VioCam : Smart Camera for Traffic Violation Detection

Capstone Project Report

Submitted by:

101816022 ARYAN TYAGI
101816016 TUSHAR KUMAR RIYAT
101803717 ADITYA SINGH

BE Fourth Year- COSE

Under the Mentorship of

DR. SEEMA WAZARKAR

Assistant Professor

Computer Science and Engineering Department

Thapar Institute of Engineering and Technology, Patiala

MAY 2022

ABSTRACT

VioCam is a smart camera system that aims to automate the process of violation detection at traffic junctions. The VioCam system takes care of all phases of a traffic violation from detection to identification of the violator. It is a state-of-the-art software solution that leverages the power of Machine and Deep Learning to take care of several traffic related tasks through a single camera. The project has great potential to make violation detection faster and more accurate while at the same time keeping the costs of setting up a software solution as low as possible. The models trained for this project have been custom engineered for Indian vehicles and perform exceptionally in most conditions.VioCam also aims to control traffic congestion near traffic lights. It can analyze the amount of traffic in a specific lane and then dynamically control the traffic light timings according to the amount of traffic.This is designed to detect various vehicles and also the type of vehicles i.e. four wheeler or two wheeler.It can also detect whether the driver is wearing helmet ,besides this human detection has significantly improved.

DECLARATION

We hereby declare that the design principles and working prototype model of the project entitled ‘VioCam : Smart Camera for Traffic Violation Detection’ is an authentic record of our own work carried out in the Computer Science and Engineering Department, TIET, Patiala, under the guidance of Dr. Seema Wazarkar during the 8th semester (2022).

Date: 11 May, 2022

Roll No	**Name**	**Signatures**
101816022	Aryan Tyagi	Aryan
101816016	Tushar Kumar Riyat	[illegible]
101803717	Aditya Singh	Aditya

Counter Signed by :

Dr. Seemka Wazarkar
Assistant Professor
CSED,
TIET, Patiala

ACKNOWLEDGEMENT

We would like to express our thanks to our mentor Dr. Seema Wazarkar. She has been of great help in our venture, and an indispensable resource of technical knowledge. She is truly an amazing mentor to have.

We are also thankful to Dr. Maninder Singh, Head, Computer Science and Engineering Department, entire faculty and staff of Computer Science and Engineering Department, and also our friends who devoted their valuable time and helped us in all possible ways towards successful completion of this project. We thank all those who have contributed either directly or indirectly towards this project.

Lastly, we would also like to thank our families for their unyielding love and encouragement. They always wanted the best for us and we admire their determination and sacrifice.

Date :11/05/2022

Roll No	**Name**	**Signatures**
101816022	Aryan Tyagi	Aryan
101816016	Tushar Kumar Riyat	[illegible]
101803717	Aditya Singh	Aditya

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Abbreviation	Full-form
YOLO	You Only Look Once
OCR	Optical Character Recognition
RFID	Radio Frequency Identification
AI	Artificial Intelligence
ML	Machine Learning
LMV	Light Motor Vehicle
HMV	Heavy Motor Vehicle
ANN	Artificial Neural Network
CNN	Convolutional Neural Network

1. INTRODUCTION

1.1 Project Overview

VioCam is a project that can be set up at every junction in each state and district of the country to reduce the number of traffic accidents and catch the traffic rule violators. The project mainly aims at catching the red light jumpers and eventually adding other traffic rules like biker helmet detection, zebra crossing violation, seat-belt detection etc. In 2015 alone, around five lakh road accidents were reported in India, which killed around 1.5 lakh people and injured 5 lakh people. Road accidents can be reduced by a significant amount just by simply following traffic rules and lowering the number of road accidents subsequently reduces the number of road accident induced deaths. The more violators are caught, the lesser will be the tendency of people to break rules. The project can be set up at every traffic light as well as regular stopping points in order to detect violations. Once functioning, the camera will be able to detect violators of the aforementioned traffic rules. The violators will be detected with the help of computer vision and machine learning and the number plates of the violators will be recognised and sent to the server where an automatic challan will be generated. The violation may also be verified by a police officer before sending out the challan to minimise false positive rate. Eventually, more features like traffic congestion control i.e. dynamically changing the duration of red and green lights depending on the flow of traffic on all sides of an intersection. Introduction of traffic congestion control will also reduce the tendency of drivers to jump the red light. Pedestrian detection will be implemented as well where if a pedestrian wishes to cross a road and there is not a lot of traffic on it, the light can be turned green dynamically.

Traffic violations happen everywhere whether it be urban or rural areas, hence the project is extremely scalable. It can be put up in each state and district of India and the world in order to automatize the traffic management system and make it more technologically advanced. Currently, most of the major cities of India have set up Speed Enforcement Cameras which track the speed of a vehicle and detect if they are going faster than the speed limit but very few cities have implemented the Traffic Light Enforcement Camera. The reason for the same might be limited budgeting, no access to accurate and scalable technology, etc.. Our project aims to solve this problem accurately and efficiently while keeping the cost of cameras and equipment relatively low, so that each city in India can set up the Traffic Violation Detection Camera System.

1.2 Need Analysis

India has a huge amount of traffic whether it be cities or rural areas, and an enormous amount of traffic leads to an enormous amount of traffic accidents. A total of 151,113 people were killed in 480,652 road accidents across India just in 2019 and this number is predicted to increase further in the future. Traffic rules are put up in order to prevent all of these road accidents but a great deal of people violate even the simplest of traffic rules. In a diverse and vast country like India the police force simply cannot be present at each and every junction in order for enforcement of traffic rules but enforcement is necessary. This is where VioCam comes in. The system can be set up at each and every junction to catch the violators. Other features of the system include dynamic traffic control, which once enabled will be able to control the traffic light times in real time and reducing the waiting time at traffic lights. Lower waiting times will ensure that drivers wait for the traffic light to turn green instead of impatiently running over the traffic light.

1.3 Research Gaps

TABLE 1 : Research gaps

S No.	Paper/Title	Gaps
1.	Traffic Violation Detection System based on RFID	The paper focuses more on detecting speed of moving vehicles (if they are within the permitted limits), lane detection, no parking area detection, and heavily relying upon RFID system, not AI/ML tools Not implemented signal jumping, not extracting the violators the number plate.
2.	Traffic Violation detection using multiple trajectories evaluation of vehicles	Focuses on lane change violations, and red light violations, no number plate localization, only uses image processing, with no other support system, which kind of limit's the whole system's capabilities.
3.	Big data platform of traffic violation detection system: identifying the risky behaviors of vehicle drivers	Captures unsafe driving behaviours, relies heavily upon the number of nodes used to run the system, for faster results, at least 10 nodes are needed to run the system and that too at 70% only. Large database capacity is needed as it is storing all kind of structured and unstructured data (big data).

4.	Deep learning approach for intelligent intrusion system	Captures only the images of vehicles used in violation, no number plate localization, no OCR support, no helmet detection, no automated challan generation.
5.	Automated Traffic Violation system using genetic algorithm and ANNs	Focuses more on 2-wheeler riders, whether they swerve or block pedestals, no support for cars, LMV, HMV etc. No dynamic congestion control system.

1.4 Problem Definition and Scope

Problem statement – To identify and catch people violating traffic rules (initially - red light jumping and no helmet riding) and automatically generate a Challan according to the violation. Additionally, control the traffic lights to avoid traffic congestion.

Scope-The project can be implemented at each and every junction in any country. The cost and size of the project will allow it to be implemented anywhere. It would be better if it is deployed at busy roads on the traffic lights.

1.5 Assumptions and Constraints

One of the assumptions for the project is that there are only two traffic rule violations. One is red light jumping and one is no helmet riding (on two-wheelers). Not all the traffic rule violations can be identified with AI, currently for the project only two traffic rule violations are considered. In further iterations of the project, we could implement identification of more rules.

It is also assumed that when working in reality, the project will be Government approved and will be able to access vehicle and owner database from the Ministry of Road Transport and Highways or VAHAN services seamlessly. The challan will be generated automatically with details from the database and will be sent to the owner at their provided contact details.

1.6 Standards

- Python 3.8
- Yolov4 and Yolov3
- HTML v5
- CSS v3
- JS ES6
- Flask 2.0.2

- OpenCV 4.5.3
- Numpy 1.20.2

1.7 Approved Objectives

- Implement the capability to differentiate between cars and other types of vehicles.
- Train the vehicle detection model on Indian cars.
- Through transfer learning, convert the multi class model to a single class model.
- Improve helmet detection.

1.8 Methodology

Traffic rule violation detection: A low frame rate video will be captured by the camera and each frame will be passed on to the model for processing. A certain threshold line will be established, outside of which the model will detect a vehicle as a violator using computer vision. The violating vehicle image will be captured and will be further processed.

Number plate detection: For every violation detected by our system, the violating vehicle will be isolated from the image and the isolated image will be processed. The number plate of the violating vehicle will be then further detected with the help of computer vision. Once found, the number plate will be cropped out and then using image thresholding, image segmentation, and OCR the vehicle number will be extracted out.

Vehicle Owner Details Extraction: Using the vehicle number, the owner details will be looked up in the database. The details will then be passed on to the challan generator.

Automatic Challan Generation: After the violator is found out using the number plate, an automatic challan will be generated. The challan record will be pushed into the database. The challan will be then sent to the address of the vehicle owner.

Helmet Detection: Our system kept track of the presence of helmets on roads.

1.9 Project Outcomes and Deliverables

After completion, the project will be able to:

- Automatically detect and capture traffic rule violators
- Capture the number plate and extract out the defaulter information from the database
- Automatically generate challan and send it to the defaulter
- Detect humans in the camera feed

The project will be able to identify red light jumping and no helmet detection at first. Further, other violations can be added in the upcoming updates.

1.10 Novelty of Work

Currently, similar smart traffic cameras are set up in some cities of Haryana, Punjab, NCR, and Chandigarh in India as well as in many foreign countries. The system currently deployed contains many cameras, one for each of the violation; sometimes, as much as 5 cameras can be set up for each road. VioCam smart camera system is an integrated system, which does the entire job with just one camera for each road. It uses YOLO algorithm, which is an object identification algorithm using convolutional neural network. It identifies objects in real time, which is necessary for VioCam as it needs to detect violations in real time. A threshold line is set-up in the system and when a vehicle passes over the threshold line, it is detected as a red-signal violation. Helmets are also detected with the help of YOLO algorithm and a similar approach is implemented for no-helmet violation. In addition, dynamic traffic light control according to the traffic congestion is a novel idea that is not being used anywhere right now. Our system incorporates many functions into a single integrated project, which can be set up at any place.

2. REQUIREMENT ANALYSIS

2.1 Literature Survey

2.1.1 Theory Associated with Problem Area

The main objective of the project is to detect signal jumping and other violations, to create a safe and sound environment for the pedestrians, physically disabled and stray animals. Firstly, the traffic which is being recorded in to the camera is analysed by openCV algorithms, then the system delves into deep learning models so that it detects and extracts the portions of the image containing vehicles like cars, bikes etc. We use a demarcation system, using openCV and YOLO algorithm that uses convolutional neural networks in order to provide real time object detection. The demarcation or a threshold line is set-up on the road at some point, which acts a barrier for red-signal violation detection; if a vehicles passes the barrier it is considered a violation. Therefore, it is being constantly monitored whether any detected vehicle escapes and breaks that demarcation. As soon as any violation is encountered, an image of the vehicle used in violation is taken and with the help of OCR, we extract the number plate and the vehicle number written on it, and store it in the backend. Then this stored vehicle number is searched against all the vehicle numbers which are present in the database (if our system is able to detect whether this vehicle is a 2-wheeler or a 4-wheeler, then we only check the number against the detected vehicle type, otherwise, it is searched amongst all the records, present in the database). Database is a relational database, which uses SQL, to query amongst the records, stored in form of tables and then returns the details associated with the searched number. Then these details are used to carry out further actions, which are desired by the local police administration, or it can be automated to generate a challan and send it to the violator.

After getting the images of the vehicles found in the camera stream, we keep track of all the detected vehicles, and detected Two-wheelers are also scanned, whether they are wearing helmets or not. If the vehicle user is not wearing a helmet, the vehicle number of the corresponding vehicle is detected from the image using OCR (Optical Character Recognition) and then, after getting the vehicle number and it's other details from the database, the local police can take desired action to tackle the problem.

Stolen car recovery can also be implemented in this system. At every time when the signal turns green and cars and vehicles are crossing the signal, we can scan every vehicle’s number, and can scan, from the list of stolen vehicles, and if there’s a match, local police authorities/patrol police are informed so that they can catch the vehicle as soon as it reaches some another signal.

2.1.2 Existing Systems and Solutions

Traffic Monitoring System, Dongguan, China

In a country like China, it is a major issue to tackle since it is the most populous country in the world. Systems have been declared everywhere to automate the traffic flow management, where there are more chances of traffic congestions and violations. High definition cameras along with flash supplement lights are used to also support night detection of the vehicles, because it becomes more difficult for the cameras to focus on vehicles because of the lights and flares produced by the powerful headlights of the vehicles.

FIGURE 1: Traffic Monitoring System in China

Traffic Violation System, Khulna, Bangladesh

Traffic violation detection systems are effective tools to help traffic administration to monitor the traffic condition. It can detect traffic violations, such as running red lights, speeding, and vehicle retrogress in real time. In this paper, we propose an improved background-updating algorithm by using wavelet transform on dynamic background, and then track moving vehicles by feature-based tracking method. A complete traffic violation detection system is realized in C++ with OpenCV.

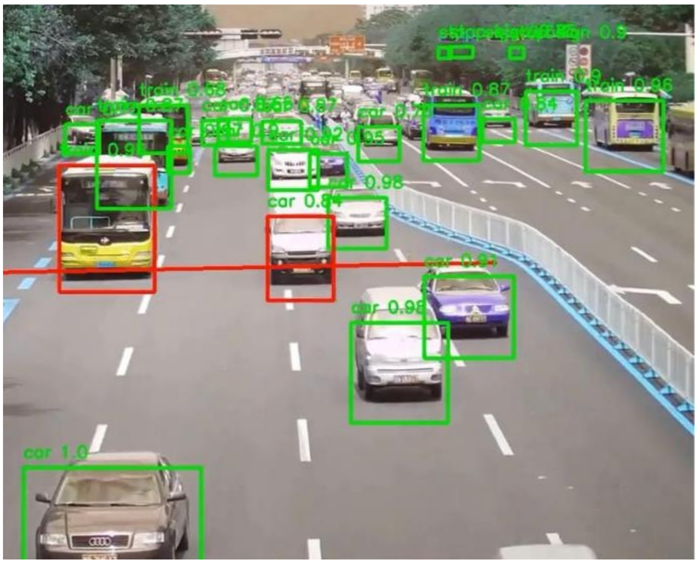

FIGURE 2: Traffic Monitoring System in Bangladesh

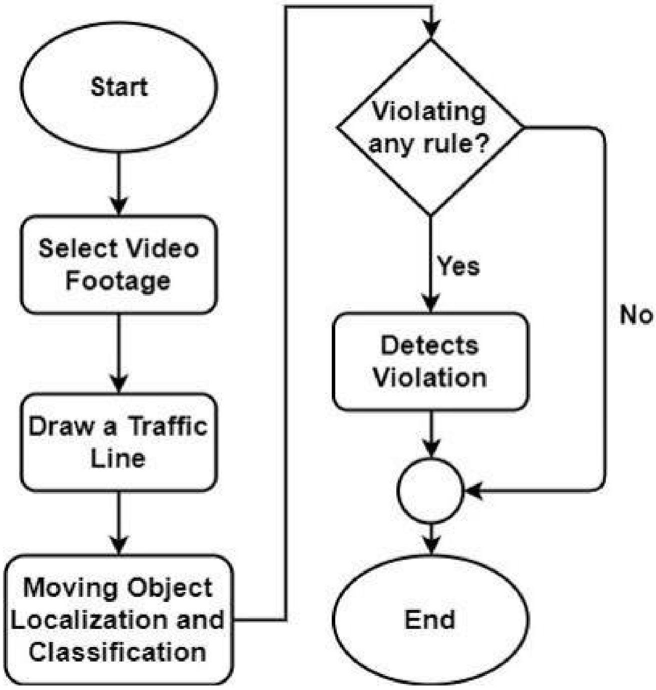

FIGURE 3: Flowchart for Traffic Violation Detection in Bangladesh

At the heart they also use openCV functions to create contours around the objects, and select that area if it is bigger than the prescribed limit than that contours are selected as buses/cars/bikes. Some of the most common openCV functions are. OnTimer(): Load video.

Processing(): Perform image processing, filtering, and background difference, and output foreground image.

LineDetect(): Detect stop lines and lanes.

LightDetect(): Detects traffic lights.

Tracking(): Perform edge detection, area setting and detection of moving vehicles, and output the original tracking window.

VehicleInfo(): Show the violation information.

TrackDetect(): Track moving vehicles, and get the centers of gravity.

Traffic Violation System, Panchkula, Haryana

Violation Detection systems in Panchkula are also very robust. Around 500+ cameras are installed at various cross-sections and roundabouts, to identify high density traffic, and to control them with automated fixed traffic lights switchings. The full process is not automated yet, all the cameras stream live to traffic control room, Panchkula, where executives are checking the feed for any violations, if any violation happens, they take the stream, zoom in the feed, find the vehicle number plate, and then take the desired action based on it

Hence, it can be said that there have been many proposed solutions for this traffic management problem, other than the discussed above, many of the countries are still practicing the manual violation catching, like India, Bhutan, Dubai, but some countries have also implemented full automated systems to to tackle it but there is still a human layer of verification on the top of the system to filter out any inconsistent, or false positive results. In America if anyone is found guilty of traffic violations, he/she is summoned to civil court for further proceedings. So slowly and slowly more and more countries are adopting these innovative AI/ML tools as per their requirements to automate this process.

2.1.3 Research Findings for Existing Literature

TABLE 2: Research Findings from each member

S No	Roll No.	Name	Paper/Title	Tools/Technology	Findings	Citation
1.	101816022	Aryan	Traffic Violation Detection System based on RFID	RFID Tags, RFID Readers Antennas, Software	Creating a system similar to toll plaza, for lane detection, embedding RFID readers on Light Poles.Track Activity in no parking zones/accident prone areas	Mehrdad Javadi, Seyyed Mohseen Hashemi, Islamic Azad University, South Tehran[14]
2.			Traffic signal violation using AI/ML	OpenCV, YoloV3, Object detection	Detect signal jumpings, using region of interest, location of vehicle in duration frames	Ruben J Franklin, Mohana, RV College of Engineering, India[15]
3.	101816016	Tushar	Traffic Violation detection using multiple trajectories evaluation of vehicles	OpenCV, Image Processing, Contouring	Image processing is being used to detect lane violation, using mean square displacement (MSD)	KatanyooKlubsuwan, WitayaKoodtalang, Sripatum University, Bangkok [16]
4.			Automated Traffic Violation system using genetic algorithm and ANNs	ANNs, genetic algorithm	Catching violators swerving or blocking the pedestrian lane with an accuracy of 90.67% with 1.34s runtime	Aaron Christian P Uy, Ana Riza F. Quiros, De La Salle University, Philippines [17]

[5.	101803717	Aditya	Big data platform of traffic violation detection system: identifying the risky behaviors of vehicle drivers	Big data, Map/reduce tech, VSM, Hadoop Distributed File System	4 Stages, MAPE - Unstructured data collection, knowledge base creation, unsafe driving behavior detection, Storing data in HDFS	Shiva Asadianfam Mahboubeh Shamsi, Qom University of Tech, Qom Iran [18]
6.			Smart Traffic violation detection system for Indian traffic	AI/ML, CNN, Shallow Neural Networks	Using pre-trained model to classify and predict violations, with that dashboard, auto-tuning is also enabled	Aman Kumar, Shakti Kundu, Teerthanker Mahaveer University, Moradabad [19]
7.	101816022	Aryan	A video-based traffic violation detection system	C++, OpenCV, background updating algorithm	Image processing tech is being used to detect red light violation, vehicle tracking by featured based tracking method	Xiaoling Wang, Li-Min Meng, Zhejiang University of Technology, Hangzhou, China[20]
8.			Deep learning approach for intelligent intrusion system	Deep Learning, DNNs	The paper proposes that, YoloV3 is a good measure to detect vehicle count detection with 97.67% accuracy	Dr.S.Raj Anand, Dr. Naveen Khilari, Vemu Institute of Technology [21]

With the growth of the number of vehicles, the number of traffic accidents is rapidly rising. Therefore, it is important to capture traffic violations to ensure traffic safety and reduce traffic accidents. Statistics show that traffic violations are the most important cause of the accidents. The intelligent transportationsystem (ITS) is being investigated globally, and is turning mature. ITS is widely deployed in the world. It is a modern transportation management system. The traffic violation detection system is an important component in ITS. It helps drivers strengthen awareness of safety when driving. This system can effectively monitor the traffic, and capture violations such that law enforcement can be applied. This helps to reduce illegal driving to ensure a smooth traffic flow.

In some research papers, Genetic algorithms have been used to apply the concept of survival of the fittest (Machine Vision for Traffic Violation Detection System through Genetic Algorithm, De La Salle University, Manila, Philippines)

2.1.4 Problem Identified

The main problem is to precisely detect the traffic violators and the vehicle used, then to extract out the details of the vehicle and the vehicle owner, so that further actions can be taken. In stolen car recovery, the system will scan all the vehicles passing through a location marked in the field, extracting the number from the feed and then matching the details against the list of stolen vehicles. Laws are for people's safety, but people do not tend to care, therefore detecting whether the two-wheeler riders are wearing a helmet or not, if they were found guilty then, the local law enforcers would take fine /other decisions.

2.1.5 Survey of Tools and Technologies Used

- Python – It is an “interpreted high-level general-purpose programming language. It is quite popular and is heavily used in the field of machine learning and data science. This programming language is particularly used for this project because there are many open source libraries made in python, which will help tremendously in implementation of the project.

- YOLOv4 and YOLOv3 – YOLO is an algorithm which uses convolutional neural network for real time object detection. This algorithm is used because the project needs fast processing of data and real time identification of violations. It offers very quick object detection with high levels of accuracy when compared to other traditional algorithms as shown in fig. 4.

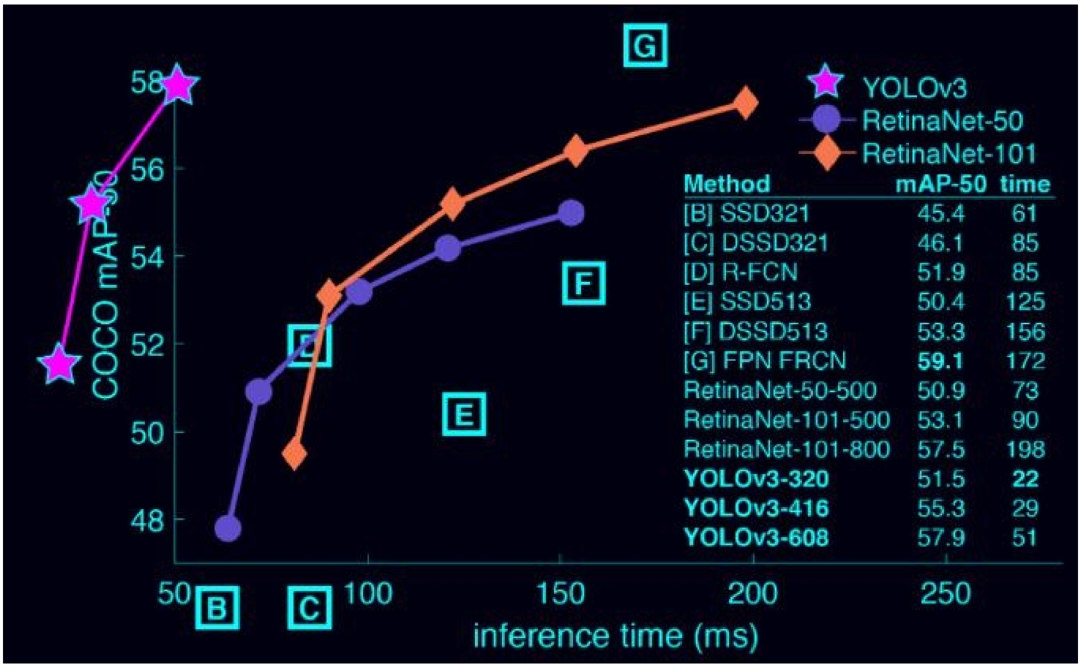

FIGURE 4: Performance of YOLOv3 on COCO Dataset

- HTML v5, CSSv3 and JS ES6 – A web-based UI will be developed for the police to monitor. The working of the model would be shown on this UI.
- Flask 2.0.2 – Flask is a web framework written in python which will be used to create the backend for UI.
- OpenCV 4.5.3 – OpenCV is a library in python which is helpful in handling video and image data. It is extensively used in computer vision tasks.
- Numpy 1.20.2 – Numpy is a library in python which adds supports for large, multi-dimensional arrays and matrices along with a large collection of high-level mathematical functions. Numpy also helps in running other python libraries smoothly.

2.2 Software Requirement Specification

2.2.1 Introduction

Software requirement specification is the description of software being developed. It consists of everything a project requires, the scope of the project, the purpose etc. Basically it provides the user with brief of the project.

2.2.1.1 Purpose

The purpose of this project is to streamline the process of catching traffic violations as well challan generation. Additionally, dynamically control the traffic lights in order to avoid traffic congestion. The project aims to help the police department in catching each and every traffic rule violators.

2.2.1.2 Intended Audience and Reading Suggestions

The intended audience for the project is State Police Departments and/or State/Central Governments. The project once deployed will be able to help catch the violators automatically, since traffic police cannot be deployed at each and every junction of the country. The project will also work in conjunction with Ministry of Road Transport & Highways/VAHAN servers as well as Police Servers in order for automatic challan generation.

2.2.1.3 Project Scope

By the end of the project, we hope to have implemented an automated system that detects primarily two traffic violations: signal jumping, and riding two-wheelers without a helmet. The system should be able to identify the violator by reading the violation-causing vehicle’s number plate using OCR and searching for that number plate in a vehicle registration database. The same camera will then be used to analyse the traffic at the junction and dynamically alter the duration of red/green lights for optimal traffic flow. For the purposes of this project, it will be assumed that the camera at the junction is placed at a favourable angle for detecting the violation(s) and identifying the violator. It is also assumed that the data required to identify the violator will be made available to our system by a governing body.

2.2.2 Overall Description

2.2.2.1 Product Perspective

VioCam is an automated violation detection system aimed to be installed at a traffic junction. The system has four main features discussed in the Product Features section. Our project cannot work independently and does need a constant stream of traffic footage and also an external interface to a vehicle registration database. Our project aims to greatly reduce the cost and the hassle of installing multiple pieces of hardware on a traffic junction for violation detection.

2.2.2.2 Product Features

The system has 4 main features:

- Traffic Rule Violation Detection - This includes detecting vehicles that are jumping traffic signals and two-wheelers where the rider is not wearing a helmet. Violating vehicles will be isolated in images extracted from a video feed of the junction.

- Number Plate Detection - Using edge-detection, the number plate of the violation-causing vehicle will be isolated from the image of the vehicle and then the number plate number will be extracted using OCR.

- Vehicle Owner Details Extraction - Once the violator’s number plate number is available, this number will be searched through a vehicle registration database to extract personal information of the violator for e-challans.

- Traffic Congestion Management via Traffic Lights - The camera installed for violation detection will also simultaneously monitor the traffic flow at the junction. Once the traffic flow is meaningfully quantified, the duration of the red/green lights can be altered dynamically for optimal congestion management.

2.2.3 External Interface Requirements

2.2.3.1 User Interfaces

The user interface will consist of four sections. Three sections will show the live feed and one section will show a terminal showing the different violations and vehicles caught by the system. One video will just show the bounding boxes around the vehicles, which will be used to show the amount of traffic in a lane. The other two videos will be used to show the violations taking place, one video feed showing red signal violation and the other video showing helmet violation. The UI will be developed in a way that it does not require any specific training to get used to, so that every police officer can easily get used to it in no-time.

2.2.3.2 Hardware Interfaces

VioCam will require cameras on the junctions for the input video feed of live traffic. Some cities already have quite a few cameras set-up on the roads and the live feed from those cameras can be fed into VioCam. For rest of the roads, cameras would have to be set-up near or on the traffic light poles in order for VioCam to run. The cameras should produce just enough high-quality video so that the system can clearly see the number plates of the vehicles.

2.2.3.3 Software Interfaces

The system also requires a dataset to train the model for identifying vehicles. COCO dataset can be used to train the model as it does work very well on any type of vehicles. Although, the model will need to be fine-tuned to not identify cycles and rickshaw pullers which have no specific rules in most of the Indian states. The YOLO algorithm will be used to facilitate the whole process of identification and make it fast.

2.2.4 Other Non-Functional Requirements

2.2.4.1 Performance Requirements

Fast I/O storage device: The local storage device attached to the camera should have a sufficiently fast I/O throughput so that the camera's working speed is not impacted by its slow I/O operations while reading or writing data to the local storage.

Access to internet: The system has to fetch some data from remote databases periodically to update its local database as well as for some other functions that are critical to working but cannot work without internet like generating Challans and issuing them to the defaulters.

Fast internet connection: The internet connection has to be sufficiently fast; around 1 Megabyte per second to make sure the system works at optimal performance.

Backup power supply: There has to be a backup power supply to power the system in case of a power outage or the absence of the Sun in case solar panels are used for powering the system during the day. The backup used for traffic signals can also be used for the system in order to keep it working.

Sufficiently lit area: The system will not be able to perform optimally in low light conditions. Therefore, it follows that the area where the system has to be implemented should be properly lit with streetlights.

2.2.4.2 Safety Requirements

Proper electrical insulation: The system should be properly insultated to make sure no electic charge is leaked to the surrounding objects like the pole of a traffic light which could pose a threat to the public.

Proper water insulation: As the system is to be installed in an outdoor environment, it is essential that it has proper protection from rains otherwise if water gets into the components, the circuits could get fried.

2.2.4.3 Security Requirements

The system should be installed at a place that is out of reach of the publicto avoid vandalism and theft. It could just be a sufficient vertical height above the road.

The enclosure of the system should not be easy to open if any unauthorised person were to tamper with it. If any intruder accesses the components of device, it could be fatal.

None of the data flow should be public as it can possibly lead to an intruder hacking onto the systems of the police or the vehicle owner database.

2.3 Cost Analysis

VioCam is an integrated system that can work with already set up cameras at different junctions. Almost all the cities across states now have CCTV Cameras for monitoring purposes. The feed form those cameras can be fed to VioCam for the functioning.

TABLE 3: Cost Analysis

S.No.	**Tools/Technology**	**Purpose**	**Apprx. Cost (in rupees)**
1	Server	To keep a track of violations and challans	1,00,000
2	Web Services	To train machine learning models	7,000

3	Website Hosting	For manual monitoring by police	1,800/month
OPTIONAL COSTS			
4	Camera	Highly quality camera for video feed	15,000
5	Pole	For camera to be set-up	25,000

2.4 Risk Analysis

TABLE 4: Risk identification

S.No.	**Risk**	**Type of risk**	**Severity**
1	Unusual number plate	Technical risk	High
2	Detection of cyclists or rickshaw pullers	Technical risk	Low
3	Malfunction in cameras due to extreme weather	Operational risk	High

Risk Management

1. Vehicles in India have a lot of unique and sometimes quirky number plates. There can be weird fonts on number plates or some phrase written on top of number plates. Newly bought vehicles often have a temporary number plate, which is just a paper with numbers written on it. Vintage vehicles often have weird number plates with just a few numbers or numbers written on top of each other. Some vehicles don't even have a number plate. All these are potential technical risks that can lead to quite a few problems. The model could potentially give a false result about the numbers on the plate; a challan is made for some innocent vehicle owner whereas the violator roams free.

   In order to mitigate this type of risk, a police control room could be set up with some employees monitoring the working of VioCam. If the model is not confident enough for the vehicle identification number, it can provide a notification and the police officer could manually look at the footage to input the exact identification number for challan generation.

   It is possible that this control room is also not necessary because some of the state governments are making it mandatory for vehicles to have the "high security registration plates" (HSRP) which will standardise the number plates and there won't be any problem in identification of numbers on the registration plate.

2. There is also a possibility that the model used for identification of vehicles also identifies cyclists or rickshaws as vehicles. In many states of India, there are no specific rules for bicycles or rickshaws and hence many cyclists are often seeing breaking the red light rule, which is not an offense for cyclists. However, if they violate any traffic rules, the model will notify about these vehicles to the police control room because these vehicles do not have a number plate. This will just be a redundant task for the police and extra work for the system. This can be mitigated if the model is trained specifically to ignore these type of vehicles.

3. Weather is a potential threat to any type of hardware that we use in VioCam. The system will be put up on junctions, possibly on traffic light poles. Hence, extreme weather like rain, hail, snow, etc. can potentially malfunction the camera or any other hardware rendering the whole system useless. The whole system would have to be replaced which would require manpower as well as cost a significant amount which is why this is a highly severe risk.

   All this can be avoided by using some weather proof encasing for the camera and hardware, which is readily available and does not cost a ton. Each VioCam can be made weather proof with these encasings and this risk can be easily managed.

3. METHODOLOGY ADOPTED

3.1 Proposed Solution

The proposed solution for tackling the problem of identifying and generating challans for traffic rule violators is to make an integrated system that performs all these tasks automatically and precisely. The current solutions as mentioned before rely on many cameras and are not fully automatic in terms that they do not have automatic challan generation capabilities. VioCam comes in as a unified and organized system that catches the violations as well as generates the challan automatically streamlining the whole process.

The initial version of VioCam will be able to catch the violation of red light signal jumping and no helmet riding (on two-wheeler vehicles). VioCam will also be able to control the traffic light signal timings for each lane dynamically for traffic congestion control.

VioCam will be placed at busy junctions with traffic signals in cities. It would be attached onto the traffic signal poles itself or separately near to the traffic lights and will work in accordance with traffic light signals. It will track each vehicle arriving and leaving the junction, thus helping in detecting various violations. The violations will be detected with the help of machine learning and then challan will be generated with the help of database of vehicle owners from VAHAN services.

The vehicles will be tracked with the concept of Computer Vision, which is a field of Artificial Intelligence that enables computers to extract meaningful information from images and videos. The system will get a low frame rate feed from the camera set-up in a lane and each frame will be operated upon accordingly.

Object detection is a phenomenon in computer vision that encompasses the recognition of various entities in images and/or videos. In order to detect vehicles, VioCam uses YOLO algorithm that stands for ‘You only look once’. YOLO is an algorithm that uses convolutional neural networks (CNN) to detect objects in real-time. The algorithm requires only a single forwards propagation through a neural network to recognize objects. It also provides class probabilities as YOLO considers object detection as a regression problem. The algorithm has excellent learning abilities that assist it to learn the representations of objects and use them in object detection. The algorithm is very fast as it can predict objects in real time and with high accuracy when compared to much slower traditional approaches like fast R-CNN etc as mentioned in fig. 4.

YOLO gives the bounding boxes for every vehicles on the lane. A threshold line can be set-up on the system, which will define the violation of red light signal jumping. The bounding boxes will be constantly monitored by the system and if a bounding box crosses the threshold line then a violation will be recorded.

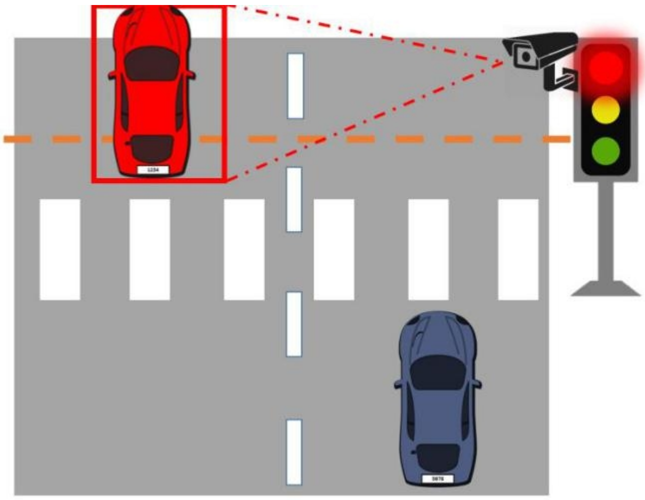

FIGURE 5: Working of VioCam for red signal violation

‘No helmet detection’ will also be done with the help of Computer Vision. YOLO algorithm will again be used for helmet detection. If a vehicle owner is not wearing a helmet, the algorithm will detect it and a violation will be detected.

Once a violation is detected, the number plate of the vehicle is segmented out of the image with the help of computer vision. Optical Character Recognition (OCR) is run on the violator’s vehicle number plate. Using OCR, text written on digital images can be converted in a form that computer can process, store and edit as a text file or as a part of a data entry software.

The numbers are then searched for in the vehicle owner database provided by the VAHAN services or Ministry of Road Transport & Highways. The information from the database is used to generate a challan for the respective violation and send it over to the owner of the vehicle. This is how the whole process of violation detection and challan generation is automated and streamlined using VioCam.

The system also offers traffic congestion control. It keeps a track of number of vehicles in a specific lane. If a lane is very congested and a certain other lane is relatively empty then the traffic lights can be dynamically adjusted. The congested lane can have more green light time and the green light time for the other lanes can be shortened. Similarly, for the red light, the congested lane will have less of a red light time as compared to other lanes.

3.2 Work Breakdown Structure

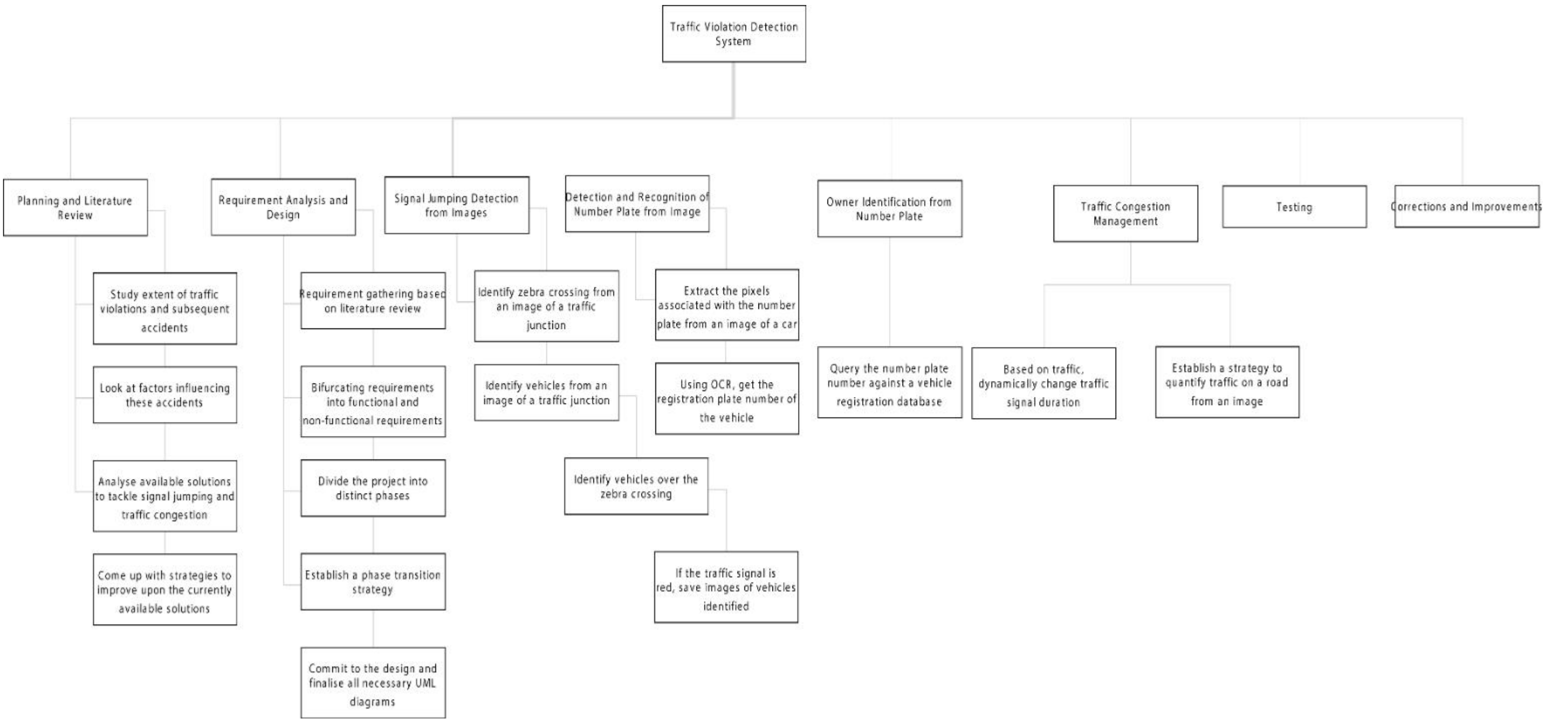

Figure 6: Work Breakdown Structure

The first phase of the project, Planning and Literature review, involves getting clarity on the exact problem at hand and looking at proposed and implemented solutions for the same or a similar problem. The project features were then conceived while making sure that the project builds upon but does not imitate existing literature and also solves the core problem. The features, requirements and design of the project was formalised in the Requirement Analysis and Design phase where all the requirements were classified into either functional or non-functional requirements and subsequently the project was divided into distinct phases. It was important to make sure that single phases of the project were not too specific or too vague as that would either render the idea of phases useless or make transitioning between phases extremely hard. All these design and structural choices were then formulated using various UML diagrams including but not limited to Class Diagram, Swim Lane Diagram, Flow Diagram, Work Breakdown Structure, and Gantt chart.

The next phase, Signal Jumping Detection from Images, was the first development phase dealing with the first key requirement out of the aforementioned four. Here, first the zebra crossing was identified from a video feed of a traffic junction. Then, using a machine-learning model, all vehicles were identified and isolated from the video feed and the vehicles that were over the zebra crossing were tagged (if the signal was red) and the images of these vehicles were stored. In the next phase i.e. Detection and Recognition of Number Plate from Image, the image was further segmented to only get the number plate of the violation-causing vehicles and this number plate was then “read” using OCR. Once the system started extracting the number plate numbers of violators successfully, we then proceeded to query these numbers from a vehicle registration database to identify the violator.

After successfully completing violation detection and management related modules, the focus was shifted towards Traffic Congestion Management where based on the traffic flow at the junction, the traffic signal was changed dynamically. The traffic flow of the junction was captured via the same camera used for violation detection. That marked the last phase of development, after which only testing and Correction of existing features and modules was executed.

3.3 Tools and Technology

A high definition camera - Used to record the footage of the traffic and serves as the eyes of the system.

Some object detection models based on the YOLO architecture, a single shot detector. Each model detects different objects in the image like a vehicle or a helmet etc.

A web server that sends the footage to the cloud built using Flask.

VioCam Dashboard - used to monitor the traffic and display information about the defaulters.

4. DESIGN SPECIFICATIONS

4.1 System Architecture

The system can be explained with a three-tier architecture according to the following architecture diagram (fig.7). A data input is provided with a camera as a video feed. All the storage happens in VioCam system and a control unit processes all the data and does all the tasks.

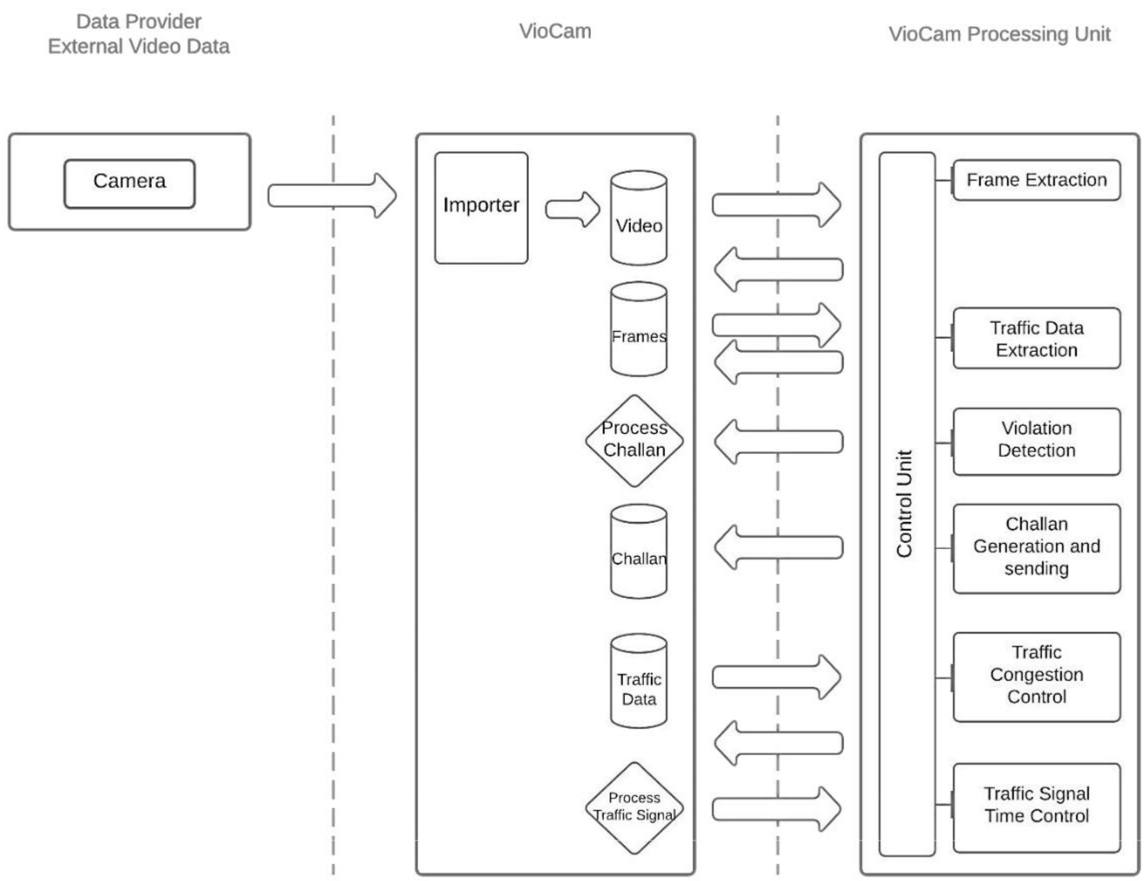

FIGURE 7: Three-tier system architecture

4.2 Design Level Diagrams

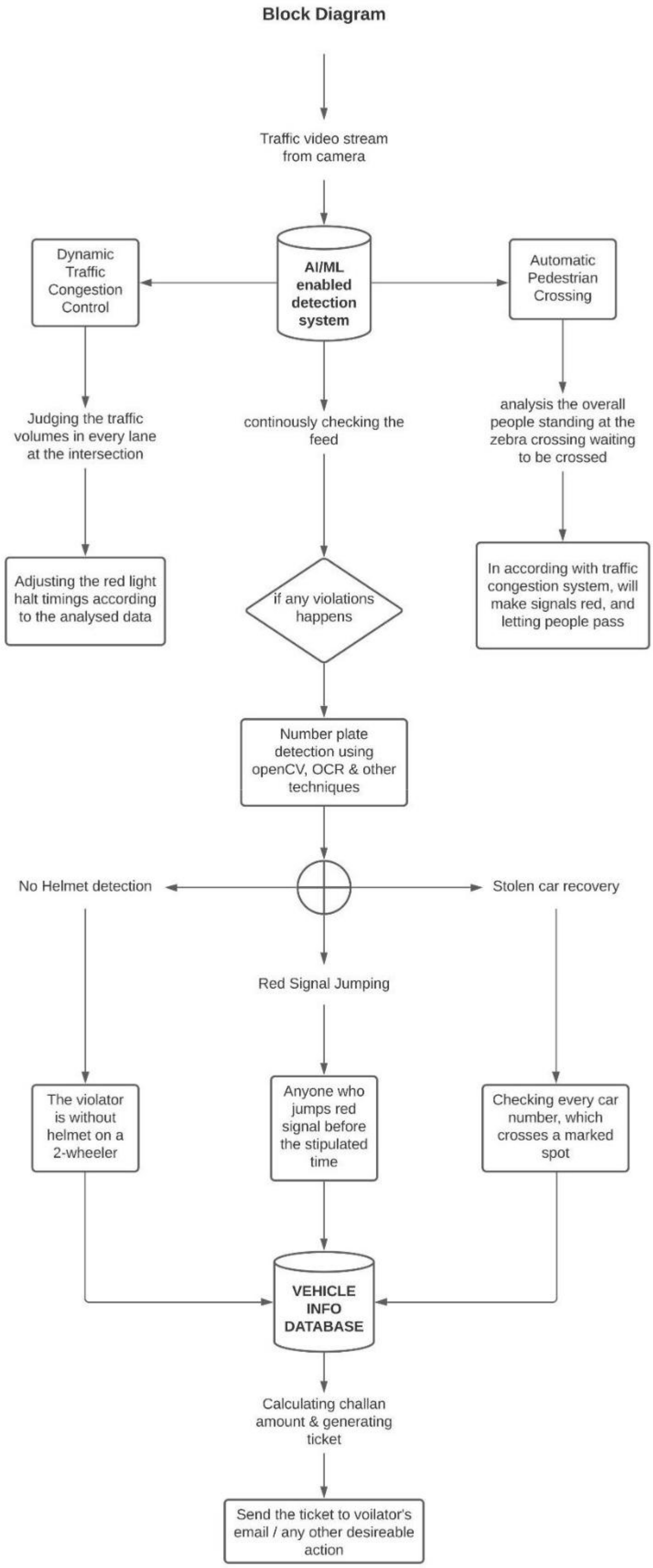

FIGURE 8: Block Diagram

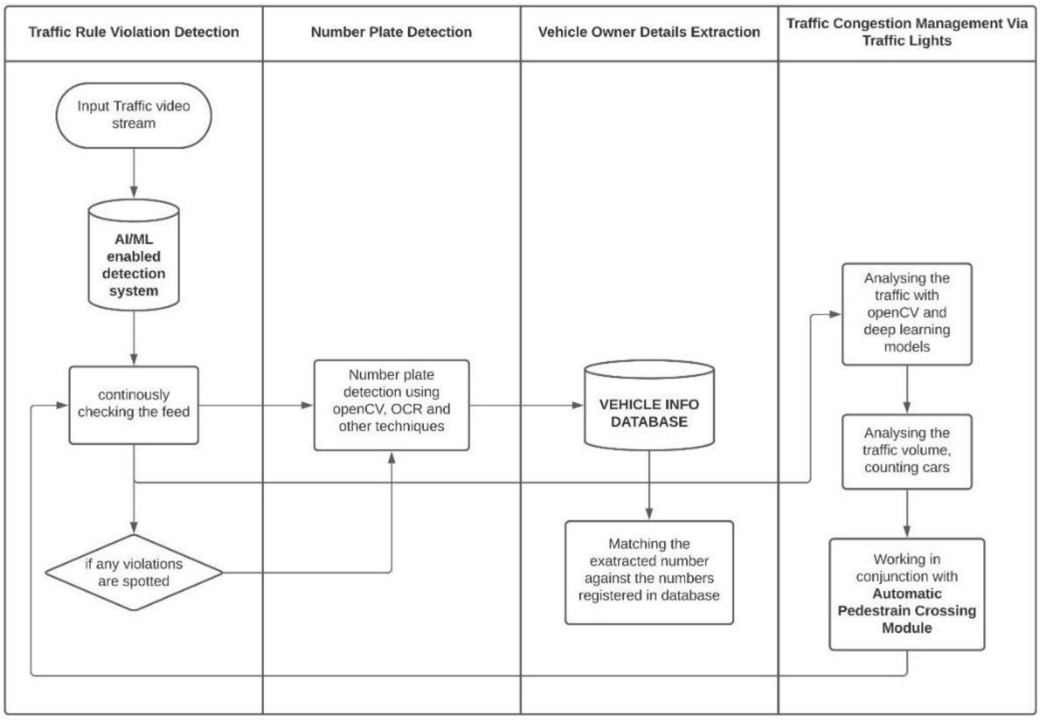

FIGURE 9: Swimlane Diagram

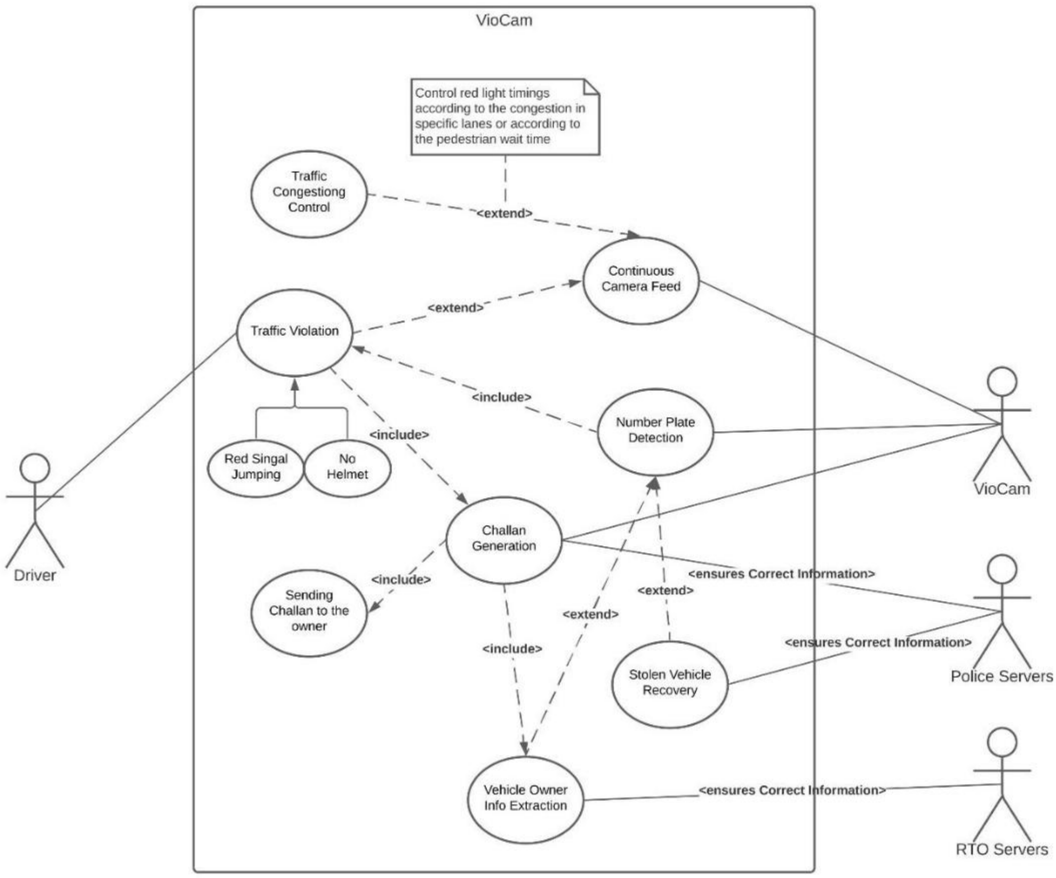

FIGURE 10: Use-case diagram

5. IMPLEMENTATION AND EXPERIMENTAL RESULTS

5.1 Experimental Setup

Trivially, VioCam requires a constant camera feed of traffic to work. As said before, VioCam can be hooked up with the already set-up cameras in traffic junctions and traffic lights.

It is assumed that there is red light currently going on in the video but in actuality, there was green light there. This has been done to showcase the jumping of red light and detection of violation.

In order to showcase the traffic congestion management system, a simulation is made in Python with the help of the PyGame library. The simulation comprises of the bird-view of four roads making a junction. Each lane has its own traffic light control. The system can look at the number of vehicles and use the formula to estimate the green signal timings of each of the traffic light accordingly. The same thing is shown in the simulation.

5.2 Experimental Analysis

5.2.1 Data

As previously mentioned, the project has used the COCO dataset to train machine learning models in order to detect vehicles on the road. The model to detect the vehicle as well as detecting the number plate was trained with the help of YOLO algorithms.

A hand-shot video of traffic light junction from Sector-20, Panchkula, Haryana has been used for showcasing the red light jumping violating. The video was shot from a phone and has a resolution of 1920 x 1080 pixels running at 30 frames/second. It was shot from on top of the Zirakpur-Kalka highway overlooking the traffic lights in Sector 20, Panchkula.

5.3 Working of the project

5.3.1 Procedural Workflow

VioCam is an integrated traffic management system that works with artificial intelligence approaches, mostly deep learning. The system uses a live camera feed to get the traffic images and monitor them. The vehicles are monitored for the basis of violation detection as well as traffic signal management to keep congestion at a minimum. The system also generates a challan automatically for any violation by extracting vehicle owner info from database.

The violation detection system is set-up to detect the violations of red-light jumping and no-helmet riding on two-wheeler vehicles. The system keeps in track of all the vehicles in the frame and constantly checks whether any vehicle is violating the law. In order to detect the red signal violation, a threshold line can be set-up anywhere on the road which would define a vehicle jumping the red light. Whenever a detected vehicle tries to cross the threshold line, it is detected as a violation. The vehicle image is then extracted out and sent for further processing. The image is used to extract out the registration number of the vehicle. The number is then used to find out the owner details from a dummy database. After getting the details, a challan is automatically generated in the form of a pdf which can be sent to the violator.

The traffic congestion management system works by dynamically controlling the green signal timings on each of the lanes. The system already keeps track of vehicles in the frame and it gives out the total number of vehicles in the frame to the traffic congestion management system. It takes into consideration the number of vehicles and the average speed by which a vehicle crosses the road. Then, with a formula green signal time is calculated. This enables the lowering of waiting time at any busy junction with traffic lights compared to the traditional static traffic light system. The whole process can be understood by the flowchart shown in fig 11.

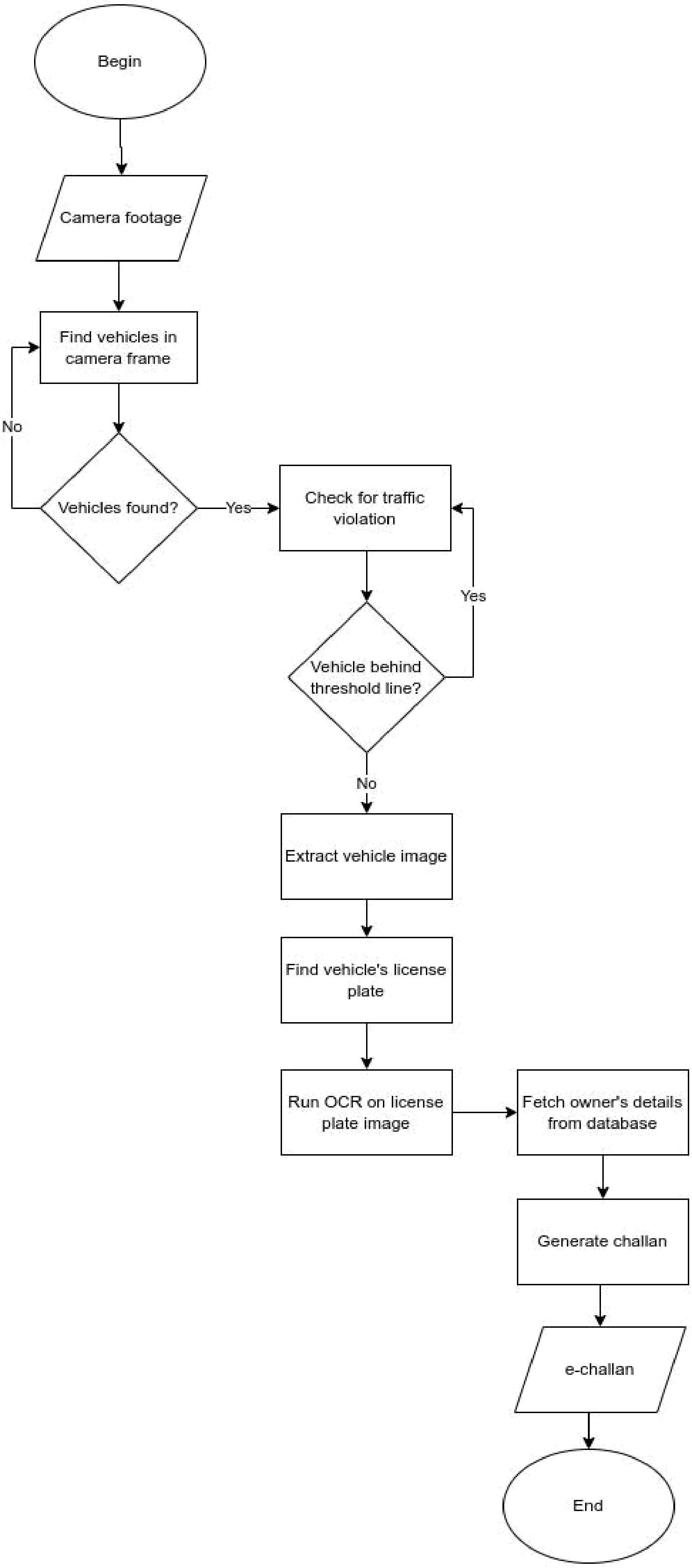

FIGURE 11: Flowchart of VioCam

5.3.2 Algorithmic Approaches Used

YOLOv4 and YOLOv5 have been used in order to train the deep neural networks for the detection of vehicles and number plates. A model pre-trained on the COCO dataset was picked and using transfer learning, a new layer was built on top of that model for detecting cars. Since, the model was already trained on a wide variety of objects from the COCO dataset, it didn't take hours to train the model again to detect vehicles or number plates, it already had some understanding of objects in the real world from its previous training.

The neural network used in the project is as follows:

Name	Filters	Output Dimension
Conv 1	7 x 7 x 64, stride=2	224 x 224 x 64
Max Pool 1	2 x 2, stride=2	112 x 112 x 64
Conv 2	3 x 3 x 192	112 x 112 x 192
Max Pool 2	2 x 2, stride=2	56 x 56 x 192
Conv 3	1 x 1 x 128	56 x 56 x 128
Conv 4	3 x 3 x 256	56 x 56 x 256
Conv 5	1 x 1 x 256	56 x 56 x 256
Conv 6	1 x 1 x 512	56 x 56 x 512
Max Pool 3	2 x 2, stride=2	28 x 28 x 512
Conv 7	1 x 1 x 256	28 x 28 x 256
Conv 8	3 x 3 x 512	28 x 28 x 512
Conv 9	1 x 1 x 256	28 x 28 x 256
Conv 10	3 x 3 x 512	28 x 28 x 512
Conv 11	1 x 1 x 256	28 x 28 x 256
Conv 12	3 x 3 x 512	28 x 28 x 512
Conv 13	1 x 1 x 256	28 x 28 x 256
Conv 14	3 x 3 x 512	28 x 28 x 512
Conv 15	1 x 1 x 512	28 x 28 x 512
Conv 16	3 x 3 x 1024	28 x 28 x 1024
Max Pool 4	2 x 2, stride=2	14 x 14 x 1024
Conv 17	1 x 1 x 512	14 x 14 x 512
Conv 18	3 x 3 x 1024	14 x 14 x 1024
Conv 19	1 x 1 x 512	14 x 14 x 512
Conv 20	3 x 3 x 1024	14 x 14 x 1024

Conv 21	3 x 3 x 1024	14 x 14 x 1024
Conv 22	3 x 3 x 1024, stride=2	7 x 7 x 1024
Conv 23	3 x 3 x 1024	7 x 7 x 1024
Conv 24	3 x 3 x 1024	7 x 7 x 1024
FC1	-	4096
FC2	-	7 x 7 x 30 (1470)

The loss function used by the neural network is as follows:

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{obj} (x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2$$

The challan generation and owner detail extraction has been done with the help of Pandas and FPDF libraries in Python. This was done because the government vehicle registration database could not accessed. But, it can be safely assumed when deployed the project will have access to the government database and would be able to work with more traditional DBMS approaches like in SQL or NoSQL.

The green signal timing to dynamically control traffic lights has been calculated with the formula:

$$GST = \frac{NoOfVehicles * AverageTime}{NoOfLanes + 1}$$

5.3.3Working Screenshots

As can be clearly seen in fig.12, the model can identify vehicles with great accuracy. It has successfully identified two-wheeler as well as three-wheeler vehicles. As mentioned in the text before, YOLO algorithm is being used here in order for vehicle detection. The model was trained using the COCO dataset and it works with any types of vehicles.

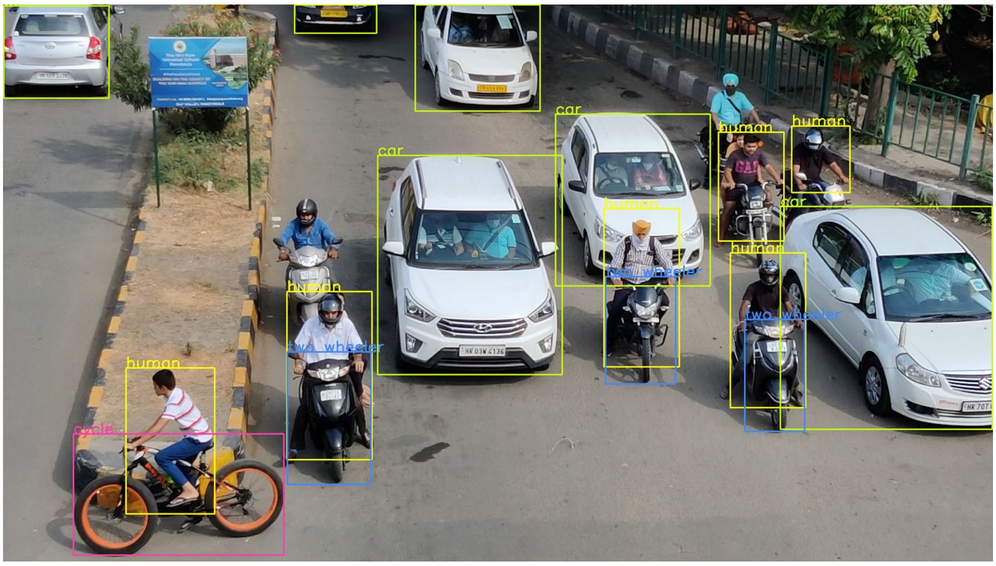

FIGURE 12: Object detection

A threshold line(green line in fig. 13(a)) can be set-up on the road at some point and any vehicle crossing the threshold line is declared as a violator. The vehicles are constantly monitored and if any bounding box of any vehicle crosses the threshold line it is declared as a violator and the bounding box of the vehicle turn red stating that it has violated the red signal law as shown in fig. 13(b).

FIGURE 13(a): Threshold line for red-signal violation detection

FIGURE 13(b): The red-signal violating vehicles (shown in red bounding box)

Helmet detection can also be done using YOLO algorithms. The screenshot of helmet detection is shown in fig. 14(a) and 14(b). If a two-wheeler vehicle driver is not wearing a helmet, it is declared a violation.

FIGURE 14(a): Helmet detection

FIGURE 14(b): Helmet detection with violet bounding box

Number plate detection and extraction of the image of number plate for recognizing the digits on the number plate is shown in fig 15.

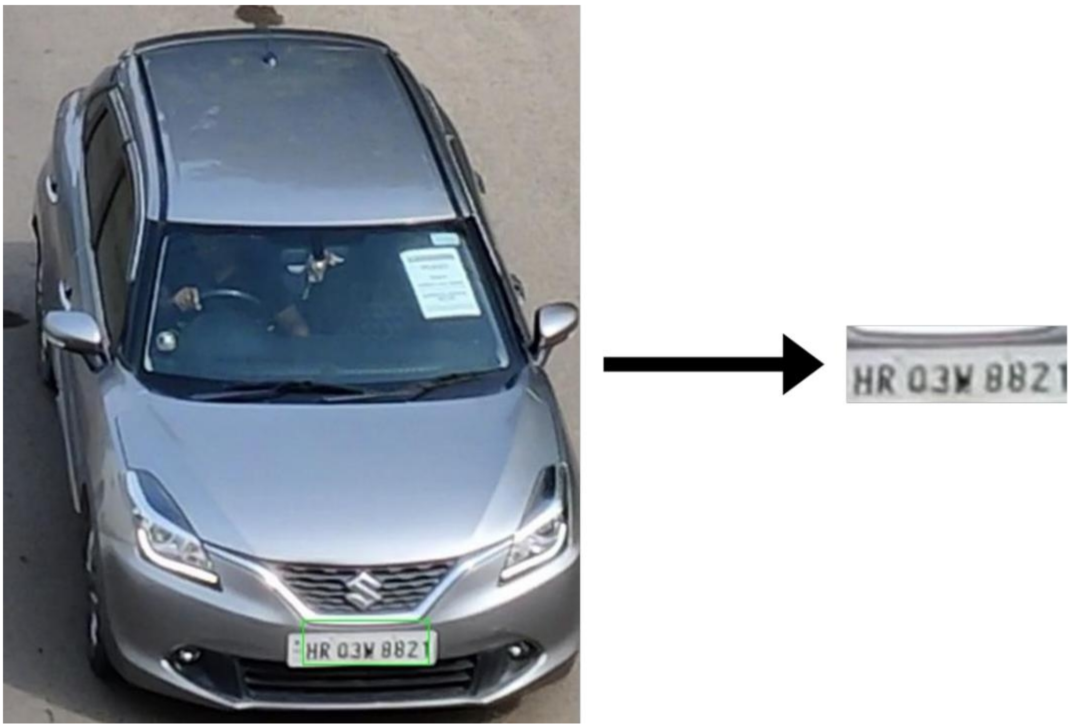

FIGURE 15: Number plate detection and extraction

The simulation depicting the dynamic traffic light controlling system has been shown in fig 16.

FIGURE 16: Simulation of dynamic traffic light

Automatic challan generation is done with the Pandas and FPDF library of Python with a dummy dataset.It is shown in fig 17.

TRAFFIC VIOLATION CHALLAN

VIOLATION TYPE: RED LIGHT VIOLATION

OWNER NAME: XIACFL IKGPR

OWNER PHONE: 9572194118

REGISTRATION NO: CH28KQ4018

VEHICLE MANUFACTURER: HYUNDAI

VEHICLE TYPE: CAR

CHALLAN AMOUNT: INR xxxx

PAYMENT: Pending

IMPOUND DOCUMENT: No Impound

FIGURE 17: Generated pdf of the challan

5.4 Testing Process

5.4.1 Test Plan

To test the process of violation detection, the system would be using two different footages of live traffic. The footage would be captured from a mobile phone with the resolution of 1920x1080 pixels running at 30 frames per second. The footage would be captured from as close a position to the actual CCTV cameras as there could be. The footage captured would include the red light violation as well as no-helmet riding violation in it. If somehow the red-light violation is not captured, we would be assuming that the vehicles running at the green light are actually violators trying to run over the red light.

The automatic challan generation would be tested with the help of a dummy dataset created with owner details, vehicle type, vehicle manufacturer etc. The dummy dataset would be created using Python libraries and would be exported a CSV file.

In order to test the dynamic traffic signal management, a simulation will be made with the help of PyGame library in Python. The simulation would be run multiple times with dynamic as well as static traffic signal control for comparison purposes.

5.4.2 Features Tested

- Vehicle detection
- Number of vehicles detected
- Red-light violation detection
- No-helmet violation detection
- Number plate detection
- Number plate OCR
- Vehicle owner details extraction
- Automatic challan generation
- Dynamic traffic light control

5.4.3 Test Techniques

- Unit Testing: Each part of the system, from smallest parts like loops not working or going in an infinite spiral etc. to single units of code tested in this part.
- Integration Testing: All the single units of code combined to make a program and the integrity of the system is tested by observing if all the features work together.
- System Testing: The whole system was tested on Linux based Operating Systems-Ubuntu, Kali and Windows Operating Systems-Windows 10, Windows 11.

5.4.4 Test Cases

Test Case 1: Non-violating/idle vehicle detection

Test Case 2: Red light violation with car

Test Case 3: Red light violation with bike/two-wheeler

Test Case 4: No-helmet detection

Test Case 5: Search for number plate

Test Case 6: Extract out information for violator from database

Test Case 7: No information for the violator

Test Case 8: Challan generation with the violator information

5.4.5 Test Results

TABLE 5: Test case results and status

Test Case	Expected Result	Actual Result	Status
1	Detect all types of vehicles in idle state	Vehicles detected with green bounding box	Passed
2	Detect car with violation	Vehicle detected with red bounding box	Passed
3	Detect bike with violation	Bike detected with red bounding box	Passed
4	Detect helmet on two-wheeler vehicles	Helmet detected with red bounding box	Passed
5	Detect number plate and recognize the numbers on it	Number plate detected with green bounding box and numbers recognized	Passed
6	Return vehicle info and owner info from database	Returned vehicle info and owner info	Passed
7	Return ‘No info’ with violator picture	Returned ‘No info’ on terminal with violator picture	Passed
8	Return pdf of challan with all info	Returned pdf of challan with all info	Passed

5.5 Results and Discussion

The process of identifying a traffic violation begins with isolating all vehicles in each frame of the video feed. As seen in Fig. 12, the model detected 15 out of 15 vehicles in the frame.

Then, it is checked if any of the bounding boxes is intersecting with the threshold line defined for red light violation. Fig. 13(b) shows around 7 detected vehicles with green bounding boxes in the frame. A dummy threshold line is setup just ahead of the zebra crossing and two vehicles violating the red light have been detected with red bounding boxes.

If a vehicle is found to be in violation, then the number plate is separated out of the vehicle and is passed to the OCR system as shown in Fig. 15. The vehicle registration number is then searched in the vehicle registry database to find out the owner details.

Violation detection system is also accompanied by a congestion managementsystem. The system works by taking into consideration the number of vehiclesand the average speed of each vehicle in order to calculate the green signaltime for each lane. Compared to the traditional static traffic signal system, thissystem reduced the average wait times by up to 70%.

5.6 Inferences Drawn

The main inference from the project is that we can accomplish multiple functionalities through a single camera. Managing traffic and detecting violations through a single camera is a novel idea and it has shown performance that competes with systems using multiple high-cost cameras. Expanding on the point of cameras, our project also uses a low frame rate and low-resolution video footage which implies that our project can run with a relatively cheap camera. Better violation detection alongside congestion management makes our project ideal for smooth traffic flow which is highly scalable at a very affordable price.

5.7 Validation of objectives

TABLE 6: Status of objectives

S.No.	Objective	Status
1	Traffic rule violation detection	Successful
2	Number plate detection	Successful

3	Vehicle owner details extraction	Successful
4	Automatic challan generation	Successful
5	Traffic congestion management via traffic lights	Successful

6. CONCLUSIONS AND FUTURE SCOPE

6.1 Conclusions

VioCam is an integrated system that identifies vehicles and detects violations (red-signal and no-helmet violation) using computer vision. It can also generate challan automatically according to the violations. In order for challan generation, VioCam uses computer vision and OCR to get the numbers from the number registration plate of the vehicle and the owner details are fetched from the vehicle-owner database. Similar systems are already being used in some states of India as well as abroad, almost all of them using multiple cameras instead of just one camera needed with VioCam. Various research papers and projects were found using variety of techniques for traffic violation control, like OpenCV, genetic algorithms and deep learning. VioCam also controls the traffic congestion by dynamically controlling the traffic light timings with intelligent algorithms to calculate the timings of green light.

6.2 Environmental, Social and Economic Benefits

- Since violators can be identified automatically, an e-challan can be generated for the same. This reduces the use of paper and subsequently helps reduce the carbon footprint of the nation.
- Our project aims to provide various violation detection and traffic congestion features through a single camera, which currently is being done using 4-5 cameras. This reduces points of failures, cost and technical hassle of setting up the system at a junction.
- Widespread use of VioCam can lighten the burden on traffic police to be present throughout the day at major junctions. Instead, a single officer can review the footage from multiple junctions from a suitable location.

6.3 Future Work Plan

- In the future, we plan to add more traffic violations other than red-signal and no-helmet violations.
- We plan to keep improving the accuracy rate of the system by training the models used on more data.
- We will also continually make the system faster by optimising our source code.

7. Project Metrics

7.1 Challenges Faced

Getting CCTV footage: The first problem we faced was sourcing a CCTV footage of a traffic intersection for training and testing the model for detection of vehicles and violations.

Number plate detection: Orientation of the vehicles with respect to the camera footage was sometimes not optimal enough for the model to detect and extract out the numbers from the number plate. In some instances, the models couldn't even detect the number plate in the first place.

Getting the vehicle owner database: We couldn't find a public dataset linking car registration numbers with their owners.

Functionality in low-light and extreme weather conditions: The model couldn't be tested in all the varying weather and light conditions and hence the functionality will probably suffer in extreme conditions.

7.2Relevant Subjects

TABLE 7: Relevant Subjects used in the project

Subject Code	**Subject Name**	**Description**
UCS655	AI APPLICATIONS - NLP, COMPUTER VISION, IOT	Object Detection
UML501	MACHINE LEARNING	Deep Neural Network
UCS538	DATA SCIENCE FUNDAMENTALS	Feature Extraction
UCP402	ARTIFICIAL INTELLIGENCE	Optical Character Recognition
UCP401	SOFTWARE ENGINEERING	UML Diagrams
UCS757	BUILDING INNOVATIVE SYSTEMS	Integration and testing

7.3 Interdisciplinary Knowledge Sharing

Varying knowledge from different types of disciplines was used for the project. We had to study deep neural networks to work on object detection models. Software engineering was used in creating whole of this document with the help of UML diagrams. We also had to look up how the traffic lights work which required us to study a variety of new topics ranging from electronics to mathematics. Finally, to bring all the components of the project together we had to have knowledge of team collaboration and work sharing which lifted our spirits of camaraderie.

7.4 Peer Assessment Matrix

		Evaluation of		
		Aryan	Tushar	Aditya
Evaluation by	Aryan	5	5	5
	Tushar	4	4	5
	Aditya	4	4	4

7.5 Role Playing and Work Schedule

Tushar: Helmet detection, Violation detection, Optimization, Testing, Documentation

Aryan: Vehicle detection, Image pre-processing, Number plate detection, Testing, Documentation

Aditya: Optical Character Recognition, Dynamic traffic light control/congestion management, Optimization, Documentation

Everyone worked approximately 5 hours per week on the weekends for each month since the starting of development of the project.

7.6 Student Outcome Description and Performance Indicators

TABLE 8: Student outcomes description

SO	SO Description	Outcome
1.1	Ability to identify and formulate problems related to computational domain	Identified the problem of traffic congestion on traffic lights. The challenge was to reduce the waiting time.

1.2	Apply engineering, science, and mathematics body of knowledge to obtain analytical, numerical, and statistical solutions to solve engineering problems.	Used computer science concepts like machine learning, artificial intelligence, deep neural network, etc. to make an integrated program.
2.1	Design computing system(s) to address needs in different problem domains and build prototypes, simulations, proof of concepts, wherever necessary, that meet design and implementation specifications.	Designed an integrated fully automated system for traffic management. Also, simulated the flow of traffic for testing of dynamic traffic light duration.
3.1	Prepare and present variety of documents such as project or laboratory reports according to computing standards and protocols.	Prepared several UML diagrams and learned to make a proper documentation for projects.
3.3	Able to communicate effectively with peers in well organized and logical manner using adequate technical knowledge to solve computational domain problems and issues.	Worked in a team of people guided by a mentor to co-ordinate and solve the problem that we were working upon.
4.3	Evaluate computational engineering solutions considering environmental, societal, and economic contexts.	Identified various advantages that were achieved with this project in various contexts.
5.1	Participate in the development and selection of ideas to meet established objective and goals.	Tried and tested several different machine learning models for identifying vehicles at fast rate.
5.2	Able to plan, share and execute task responsibilities to function effectively by creating collaborative and inclusive environment in a team.	Each and every idea and task was criticized upon and reviewed by all the members of the team to enable a collaborative and inclusive environment.
6.1	Ability to perform experimentations and further analyze the obtained results.	Found several research papers on similar topics and worked upon

		them to make a more optimal project.
6.2	Ability to analyse and interpret data, makenecessary judgement(s) and draw conclusion(s).	Used transfer learning approaches on model trained for COCO dataset to identify vehicles.
7.1	Able to explore and utilize resources to enhance self-learning.	Learned many new concepts in machine learning and deep learning to implement in the project.

7.7 Brief Analytical Assessment

Q1. What sources of information did your team explore to arrive at the list of possible Project Problems?

A1. We always had the idea that we wanted to something in the field of machine learning. We were exploring several websites to find out problems that the world is facing like energy crisis, global warming etc. but we weren't able to find a really solid topic to start a project with. Then, we looked at all the previous projects we had done together as a team and remembered all the things that we had planned to do and couldn't start. We remembered when we were visiting Chandigarh, we saw a bunch of cameras for traffic monitoring and one of us realised we can all this can be done with just a single camera. Then it struck us, to make an automated traffic management system that could work with just a single camera feed. Our mentor guided us through and gave us the input of dynamic traffic lights depending on the amount of traffic that could be added onto the project.

Q2. What analytical, computational and/or experimental methods did your project team use to obtain solutions to the problems in the project?

A2. We have used quite a few methods to train the deep neural network like the YOLO system for object detection. But the one which is worth mentioning is the transfer learning approach that we used. We did not have really powerful systems to train the deep neural networks from scratch. We picked up the models trained on the COCO dataset which is a dataset containing various categories of visual images and we used transfer learning to train it to detect vehicles for our system. The same approach was used while detecting number plates where we picked up the encodings for number plate detection and moulded it to work on Indian number plates.

Q3. Did the project demand demonstration of knowledge of fundamentals, scientific and/or engineering principles?

A3. Yes, the project demanded the knowledge of quite a few engineering principles. We always had to keep in mind the testability of the components we were developing. Every component of the project was made in such a way to hold up to with all the testing processes. The project also required modularity in terms of development, so that it can be easily updated and upgraded going on in the future. We also had to keep in mind the efficiency of the project. It was to be made the most optimal to run in real-time with all the traffic data.

Q4. How did your team shares responsibility and communicate the information of schedule with others in team to coordinate design and manufacturing dependencies?

A4. Our team was really comfortable working with each other because of the previous experiences we had in hackathons and other projects. Each one of us knew what domain of the project they had to work upon and the work was equally split according to their area of expertise. Once a member made a component of the project, the whole team would gather in an online meeting and would have a criticising as well as reviewing process to enable equal participation from all. It was a collaborative process where anyone could give an input at any step in the project development. Everyone of us was supposed to free several hours of their schedule on weekends to work towards the project.

Q5. What resources did you use to learn new materials not taught in class for the course of the project?

A5. We used various resources to learn new topics needed for this project. We used online course providing web-sites like Coursera to learn new things from scratch and build up a theoretical knowledge. Coursera also helped with practical knowledge of various topics with the "Guided Project" feature which taught us a bunch of elementary projects to get familiar with new territory. Then, the documentation of various libraries and software really helped a lot in the coding part of the project. Apart from this, we also learned a lot by studying the code of various related projects on GitHub.

Q6. Does the project make you appreciate the need to solve problems in real life using engineering?

A6. It can really be appreciated that the project we developed can be used to solve real life problems. It is not just a fun project that we developed to put on our resume but it had a positive impact on a problem. We cherished these moments where we observed that the things we have studied in the 4 years of engineering can be racked up to create an optimal solution to a real life problem and it really is an amazing feeling.

REFERENCES

[1] Road Traffic injuries - WHO | World Health Organization Data

[2] Stoohs, R. A., Guilleminault, C., Itoi, A., & Dement, W. C. (1994). Traffic accidents in commercial long-haul truck drivers: the influence of sleep-disordered breathing and obesity. *Sleep*, *17*(7), 619–623.

[3] M. Abdel-Aty, “Analysis of driver injury severity levels at multiple locations using ordered probit models,” Journal of Safety Research, vol. 34, no. 5. Elsevier BV, pp. 597–603, Jan. 2003. doi: 10.1016/j.jsr.2003.05.009.

[4] GM Saltzman, MH Belzer “Truck driver occupational safety and health; 2003 conference report and selective literature review”.

[5] L.-Y. Chang and F. Mannering, “Analysis of injury severity and vehicle occupancy in truck- and non-truck-involved accidents,” Accident Analysis & Prevention, vol. 31, no. 5. Elsevier BV, pp. 579–592, Sep. 1999. doi: 10.1016/s0001-4575(99)00014-7.

[6] National Crime Records Bureau, Ministry of Road Transport & Highway, Law commission of India, Global status report on road safety http://tripp.iitd.ac.in/assets/publication/Road_Safety_in_India2018.pdf

[7] Press Information Bureau, Ministry of Road Transport and Highways, Government of India; 20thOctober 2011 https://morth.nic.in/

[8] D. Mohan, “Road traffic injuries: a stocktaking,” Best Practice & Research Clinical Rheumatology, vol. 22, no. 4. Elsevier BV, pp. 725–739, Aug. 2008. doi: 10.1016/j.berh.2008.05.004.

[9] R. A. Retting, R. G. Ulmer, and A. F. Williams, “Prevalence and characteristics of red light running crashes in the United States,” Accident Analysis & Prevention, vol. 31, no. 6. Elsevier BV, pp. 687–694, Nov. 1999. doi: 10.1016/s0001-4575(99)00029-9

[10] Mahalank, S.N., Malagund, K.B., &Banakar, R.M. (2016). Device to device interaction analysis in IoT based Smart Traffic Management System: An experimental approach. *2016 Symposium on Colossal Data Analysis and Networking (CDAN)*, 1-6.

[11] Balid, W., Tafish, H., &Refai, H. (2016). “Versatile real-time traffic monitoring system using wireless smart sensors networks. 2016 IEEE Wireless Communications and

[12] D. Shinar and R. Compton, “Aggressive driving: an observational study of driver, vehicle, and situational variables,” Accident Analysis & Prevention, vol. 36, no. 3. Elsevier BV, pp. 429–437, May 2004. doi: 10.1016/s0001-4575(03)00037-x.

[13] S. Smith, M. Carrington, and J. Trinder, “Subjective and predicted sleepiness while driving in young adults,” Accident Analysis & Prevention, vol. 37, no. 6. Elsevier BV, pp. 1066–1073, Nov. 2005. doi: 10.1016/j.aap.2005.06.008.

[14] S. Hajeb, M. Javadi, S. M. Hashemi, en P. Parvizi, “Traffic Violation Detection System based on RFID”, 10 2013.

[15] R. Franklin en M. Mohana, “Traffic Signal Violation Detection using Artificial Intelligence and Deep Learning”, 06 2020, bll 839–844.

[16] K. Klubsuwan, W. Koodtalang and S. Mungsing, "Traffic Violation Detection Using Multiple Trajectories Evaluation of Vehicles," 2013 4th International Conference on Intelligent Systems, Modelling and Simulation, 2013, pp. 220-224, doi: 10.1109/ISMS.2013.143.

[17] A. C. P. Uy et al., "Automated traffic violation apprehension system using genetic algorithm and artificial neural network," 2016 IEEE Region 10 Conference (TENCON), 2016, pp. 2094-2099, doi: 10.1109/TENCON.2016.7848395.

[18] S. Asadianfam, M. Shamsi, and A. Rasouli Kenari, “Big data platform of traffic violation detection system: identifying the risky behaviors of vehicle drivers,” Multimedia Tools and Applications, vol. 79, no. 33–34. Springer Science and Business Media LLC, pp. 24645–24684, Jun. 23, 2020. doi: 10.1007/s11042-020-09099-8.

[19] A. Kumar, S. Kundu, S. Kumar, K. Tiwari, en J. Kalra, “S-TVDS: Smart Traffic Violation Detection System for Indian Traffic Scenario”, International Journal of Innovative Technology and Exploring Engineering, vol 8, bll 2278–3075, 03 2019.

[20] Xiaoling Wang, Li-Min Meng, Biaobiao Zhang, Junjie Lu and K. . -L. Du, "A video-based traffic violation detection system," Proceedings 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC), 2013, pp. 1191-1194, doi: 10.1109/MEC.2013.6885246.

[21] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat and S. Venkatraman, "Deep Learning Approach for Intelligent Intrusion Detection System," in IEEE Access, vol. 7, pp. 41525-41550, 2019, doi: 10.1109/ACCESS.2019.2895334.

[22] Abhineet Sharma, Robotics Made Easy Using Arduino, Robosapians Publications

[23] *Tutorials from the given websites*, [online] Available: http://1sheeld.com/tutorials/

[24] Raspberry Pi | https://simple.wikipedia.org/wiki/Raspberry_Pi

[25] Raspberry Pi | https://www.raspberrypi.org/products/raspberry-pi-4-model-b/

[26] OpenCV Documentation | www.opencv.org

[27] Python Documentation | www.python.org

[28] YOLO | YOLO: Real-Time Object Detection (pjreddie.com)