, pt in
self.center_points.items():
        dist = math.hypot(cx - pt[0], cy - pt[1])
         if dist <
150:
            self.center_points[id] = (cx, cy)
#print(self.center_points)
            objects_bbs_ids.append([x, y, w, h,
id])             same_object_detected = True
```

## 9.5 Speed Estimation

### 1. Timer Start and stop

```
if (v >= 325 and v <= 345):

    self.s1[id] = time.time()


if (v >= 150 and v <= 170):
    self.s2[id] = time.time()

    self.s[id] = self.s2[id] - self.s1[id]
```

### 2. Speed Formula

```
def getsp(self,id):
    if (self.s[id]!=0):
        s = 439.8/self.s[id]
else:
        s = 0
return s
```

## 9.6 Drawing Rectangles and displaying on the screen

```
for box_id in boxes_ids:
x,y,w,h,id = box_id


    cv2.putText(roi,str(id)+" "+str(tracker.getsp(id)),(x,y-15),
cv2.FONT_HERSHEY_PLAIN,1,(255,255,0),2)
    #print(tracker.getsp(id))
    cv2.rectangle(roi, (x, y), (x + w, y + h), (0, 255, 0), 3)
```

## 9.7 Drawing Reference Lines

```
cv2.line(roi, (0, 325), (960, 325), (0, 0, 255), 2) cv2.line(roi,
(0, 345), (960, 345), (0, 0, 255), 2)

cv2.line(roi, (0, 150), (960, 150), (0, 0, 255), 2) cv2.line(roi,
(0, 170), (960, 170), (0, 0, 255), 2)
```

## 9.8 Save Vehicle Images and speeds

```
def capture(self,img,x,y,h,w,sp,id):
    if(self.capf[id]==0):
        self.capf[id] = 1
        self.f[id]=0
        crop img = img[y-5:y + h+5, x-5:x + w+5]
        n = str(id)+"_speed_"+str(sp)
        file = 'D://TrafficRecord//' + n + '.jpg'
        cv2.imwrite(file, crop img)
        self.count += 1
        filet = open("D://TrafficRecord//SpeedRecord.txt", "a")
        if(sp>limit):
            file2 = 'D://TrafficRecord//exceeded//' + n + '.jpg'
            cv2.imwrite(file2, crop img)
            filet.write(str(id)+" \t "+str(sp)+"<---exceeded\n")
            self.exceeded+=1
        else:
            filet.write(str(id) + " \t " + str(sp) + "\n")
        filet.close()
```

## 9.9 Create Summary

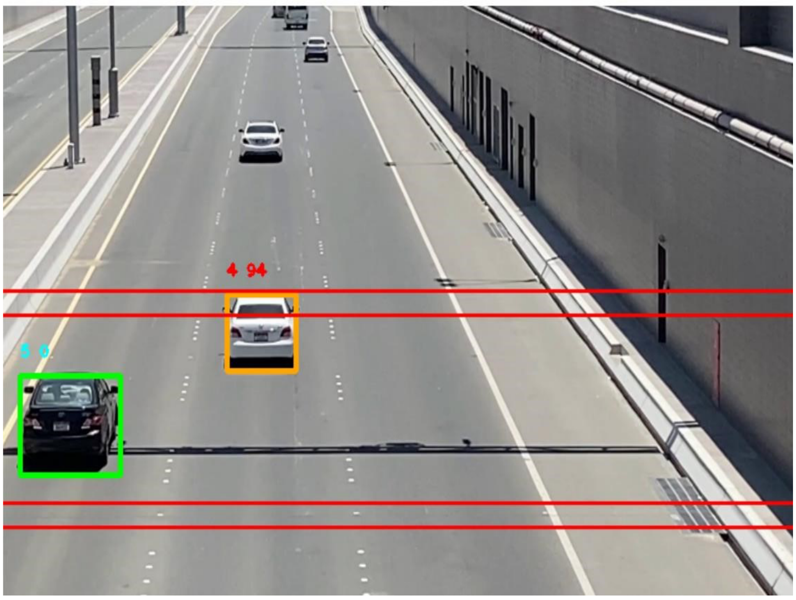
```
def end(self):
    file = open("D://TrafficRecord//SpeedRecord.txt", "a")
    file.write("\n-------------\n")
    file.write("-------------\n")
    file.write("SUMMARY\n")
    file.write("-------------\n")
    file.write("Total Vehicles :\t"+str(self.count)+"\n")
    file.write("Exceeded speed limit :\t"+str(self.exceeded))
    file.close()
```

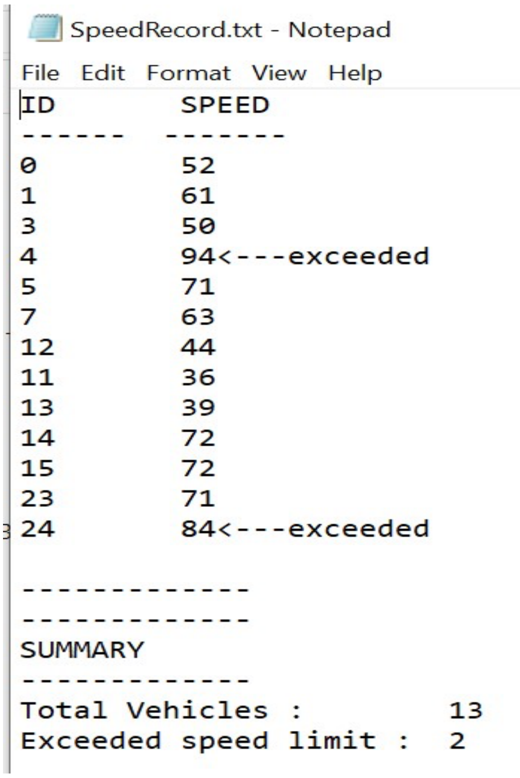
## Chapter-10

# RESULT ANALYSIS

In this project of vehicle detection and speed estimation it involves four major steps video acquisition, image recognition, speed calculation and Save Vehicle Image. At first, we define the ROI, afterwards image recognition is done by Image subtraction, Masking and Erosion, counter detection and object tracking and drawing the bounding box. When the vehicle reaches line 1, counting module starts. When it reaches line 2, we start the timer and at line 3 we stop the timer and save in array and calculate speed by using speed function and display it on bounding box. When the vehicle crosses the line 4 the cropped image based on counter detections is saved with a particular save ID and speed in text file with the summary at the end of the text file.

### OUTPUT SCREENSHOTS





SpeedRecord.txt - Notepad

File Edit Format View Help

```
ID        SPEED
------   -------
0         52
1         61
3         50
4         94<---exceeded
5         71
7         63
12        44
11        36
13        39
14        72
15        72
23        71
24        84<---exceeded

-------------
-------------
SUMMARY
-------------
Total Vehicles :        13
Exceeded speed limit :  2
```

Figure 9.1: Speed Radar Main Output

Figure 9.2 : Saved Vehicle Data

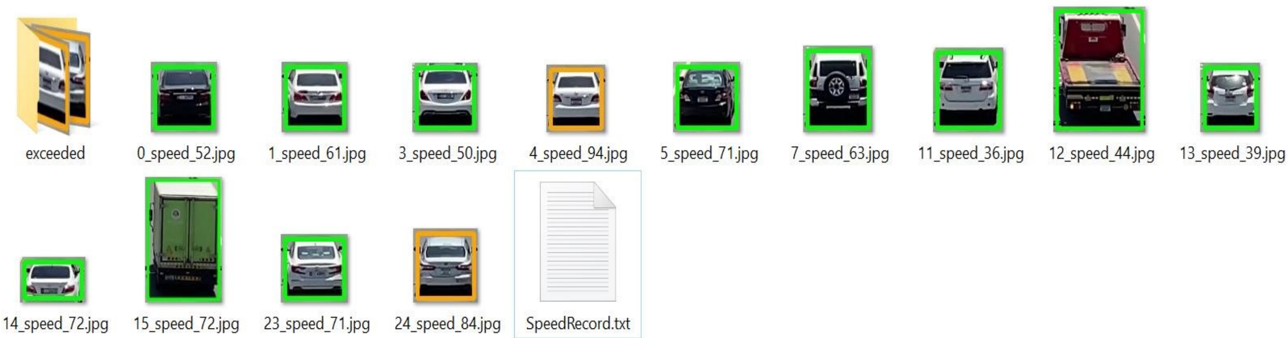



Figure 9. 3 :Saved Vehicle Images

**Chapter-11**

# CONCLUSION

This project can successfully detect multiple vehicles and able to track them and it can evaluate its speed and saves the vehicle data. Vehicle detection is accurate for the static background, estimated speed may vary by 0 - 2 km/h depending on program speed. However, when two vehicles move very closely, they may be detected as one object. Because this project uses movement to distinguish the vehicle from the background, the camera should be as still as possible. It helps in Monitoring speed violators, Providing road safety and to improve traffic rules and regulations.

## 11.1 PROJECT CHALLENGES

1. Objects which are closer to the specified value that were not vehicles are also being detected
   -This problem is resolved by changing it to a masking method.

2. Number Plates were not readable
   -This problem is resolved by saving images of the vehicle.

3. Multiple vehicles were not able to be detected
   -This problem is solved by saving the speed timer in array format instead of a single integer format.

4. Vehicles which were close together were considered as a single object
   -Eroded mask image to better distinguish nearby objects
   -Moved detection line further backwards

## 11.2 FUTURE SCOPE

1. Number plate detection.
2. Variation with different camera angles
3. Finding other violations like wrong lane detection and tailgating detection.
4. Speed detection of small and large vehicles
5. Live speed detection


# ACKNOWLEDGEMENT

I would like to thank Prof. B.A Sridhara sir,  Siddaganga institute of technology,tumkur, for his guidance and support throughout the entire project and for giving us the opportunity to explore deeper into Image Processing Concepts both theoretically and practically.

# REFERENCES

[1] J. Gerát, D. Sopiak, M. Oravec and J. Pavlovicová, "Vehicle speed detection from camera stream using image processing methods," 2017 International Symposium ELMAR, 2017, pp. 201-204, doi:10.23919/ELMAR.2017.8124468.

[2] P. K. Thadagoppula and V. Upadhyaya, "Speed detection using image processing," 2016 International Conference on Computer, Control, Informatics, and its Applications (IC3INA), 2016, pp. 11- 16, doi: 10.1109/IC3INA.2016.7863015.

[3] S. Hua, M. Kapoor, and D. C. Anastasiu, "Vehicle Tracking and Speed Estimation from Traffic Videos," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2018, pp. 153-1537, doi: 10.1109/CVPRW.2018.00028

[4] ScienceDirect – 2020 - Determining Vehicle Speeds Using Convolutional Neural Networks by Alexander Grents, Vitalii Varkentin*, Nikolay Goryaev

[5] ScienceDirect – 2020 - Detection of Vehicle Position and Speed using Camera Calibration and Image Projection Methods by Alexander A S Gunawana,* , Deasy Aprilia Tanjunga, Fergyanto E. Gunawan

[6] ScienceDirect – 2020 - An Efficient Approach for Detection and SpeedEstimation of Moving Vehicles by Tarun Kumar and Dharmender Singh Kushwaha

[7] "Image recognition method based on deep learning," 2017 29th Chinese Control and Decision Conference (CCDC), 2017, pp. 4730-4735, doi: 10.1109/CCDC.2017.7979332.

[8] Vision-based vehicle detection and counting system using deep learning in highway scenes- a journal by Huansheng Song, Haoxiang Liang*, Huaiyu Li, Zhe Dai and Xu Yun

[9] International Research Journal of Engineering and Technology (IRJET) Volume: 07 Issue: 05 | May 2020 Speed Detection using Python - MD MUBEEN KHAN1, K SRINIVAS2

[10] Vehicle Detection, Tracking, and Speed Estimation Using Deep Learning and Computer Vision: An Application Perspective – a journal by Shiksha Jaiswal Shristi V. Kotaiah Chaurasia Sandeep Chaurasia Sandeep

[11] An Efficient Approach for Detection and Speed Estimation of Moving Vehiclespublished by Procedia Computer science- Tarun Kumar and Dharmender Singh Kushwaha∗ National Institute of Technology, Allahabad, India

[12] H. Chung-Lin and L. Wen-Chieh, "A vision-based vehicle identification system," in Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on, 2004, pp. 364-367 Vol.4.

[13] Z. Wei, et al., "Multilevel Framework to Detect and Handle Vehicle Occlusion," Intelligent Transportation Systems, IEEE Transactions on, vol. 9, pp. 161-174, 2008.

[14] N. K. Kanhere and S. T. Birchfield, "Real-Time Incremental Segmentation and Tracking of Vehicles at Low Camera Angles Using Stable Features," Intelligent Transportation Systems, IEEE

Transactions on, vol. 9, pp. 148-160, 2008.

[15] N. K. Kanhere, "Vision-based detection, tracking and classification of vehicles using stable features with automatic camera calibration," ed, 2008, p. 105.

[16] A. H. S. Lai, et al., "Vehicle type classification from visual-based dimension estimation," in Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE, 2001, pp. 201-206.

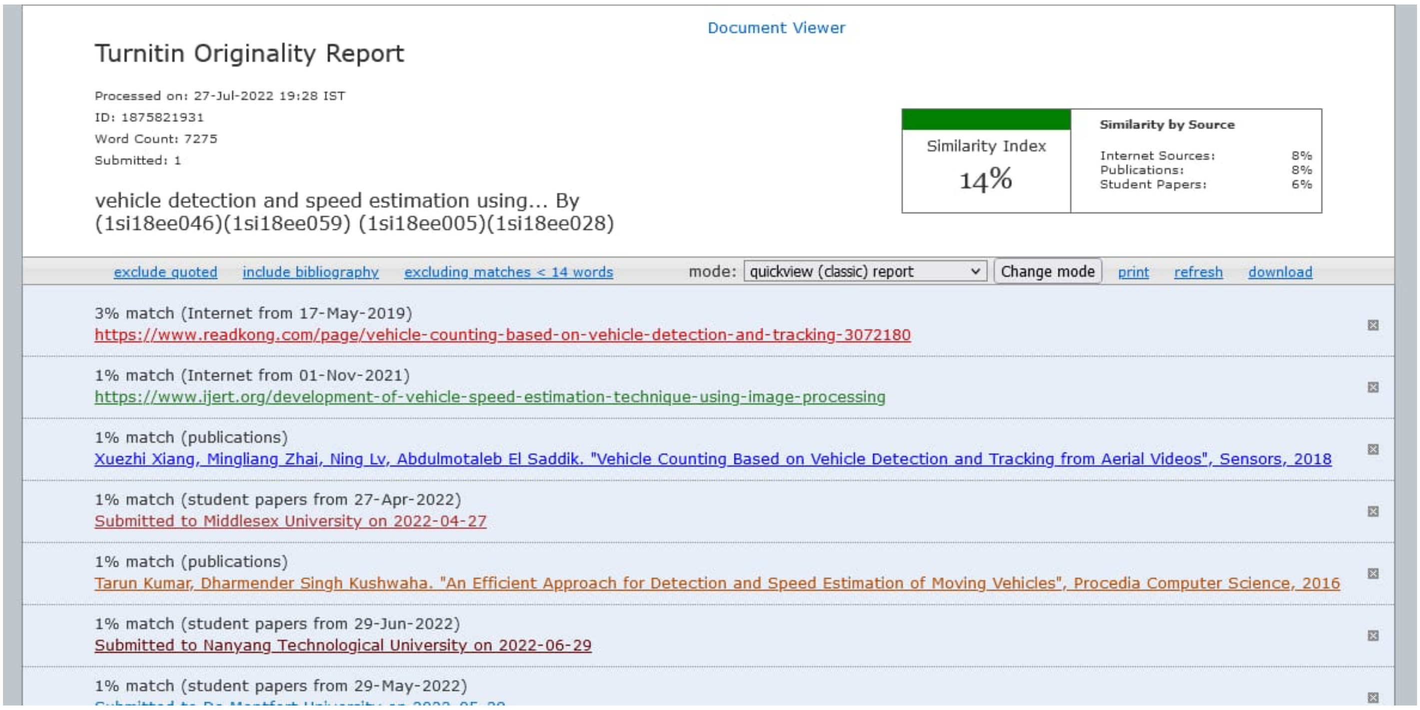
[17] J.-W. Hsieh, et al., "Automatic traffic surveillance system for vehicle tracking and classification," Intelligent Transportation Systems, IEEE Transactions on, vol. 7, pp. 175-187, 2006.

[18] N. K. Kanhere, et al., "Vehicle Segmentation and Tracking in the Presence of Occlusions," TRB Annual Meeting Compendium of Papers, Transportation Research Board Annual Meeting, pp. 89-97, 2006.

[19] H. Mao-Chi and Y. Shwu-Huey, "A real-time and color-based computer vision for traffic monitoring system," in Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference on, 2004, pp. 2119-2122 Vol.3.

[20] G. D. Sullivan, et al., "Model-based vehicle detection and classification using orthographic approximations," Image and Vision Computing, vol. 15, pp. 649-654, 1997.

[21] A. N. Rajagopalan, et al., "Higher order statistical learning for vehicle detection in images," in Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on, 1999, pp. 1204-1209 vol.2.



Document Viewer

Turnitin Originality Report

Processed on: 27-Jul-2022 19:28 IST
ID: 1875821931
Word Count: 7275
Submitted: 1

vehicle detection and speed estimation using... By (1si18ee046)(1si18ee059) (1si18ee005)(1si18ee028)

| Similarity Index | Similarity by Source | |
|---|---|---|
| 14% | Internet Sources: | 8% |
| | Publications: | 8% |
| | Student Papers: | 6% |

exclude quoted include bibliography excluding matches < 14 words mode: quickview (classic) report Change mode print refresh download

3% match (Internet from 17-May-2019)
https://www.readkong.com/page/vehicle-counting-based-on-vehicle-detection-and-tracking-3072180

1% match (Internet from 01-Nov-2021)
https://www.ijert.org/development-of-vehicle-speed-estimation-technique-using-image-processing

1% match (publications)
Xuezhi Xiang, Mingliang Zhai, Ning Lv, Abdulmotaleb El Saddik. "Vehicle Counting Based on Vehicle Detection and Tracking from Aerial Videos", Sensors, 2018

1% match (student papers from 27-Apr-2022)
Submitted to Middlesex University on 2022-04-27

1% match (publications)
Tarun Kumar, Dharmender Singh Kushwaha. "An Efficient Approach for Detection and Speed Estimation of Moving Vehicles", Procedia Computer Science, 2016

1% match (student papers from 29-Jun-2022)
Submitted to Nanyang Technological University on 2022-06-29

1% match (student papers from 29-May-2022) turnitin

# Digital Receipt

This receipt acknowledges that Turnitin received your paper. Below you will find the receipt information regarding your submission.

The first page of your submissions is displayed below.

| | |
|---|---|
| Submission author: | (1si18ee046)(1si18ee059)(1si18ee005)(1si18ee028) |
| Assignment title: | BE |
| Submission title: | vehicle detection and speed estimation using deep learning |
| File name: | VSD_final_report.docx |
| File size: | 10.41M |
| Page count: | 30 |
| Word count: | 7,275 |
| Character count: | 38,175 |
| Submission date: | 27-Jul-2022 07:28PM (UTC+0530) |
| Submission ID: | 1875821931 |


**ABSTRACT**

Intelligent traffic control and surveillance are the basic requirements for developing a smart city in India. As urban traffic problems increase, video-based traffic surveillance systems have become popular over the last few decades. These systems are extremely useful for monitoring and managing a variety of traffic conditions, including traffic management, accident prevention and safe transportation. Within that range, one of the purposes is to measure vehicle speed on roads. Determining the speed of the vehicle thus becomes a challenging task. In determining the speed of vehicle detection and vehicle tracking are key steps. To overcome disadvantages of the traditional RADAR/LIDAR technology here we are using the method of determining vehicle speed using image processing. This includes detecting moving vehicles, estimating their speed and detecting them about exceeding the speed limit. This project detects and tracks vehicle passing through the tracking area, vehicle tracking is based on relative positions of vehicles in consecutive frames and keeps records of vehicle positions and red marks vehicles that exceed the speed limit and alerts the traffic manager.

The fundamental requirement for the development of India's smart cities is an intelligent system of traffic control and surveillance. Traffic monitoring systems based on video have gained popularity in recent years as urban traffic issues worsen. These systems are particularly helpful for managing various traffic conditions, such as traffic management, accident prevention, and safe transportation, with one of the goals being to determine vehicle speed on the road. Therefore, determining a vehicle's speed has become a difficult process. Vehicle identification and vehicle tracking are the crucial steps in speed determination. Here, image processing is used to determine vehicle speed in order to overcome the drawbacks of the old approach. Detecting moving cars, determining their speed, and identifying speed limit violations are all included in this. The method described in this study uses a single camera in a well-lit environment to detect moving vehicles and estimate their speeds. It is both effective and innovative. The suggested method identifies and tracks any vehicle that passes through the monitoring area. The tracking of the vehicles is based on their relative positions in successive frames, and a record of their positions is kept. Additionally, it warns the traffic manager and highlights in red any vehicles travelling at excessive speeds.