**Institute of Engineering**

**National College of Engineering**

**Parking Violation for Single Vehicle, Lane and Retrogress Violation Detection Using Object Detection and Tracker**

**Submitted By: Submitted To:**

Aashish Tamrakar (79101) Department of Computer

Asim Aryal (79104) Engineering

Niroj Bajracharya (79121) (National College of Engineering)

Sudip Gyawali (79138) Date: 7/21/2021

**Supervised By**: Er. Anup Shrestha

**(On the partial requirement of fulfillment of study on Computer Engineering)**

**Parking Violation for Single Vehicle, Lane and Retrogress Violation Detection Using Object Detection and Tracker**

Submitted by:

Aashish Tamrakar [79101]

Asim Aryal [79104]

Niroj Bajracharya [79121]

Sudip Gyawali [79138]

Supervised by:

Er. Anup Shrestha

**MINOR PROJECT REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR IN COMPUTER ENGINEERING**

Submitted to:

Department of Computer Engineering

National College of Engineering

Talchhikhel, Lalitpur

MARCH 2021

# Copyright

The author has agreed that the Library, Department of Computer Engineering, National College of Engineering and Institute of Engineering may make this report freely available for inspection. Moreover, the author has agreed that for extensive copying of this project report for scholarly purpose may be granted by the supervisors who supervised the project work record herein or, in their absence, by the Head of Department wherein the project report was done. It is understood that the recognition will be given to the author of this report and to the Department of Computer Engineering, National College of Engineering, in any use of the material of this project report, copying or publication or the other use of this report for financial gain without approval of the Department of Computer Engineering, National College of Engineering and author’s written permission is prohibited.

Request for the permission to copy or to make any other use of the material in this report in whole or in part should be address to.

**Er.** **Pradip Adhikari**

Head of Deparment

Department of Computer Engineering

National College of Engineering

Talchhikhel, Lalitpur

Nepal

**National College of Engineering**

**“Department of Computer Engineering”**

Approval Letter

The undersigned certify that they have read and recommended to the Institute of Engineering for acceptance, a project entitled “**Parking Violation for Single Vehicle, Lane and Retrogress Violation Detection Using Object Detection and Tracker**” submitted by **Aashish Tamrakar**, **Asim Aryal**, **Niroj Bajracharya**, and **Sudip Gyawali** in partial fulfillment for the degree of Bachelor of Computer Engineering.

………………………

Supervisor

Er. Anup Shrestha

Lecturer

Department of Computer Engineering

………………………

External Examiner

………………………

Er. Pradip Adhikari

Head of Deparment

Department of Computer Engineering

Date: March 2021

# Abstract

This project titled Parking Violation for Single Vehicle, Lane and Retrogress Violation Detection Using Object Detection and Tracker is an application, that is able to determine the violators of specific traffic rules from the CCTV stream. The project is developed in Python using OpenCV library and is able to detect lane violations, parking offence and retrogress in a specific scenario, and give the violation footage to the operator in case of Lane Violation and picture in case of Parking Violation. User can choose to perform lane or parking violation. Region of Interest (ROI) is selected manually by the user. After ROI is selected object detection and tracking is performed. YOLO is applied to detect and classify the vehicles in lane violation. This will also generate bounding boxes. MOSSE tracker is used for tracking single object in parking violation. Deep SORT algorithm is used to track many vehicles in lane violation. While, the Hough Transform is used to detect lane lines. Some logic is applied on the basis of tracked bounding boxes. After the violation is detected the footage of the event and a picture is recorded by the system and stored inside a storage system in case of lane and a picture in case of parking violation. Traffic Department would be able to implement it to detect certain violations and catch the offenders.

Keywords: **OpenCV, Region of Interest, YOLO, MOSSE, Deep SORT, Hough Transform**

# Table of Contents

[Copyright ii](#_Toc66392758)

[Approval Letter iii](#_Toc66392759)

[Abstract iv](#_Toc66392760)

[Table of Contents v](#_Toc66392761)

[List of Tables viii](#_Toc66392762)

[List of Figures ix](#_Toc66392763)

[List of Abbreviations x](#_Toc66392764)

[Chapter 1. Introduction 1](#_Toc66392765)

[1.1. Background 1](#_Toc66392766)

[1.2. Evolution of traffic detection system 3](#_Toc66392767)

[1.2.1. Latest status of traffic violation detection system 3](#_Toc66392768)

[1.2.2. Benefits 3](#_Toc66392769)

[1.3. Existing System 5](#_Toc66392770)

[1.3.1. A video-based traffic violation detection system 5](#_Toc66392771)

[1.3.2. Traffic Violation Detection System based on RFID 5](#_Toc66392772)

[1.3.3. Traffic Rules Violation Detection with Computer Vision 6](#_Toc66392773)

[1.3.4. Intersection Violation Detection by Hikvision 7](#_Toc66392774)

[1.4. Customer’s perspective 8](#_Toc66392775)

[1.5. Observation of Existing System 8](#_Toc66392776)

[1.5.1. Common Features 10](#_Toc66392777)

[1.6. Problem Statement 10](#_Toc66392778)

[1.7. Aims & Objectives 12](#_Toc66392779)

[1.7.1. Aims 12](#_Toc66392780)

[1.7.2. Objectives 12](#_Toc66392781)

[1.8. Significance of Study 12](#_Toc66392782)

[1.9. Project Outline 13](#_Toc66392783)

[Chapter 2. Literature Review 14](#_Toc66392784)

[Chapter 3. Software Requirements Specification 29](#_Toc66392785)

[3.1. Revision History 29](#_Toc66392786)

[3.2. Introduction 29](#_Toc66392787)

[3.2.1. Purpose 29](#_Toc66392788)

[3.2.2. Document Conventions 29](#_Toc66392789)

[3.2.3. Intended Audience and Reading Suggestions 29](#_Toc66392790)

[3.2.4. Product Scope 30](#_Toc66392791)

[3.2.5. References 30](#_Toc66392792)

[3.3. Overall Description 30](#_Toc66392793)

[3.3.1. Product Perspective 30](#_Toc66392794)

[3.3.2. Product Function 31](#_Toc66392795)

[3.3.3. User Classes and Characteristics 31](#_Toc66392796)

[3.3.4. Operating Environment 31](#_Toc66392797)

[3.3.5. Design and Implementation Constraints 32](#_Toc66392798)

[3.3.6. User Documentation 32](#_Toc66392799)

[3.3.7. Assumptions and Dependencies 32](#_Toc66392800)

[3.4. External Interface Requirements 34](#_Toc66392801)

[3.4.1. User Interfaces 34](#_Toc66392802)

[3.4.2. Hardware Interfaces 34](#_Toc66392803)

[3.4.3. Software Interfaces 34](#_Toc66392804)

[3.4.4. Communication Interfaces 34](#_Toc66392805)

[3.5. System Features 35](#_Toc66392806)

[3.5.1. Detect Parking Violation 35](#_Toc66392807)

[3.5.2. Detect Lane Violation 36](#_Toc66392808)

[3.6. Other Nonfunctional and Miscellaneous Requirements 37](#_Toc66392809)

[3.6.1. Performance Requirements 37](#_Toc66392810)

[3.6.2. User Friendly 37](#_Toc66392811)

[3.6.3. Extensibility 37](#_Toc66392812)

[3.6.4. Integrability 37](#_Toc66392813)

[3.6.5. Maintainability 37](#_Toc66392814)

[3.6.6. Legal Requirements 38](#_Toc66392815)

[3.7. Appendix 38](#_Toc66392816)

[3.8. Justification of Software Process 40](#_Toc66392817)

[Chapter 4. System Analysis 41](#_Toc66392818)

[4.1. Activity Diagram of main processes 41](#_Toc66392819)

[4.1.1. Parking Violation 41](#_Toc66392820)

[4.1.2. Lane Violation 42](#_Toc66392821)

[4.2. Domain Model Diagram 43](#_Toc66392822)

[4.3. Sequential Diagram 44](#_Toc66392823)

[Chapter 5. System Design 45](#_Toc66392824)

[5.1. UI Design 45](#_Toc66392825)

[5.2. Process Design: Interaction Diagram 46](#_Toc66392826)

[5.3. Data Design 47](#_Toc66392827)

[5.4. Class Diagram 48](#_Toc66392828)

[Chapter 6. Implementation 49](#_Toc66392829)

[6.1. Mapping Design to Code 49](#_Toc66392830)

[6.2. Main Class with Functions 51](#_Toc66392831)

[6.3. Main Algorithm Implementation 54](#_Toc66392832)

[6.3.1. Parking Violation Detection for single vehicle 54](#_Toc66392833)

[6.3.2. Lane Violation Detection 56](#_Toc66392834)

[6.3.3. Algorithms Description 61](#_Toc66392835)

[Chapter 7. Unit Testing 65](#_Toc66392836)

[7.1. Unit Testing of Parking Violation 65](#_Toc66392837)

[7.1.1. Unit Testing of YOLO module 66](#_Toc66392838)

[7.1.2. Unit Testing of Detector Tracker module 67](#_Toc66392839)

[7.1.3. Unit Testing of Parking Violation Logic module 68](#_Toc66392840)

[7.2. Unit Testing of Lane Violation 69](#_Toc66392841)

[7.2.1. Unit Testing of Lane Violation Logic Module 72](#_Toc66392842)

[7.2.2. Unit Testing of Lane Processing Module 73](#_Toc66392843)

[7.2.3. Unit Testing of Vehicles Module 75](#_Toc66392844)

[7.2.4. Unit Testing of GUI Lane Module 76](#_Toc66392845)

[7.3. Miscellaneous Unit Testing 77](#_Toc66392846)

[7.3.1. Unit Testing of Region of Interest Module 77](#_Toc66392847)

[Chapter 8. System Testing 79](#_Toc66392848)

[8.2. Parking Violation System Testing 82](#_Toc66392849)

[8.3. Lane Violation System Testing 82](#_Toc66392850)

[Chapter 9. Discussions 83](#_Toc66392851)

[Chapter 10. Conclusions 85](#_Toc66392852)

[Chapter 11. References 86](#_Toc66392853)

# List of Tables

[Table 1. Table of Comparisons 8](#_Toc66392854)

[Table 2. Unit Testing of Parking Violation module 65](#_Toc66392855)

[Table 3. Unit Testing of YOLO module 66](#_Toc66392856)

[Table 4. Unit Testing of Detector Tracker module 67](#_Toc66392857)

[Table 5. Unit Testing of Parking Violation Logic module 68](#_Toc66392858)

[Table 6. Unit Testing of Lane Violation module 69](#_Toc66392859)

[Table 7. Unit Testing of Lane Violation Logic module 72](#_Toc66392860)

[Table 8. Unit Testing of Lane processing module 73](#_Toc66392861)

[Table 9. Unit Testing of Vehicles module 75](#_Toc66392862)

[Table 10. Unit Testing of GUI Lane module 76](#_Toc66392863)

[Table 11. Unit Testing of Region of Interest module 77](#_Toc66392864)

[Table 12. System Testing 79](#_Toc66392865)

# List of Figures

[Figure 1. Flowchart for Traffic Violation Detection with Computer Vision 15](#_Toc66392866)

[Figure 2. Flowchart of Traffic Signal Violation Detection System using Computer Vision [2020] 16](#_Toc66392867)

[Figure 3. Flowchart of RLVDS 18](#_Toc66392868)

[Figure 4. Flowchart of A Video based traffic violation detection system 20](#_Toc66392869)

[Figure 5. Flowchart of High-Level Traffic-Violation Detection for Embedded Traffic Analysis 21](#_Toc66392870)

[Figure 6. Flowchart of Machine Vision for Traffic Violation Detection System through Genetic Algorithm 23](#_Toc66392871)

[Figure 7. Flowchart of Automatic Real Time System for Traffic Security and Violation Detection 25](#_Toc66392872)

[Figure 8. Flowchart of Traffic Violation Detection Demo 27](#_Toc66392873)

[Figure 9. Use Case Diagram 38](#_Toc66392874)

[Figure 10. Data Flow Diagram 39](#_Toc66392875)

[Figure 11. Activity Diagram of Parking Violation](../../../../../C:/Users/im_re/Documents/Custom%20Office%20Templates/tvds_final_report.dotx" \l "_Toc66392876) 41

[Figure 12. Activity Diagram of Lane Violation 42](#_Toc66392877)

[Figure 13. Domain Model Diagram](../../../../../C:/Users/im_re/Documents/Custom%20Office%20Templates/tvds_final_report.dotx" \l "_Toc66392878) 43

[Figure 14. Combined Sequential Diagram of Lane and Parking Violation 44](#_Toc66392879)

[Figure 15. Combined Collaboration Diagram for Parking and Lane Violation](../../../../../C:/Users/im_re/Documents/Custom%20Office%20Templates/tvds_final_report.dotx" \l "_Toc66392880) 46

[Figure 16. Class Diagram](../../../../../C:/Users/im_re/Documents/Custom%20Office%20Templates/tvds_final_report.dotx" \l "_Toc66392881) 48

[Figure 17. Block Diagram of Parking Violation 56](#_Toc66392882)

[Figure 18. Block Diagram of Lane Violation and Retrogress Detection 61](#_Toc66392883)

[Figure 19. YOLO operation (*Source:www.researchgate.net/publication/340883401\_YOLOv4\_Optimal\_Speed\_and\_Accuracy\_of\_Object\_Detection)* 62](#_Toc66392884)

# List of Abbreviations

BDM - Background Difference Method

CCS - Code Composer Studio

CCTV - Closed Circuit Television

CNN - Convolutional Neural Network

COCO - Common Objects in Context

CPU - Central Processing Unit

DDA - Digital Differential Analyzer

FPGA - Field Programmable Gate Array

GB - Giga Bytes

GPU - Graphics Processing Unit

GSM - The Global System for Mobile Communications

GTK - GNU Tool Kit

GUI - Graphical User Interface

HD - High Definition

HDD - Hard Disk Drive

IDE - Integrated Development Environment

IEEE - Institute of Electrical and Electronics Engineers

IRV - Infrared Vision

ITMS - Integrated Traffic Management Systems

LAN - Local Area Network

LBP - Local Binary Patterns

LSD - Least Significant Digit First

MATLAB - Matrix Laboratory

MB - Million Bytes

MOSSE - Minimum Output Sum of Squared Error

MP - Million Pixels

MS - Microsoft

OpenCV - Open Source Computer Vision Library

PC - Personal Computer

PTZ - Pan Tilt Zoom

RAM - Random Access Memory

RFID - Radio Frequency Identification

RGB - Red Green Blue

RLVDS - Red Light Violation Detection System

ROI - Region of Interest

RTA - Road Traffic Accidents

SADP - Search Active Devices Protocol

SORT - Simple Online Real Time Tracking

SPP - Spatial Pyramid Pooling

SRS - Software Requirements

SSD - Solid State Drive

TSO - Thin Structure Occlusions

TVDS - Traffic Violation Detection System

TVM - Traffic Violation Management

UI - User Interface

YOLO - You Look Only Once

# Introduction

## Background

Traffic on roads means all the vehicles and pedestrians. Traffic rules are those rules which govern traffic and regulate vehicles. Traffic is formally organized in many jurisdictions, with marked lanes, junctions, intersections, traffic signals or signs. As the number of vehicles on the road increased, need for the traffic rules was realized due to recurring accidents which could be prevented by following simple rules. In 1865, the first traffic rule was introduced in Britain. Later as the number of vehicles grew, rules were added to increase safety and make traffic efficient. In Kathmandu valley though the number of vehicles grew the road, technology used are still the same. Many people do not know all the rules and many of those who know do not follow the rules. To prevent accidents many traffic rules are enforced. These rules must be followed by all the motorists. [1]

Traffic violation occurs when a driver fails to follow the traffic rules that regulates vehicle operation on the road. If any motorists fail to follow the traffic rules, a traffic ticket is issued to the driver. The main duty of the traffic police is to make motorists and two-wheel drivers follow the rules and if they do not follow the rules traffic tickets are issued. A traffic ticket is a notice issued by a law enforcement official to a driver, indicating that the user has violated traffic laws. Traffic tickets generally come in two forms: a moving violation such as exceeding the speed limit or non-moving violation such as a parking violation with the ticket also being known as a parking ticket.

In Nepal, any person caught violating traffic rules is slapped a fine between Rs. 500 to Rs. 1,500. As many as 378,927 rule violators were booked in the fiscal 2017-18. Out of them 5,060 motorists and two-wheeler riders were caught with the help of CCTV (Closed Circuit Television) cameras installed at more than 200 locations in the valley. Less than 2% of violators were caught using the CCTV cameras. [25]

Violation of lane discipline is driving recklessly on the road without using turn signals and not following the lane discipline. Violation of lane discipline tops the traffic offence chart with 33,377 violations though many violators of lane discipline are not fined.

Speeding is excessive speed (driving above the speed limit) or inappropriate speed (driving too fast for the prevailing conditions) recognized as a major contributory factor in both the number and severity of traffic crashes. Very few speeding tickets are issued in Kathmandu valley as speed monitoring is only enforced randomly on very few occasions with limited resources. But still, the problem of speeding is conventionally manageable by traffic speed sensors.

A road is divided into several sections by Lane lines or dividers. Dividers basically solve the problem of lane discipline and lane retrogress but, in the cases, where only lane lines are present, drivers intentionally cross the lines as well as drive in opposite lanes without any hesitations or ignorance.

Parking in spots where parking is prohibited is one of the major traffic violations that are encountered in Nepal. The reckless parking results in large traffic jams in crowded areas and creates unwanted disturbances for shops as well as road users who travel in that road. Parking violation is controlled in Nepal by a person in site who monitors continually for offending vehicles which is not as effective as it should be. Thus, due to this reason, traffic offenders freely park in no parking and flee away without the punishments they deserve.

Some of the traffic rules violation that our system can detect are:

* Violation of lane discipline
* Parking Offence
* Lane Retrogress

Our program is the software that uses real time object detection to detect the violation committed by the driver. Object detection is the task of detecting objects in video framewith fast inference while maintaining a base level of accuracy. Video footage from the CCTV is used as the data input for the system. OpenCV can be used for real time object detection with the help of frameworks like You Only Look Once (YOLO) as well as state of art trackers can be used to track the activity of vehicles on the road. [2]

Speed of the YOLO framework is 30 frames per second on high end GPUs. YOLO is pretty accurate in object detection and its classification.

Trackers like MOSSE and DEEP SORT which are capable of tracking single or multiple objects have found a strong base in modern applications of computer vision as well as offense detection.

## Evolution of traffic detection system

Traffic violation has been one of the major problems since the development of roads and vehicles. On average 3,287 people are killed every day in road accidents [3]. Different remedies have been made through the period. Due to advancement of technologies and increase of population the problem doesn’t seem to be declining. In cities, where the number of vehicles continuously increases faster than the available traffic infrastructure to support them, congestion is a difficult issue to deal with and it becomes even worse in case of car accidents.

### Latest status of traffic violation detection system

The invention of vehicles dates many years ago but the first car brought to Nepal was in 1958 BS. Traffic control system was formulated in 2007 BS. The first traffic light was implemented in 2023 BS in Kathmandu. Traffic lights weren’t that useful back then since there weren’t many vehicles or we could say we were way ahead of our time. In today’s scenario, there are not sufficient traffic lights in Kathmandu. The latest development in the traffic control system is the use of RFID systems. RFID systems overcome the drawbacks of problems related to image processing. Radio-Frequency Identification (RFID) is the use of radio waves to read and capture information stored on a tag attached to an object. Although traffic violation has been one of the major problems all over the world, very few countries have taken the step towards a smart traffic system. For example, Dubai has implemented around 15,000 cameras which is constantly monitored by Traffic police officers but detection is still done manually by officers themselves. The people who violate the rules are fined digitally especially in Dubai. It is very rare to find these new detection systems. Speed detection systems have been implemented in most of the developed countries. Even in Nepal, every now and then traffic officers manually try to implement speed detection. We have CCTV control rooms for traffic management. [4]

### Benefits

There are over 1.025 billion vehicles today in the world [5]. If an appropriate system is implemented to manage these vehicles, road accidents will be drastically reduced. It will help all the traffic officers and volunteers who work extremely hard. These following points will help us to show some benefits of the traffic management system.

1. Improving traffic safety

Non appropriate parking, inappropriate changing of lanes, heavy traffic can lead to road accidents; our program will help with all of these. Our system can detect the vehicle who violated the rule. It improves the traffic safety.

1. Reduce in infrastructure damage

Road accidents not only lead to damage of life but also destroy our road structures. It also hampers the decoration of the road. If we implement our system then we can reduce the expenditure on road repair, allowing it to be allocated somewhere. Lane and retrogress violation causes road destruction, which cause infastructure damage.

1. Traffic control

We can clearly distinguish that today’s traffic system will not be able to have a huge mass of vehicles. Mostly people try to escape from the situation and traffic officers couldn’t care less. It’s like catching fish in the sea. There are so many of them that if we catch like thousands then other hundreds still run away. So, a program is needed in order to keep everyone inside the rule, not letting even one to escape.

1. Improved journey times

Most of the people violate traffic rules so that they can reach the destination as fast as possible but unknowingly they are creating a mess and disturbing everybody on the road. Our system will make the road traffic condition better and everybody will reach their destination in time.

1. Prevents road accidents

People claim accidents are unfortunate events but most of the accidents are due to failure of people's discipline on the road. There are many causes behind the accidents. Lack of experience, overconfidence, overcrowded roads, are the cause of accidents.

1. Serviceable help for traffic volunteers

In the context of developing countries where the government doesn’t invest the required amount of money for road development, traffic controllers play a huge role. We can see their hard work but it seems insufficient and it sometimes turns violent. Increment in vehicles every day make their traffic controlling task burdensome so let the system take all the stress and traffic officers and use their time someplace else.

## Existing System

We have studied the following systems and observed the respective strengths & weaknesses of the system.

### A video-based traffic violation detection system

This is a proposal of an improved background-updating algorithm by use of wavelet transform on dynamic background, and track moving vehicles by feature-based tracking method. It is realized in C++ with the help of Open-CV. It proposes Background Difference Method (BDM) & feature-based tracking for the detection of moving vehicles.

Strength(s):

1. BDM proposed in this paper is computationally fast.
2. This paper realizes intelligent traffic management.
3. More fast and accurate detection and tracking.

Weakness(s):

1. The main weakness of BDM is it needs to update the background image in real time when the environment changes.
2. The video module proposed in this system has less frame rate compared to the modern video systems. [6]

### Traffic Violation Detection System based on RFID

This article is published in an International Journal. This article attempts to introduce an intelligent control system based on RFID technology. By the help of RFID technology, vehicles are connected to computerized systems and intelligent light poles. In this project, an intelligent control system is capable of tracking all vehicles, crisis management and control, traffic guidance and recording driving offences along the highway.

The methodology employed for achieving the detection deals with the introduction of intelligent highways with RFID scanners attached to light poles referred to as “intelligent light poles”. An RFID tag is attached to the vehicle while entering the highway, the light poles gather different disseminating information such as traffic guidance and warning. The information thus disseminated is then sent to the central computer. The offences are then recorded while the driver drives along the highway. After the highway is completed, then the tag is removed off the vehicle.

Strength(s):

1. All of the data that are being sent helps to locate the driver well.
2. The poles can also be used for other purposes than just placing RFID scanners.
3. The method is fast; it has less delay in comparison to other methods.
4. Can detect multiple offences.

Weakness(s):

1. The methodology proposed by this article can be difficult & time consuming.
2. Materials like metal and liquid can impact RFID signals.
3. Works well on wide highways only. [7]

### Traffic Rules Violation Detection with Computer Vision

This project is made for the third year second semester System Development (CSE-3200) course of Khulna University, Bangladesh. This system can detect most common three types of traffic violation in real-time which are signal violation, parking violation and wrong direction violation. A user-friendly graphical interface is associated with the system to make it simple for the user to operate the system, monitor traffic and take action against the violations of traffic rules.

In the methodology, first, the CCTV footage from the roadside is sent to the system. Vehicles are detected from the footage. Tracking the activity of the vehicle system determines if there is any violation or not. Different types of violations have different algorithms to determine the violation. Gray scaling and blurring, Background Subtraction, Binary Threshold and Dilation and find the contour are used in the system. Open-CV computer vision library is used in Python for image processing purposes. For implementing the vehicle classifier with, Tensor flow machine learning framework is used. SQLite database of Python is used.

Strength(s):

1. User Friendly Graphical User Interface.
2. Low RAM Usage.
3. Can run on almost any device.
4. Can be extended further.

Weakness(s):

1. Only a limited no. of violation is only detected.
2. Fails when the violation pattern changes.[8]

### Intersection Violation Detection by Hikvision

Hikvision is a provider of innovative security products and solutions. Intersection Violation Detection is also one of the systems provided by them.

Hikvision manufactures cameras equipped with software specialized for violation breaches in the intersection. It provides two camera choices: 3 MP iDS-TCE300-A6 & 9 MP iDS-TCE900-A.

Strength(s):

1. This system allows high video compression.
2. It can support multiple application modes: e.g.: external input, checkpoint vehicle detector, RS-485 radar, mixed-traffic lane and video analysis E-police.
3. It uses wire over the LAN, so is more reliable.
4. Can be activated through both web browser and SADP application.
5. Rich interface and advanced detection system

Weakness(s):

1. This system can be complex for usage by some users.
2. The system can crash sometimes when a lot of the violations take place at the same instance. [9]

## Customer’s perspective

Our customer whom we interviewed was the Traffic Head from Traffic Police, Satdobato. On the basis of existing systems mentioned above, the customer had an overall positive perspective towards the system. He highlighted following points:

1. There is no any prevailing system and there have been no attempts to computerize the traffic rule violation.
2. When a misunderstanding takes place in between drivers (referred to both motorists and motorcyclists), the current system of showing CCTV footage proof is both tedious and time consuming.
3. He mentioned how the traffic violence was increasing at an exponential pace and was going out of control with only a manual existing system.
4. He highlighted the need for a computerized system in traffic violence control as manual detection and fine to individual breaches are close to impossible.
5. He pointed out the lack of enough manpower in the field of traffic control as a whole, most of the manpower is centralized towards traffic control rather than traffic violation monitoring.
6. He also pointed out the increasing ignorance of drivers due to incompetence in implementation of rules & regulations well.
7. Less than 2% of the all of the violations are captured through CCTV, thus, the system is necessary. [10]

## Observation of Existing System

Table 1. Table of Comparisons

| **S.N.** | **Title** | **Method** | **Features** |
| --- | --- | --- | --- |
|  | A video-based traffic violation detection system | Background Difference Method | Red light violation, vehicle tracking, speeding, vehicle retrogress, saving & displaying information |
|  | Traffic Violation Detection System based on RFID | RFID technology | Parking in no parking, stop, accident detection, wrong lane detection, speeding, car breakdown probability |
|  | Traffic Rules Violation Detection with Computer Vision | Gray scaling and blurring, Background Subtraction, Binary Threshold and Dilation and find the contour | Signal violation (red light violation) [by drawing line], parking violation [pre-figured rectangle, direction violation [use of lines for regions] |
|  | Intersection Violation Detection by Hikvision | Not specified | Unsafe driving behaviors, speeding, red light violation, Lane violation, overload (motorcycle), parking violation, wrong Lane driving, incident detection, no helmet, targeted vehicle detection, object detection |

### Common Features

The common Features between all the systems are:

1. Red light violation
2. Speeding
3. Retrogress violation
4. Surveillance Camera

## Problem Statement

According to the Metro Traffic Division, there is an average of 40-60 accidents per day [11]. The one of the major causes of this is violation of prevailing traffic rules. In the last 10 years, 22,461 lost their lives in Road Traffic Accidents (RTA), according to Nepal Police. Police records blame 95% of accidents in Nepal is due to negligence of drivers. On a daily basis, 1,500 traffic police officers work to manage the traffic in Kathmandu Valley and in the last five months they penalized 185,436 individuals for violating traffic rules — 1,236 per day. Similarly, 472,407 vehicle drivers and owners were fined in the last fiscal year. A whopping 3 million people have been penalized for violating various traffic rules in the last 5 years and this has resulted in the government earning more than Rs1.238 billion.[12] All of these reports show drivers negligence towards following traffic rules is one of the major reasons for these accidents. Manual efforts are basically not enough to control all of these, as well as implant a cognizance in the minds of drivers. The major violations include lane violation, illegal parking & speeding.

The scope of this problem is every road user. It can be clearly seen that all of the people will face problems until the problem mentioned above is patched. Pedestrians walking on the road, crossing the road are constantly in threat of being a victim of traffic violation and rash driving. Pedestrians can get hit by speeding vehicles. Similarly, Cyclists and Motorcyclists are also posed to the threat. Large speeding vehicles when comes out of control, or the motorcyclists and cyclists lie on the black spot, then they also can get collided with each other resulting in a fatal accident. Also, cars, buses, and truck drivers can collide with each other in lane violations. Red Light Violation also causes accidents. Children won’t be able to walk freely, even on footpaths. Old aged, blind are also the ones who get most affected by accidents due to traffic violations. Traffic Police that constantly manage the flow of vehicles also put their lives constantly in danger due to traffic violations, as said by Mr. Rajesh Silwal, Head Supervisor, Traffic Police Division, Satdobato.[10]

Traffic violation problem is a genuine problem serving to impede the growth of a country, it’s traffic set of rules & regulations and a serious safety issue. Traffic Violation related issues are growing day by day and so are the fatalities related to it. Living in a constant threat in his/her day to day life is not what someone would want & look forward to. Not only does this pose a threat to road users, but a traffic violation accident or stopping for a penalty causes a traffic jam around, not if, traffic jam, it affects the smooth flow of the traffic as well as may cause more accidents to happen. “In your body; a biological system, all your cells must obey the rules too, like where to be, what to do, when to multiply. Cells which can be free of the rule telling them where to be our Lymphocytes (white blood cells / police cells)”.[13] There is another reason that such behavior is so prevalent. There is no correlation between the amount of the fine and the magnitude of the crime, fines are ridiculously low. The ability to prove guilt is limited due to shortcomings in the law: it's much more profitable to break the law. Case and point: The Taiwanese factory, which has destroyed the environment and wildlife, that bypassed all environmental controls and promises; thereby making profits in the millions of dollars while the penalties are inconsequential. [14] This is not a static problem, as traffic violations increase, so does the negligence of the driver to follow the rules, psychology states. As the number of vehicles start to grow, and traffic congestion lumps up even more, traffic violation has started to become an even more and more serious problem.

The system tries to solve the major problems of Parking offences and Lane Crossing Violation with the help of surveillance camera. We have implemented YOLO as a watch guard for detecting vehicles in the scene and used object tracker to determine whether it is commencing some parking violations or not using a video.

For lane violations as well as retrogress, we use YOLO with deep SORT tracker in order to track multiple objects in scene and check whether each vehicle is moving into their right lane as well as not crossing any lanes.

It is most useful to traffic police officers. They are able to catch offenders more effectively than manual ways of catching. Both on duty officers and officers staying in the headquarters can get the offense report, and can make required law implementation. It is also beneficial for students willing to learn more in this field, as code written by a student will be understood by another student easily. Besides this, it is beneficial to the nation as a whole as it is able to implement its law easily. This law enforcement will probably change the driver’s psychology and drivers will follow rules & regulations thus making a road a safer place to travel on. Drivers will also be going to have a proper privilege of driving on the roads as with other drivers following rules properly.

## Aims & Objectives

### Aims

Our aim that we have planned in to achieve is given below:

* Parking Violation for Single Vehicle, Lane and Retrogress Violation Detection Using Object Detection and Tracker

### Objectives

Objectives of our system that will be there to achieve the above aim are listed below:

* Detection of vehicles on the road
* Lane crossing violation detection
* Detection of Lane Lines
* Separation of Lane Areas
* Detection of Parking offence
* Vehicle retrogress violation detection
* Capture a picture of the violating vehicle or footage
* Create a violation alert on violation

## Significance of Study

Our research was important considering lack of similar researches in Nepal and non-existence of such studies or systems. It made the requirement of the system clear. From the problem statement above, it can be seen that it is quite of a problem. The study can contribute to people who further wants to develop in this field. We were able to develop the bounding box approach and clarify about it’s importance in proper detections.Still the proper work needs to be done in violation detection field and importance of use of smart techniques in bounding box based violation detection is clarified here within.

## Project Outline

This project discusses on development of a parking, retrogress and lane violation detection application based on Object Detection and Tracking.

Parking Violation is detected by the use of the object detector which detects the vehicle on the frame and a tracker, which tracks it to find it’s location. After a vehicle remains in the parking spot for more than a designated period of time, then it’s regarded as a violation. Simply, the location of bounding box is obtained from tracker by applying detector and tracker in each frame.

Similar to Parking Violation, lane as well as retrogress violation focuses on detection of lane cross as well as opposite direction driving violation. This violation detection is for multiple objects. A multi-object tracker supplies with us bounding box co-ordinates of vehicles. With the bounding box co-ordinates in couple of frames, we can detect the violation. We also analyze the road for middle lanes with the help of a line detector. Thus, we receive lane line co-ordinates and on the basis of the lane line co-ordinates, we separate the lane areas.

On the basis of vehicle co-ordinates and lane line co-ordinates, we are able to detect lane violations. We can detect when there is a cross between the two which is basically the lane cross violation part.

Similarly, on the basis of vehicle being on opposite lane and driving through that lane is detected by calculating area of the bounding box of the vehicles. There is a particular characteristic that in a designated lane, either the area of the bounding box increases or decreases, if in a lane area, a suspicious increase or decrease is being taken place in a lane area then it is regarded as violation.

# Literature Review

Following the many papers, there are many approaches to realization of traffic violation detection systems. But, in all of the papers, the common primary aspect is the detection of the object, classification of that object & the nature of violation it commits. Object detection can be achieved using different algorithms like Background Difference Algorithm, Dynamic Background Difference with Wavelet transform (improved background difference) algorithm, Edge Detection with Hough Transform, & numerous other methods. The classification of that object is made by different architectures like Darknet, mobile net etc.

Released in late 2018, Traffic Rules Violation Detection with Computer Vision is a project that has been implemented on the basis of different algorithms. First the CCTV camera footage is sent to the system. Vehicles are detected from the footage. Tracking the activity of the vehicle system determines if there is any violation or not. Different types of violations have different algorithms to determine the violation. Image processing technique is used for object detection. Specifically, Gray scaling & blurring is used on the CCTV footage, then this input is fed to the Background Subtraction section where desired object’s area is evaluated subtracting the current frame from the reference frame using Background Subtraction Method. For the refinement of the background, to remove all the holes and noises from the frame and get the desired object area accurately, the binary threshold operation is performed by the Binarization Method. The threshold image is dilated to fill the holes and the contour is found from the image. Drawing rectangle boxes over the contours desired moving objects are taken. OpenCV computer vision library is used in Python for image processing purpose. The classification of the moving objects is done into three classes – car, motorbike & non-vehicle. Classification is moreover done with the use of mobile net v1 neural network architecture constituted in TensorFlow machine learning framework. Transfer learning approach is used to train the model with their dataset; the dataset consists of 500 images per class. Lines & areas are drawn for different types of violations. For example: When the traffic light is red and a car is crossing the straight line, a picture of that car is registered in the database along with some environmental values.[8]

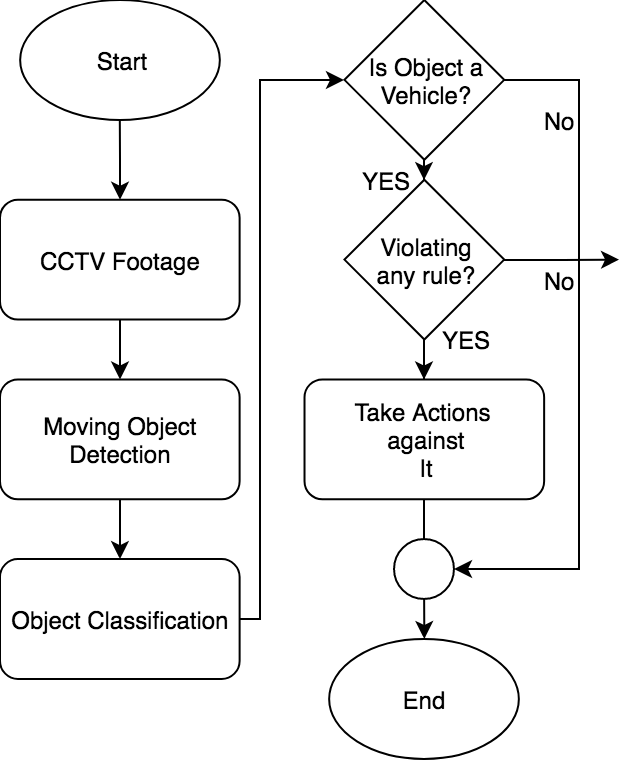


Figure 1. Flowchart for Traffic Violation Detection with Computer Vision

In 2020, from the same university, by a different group of students, Traffic Signal Violation Detection System using Computer Vision has been developed. Input is the same. The vehicles are detected using the YOLOv3 model/algorithm, which is a third object detection algorithm in YOLO (You Look Only One) family. Classifier model is built with Darknet-53 neural network architecture also included in the newer version of TensorFlow. Image processing & Object detection as previously is implemented with the help of YOLO which is a constituent algorithm in OpenCV. A region of interest can be defined by the administrator for the violation. After that the administrator will need to select two points to draw a line that specifies the traffic signal. This system only supports signal violations, and not parking as well as wrong lane violations. An output footage showing violation is also generated in the background. [15]

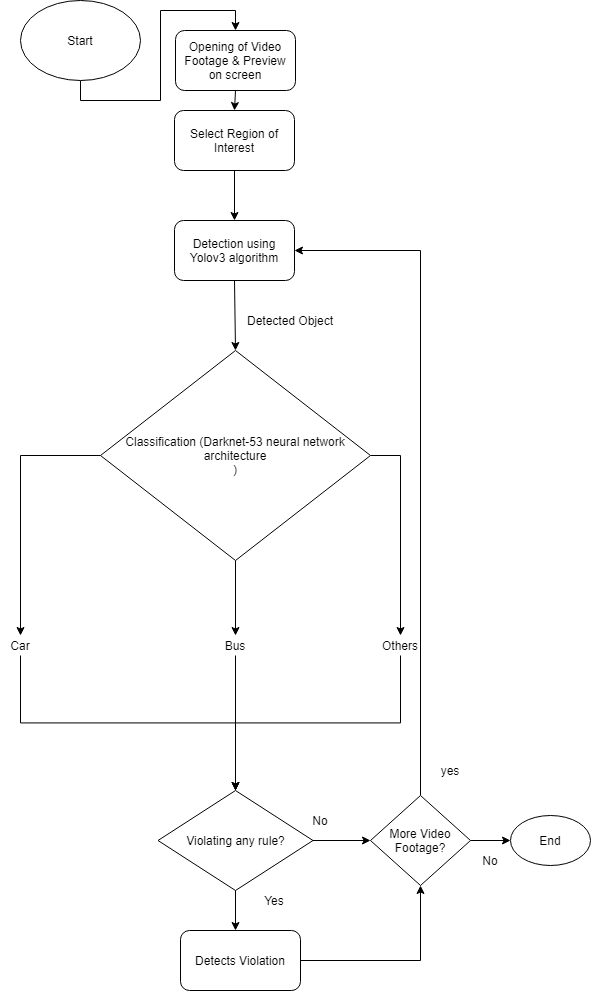


Figure 2. Flowchart of Traffic Signal Violation Detection System using Computer Vision [2020]

In the completed development of Red-Light Violation Detection System (RLVDS) for Indian vehicles under Integrated Traffic Management Systems (ITMS), there are three cameras equipped for detecting the red-light violation detection, two are fixed & one is a PTZ (pan-tilt-zoom) camera. The scheme for detection of red-light violating vehicles is based on background subtraction technique where the intrusion of any other object into the scene may be obtained by subtracting the non-background image from the background image. 5 successive images’ average difference with the background image generates an average background image & after it, background subtraction is performed on the image obtained by polling cameras available on the site every 3 seconds. The requirement for it arises due to the fact that to detect an object from an image captured at night, if the image is subtracted from the background image of morning then obviously wrong prediction of object will result. Two fixed cameras generate two types of fixed images whose pan angles are corrected internally during processing. For PTZ cameras, it captures images in 6 different angles, each angle has a corrected value which is used for correction of angle of all the angles. Mean gray value is evaluated and if the difference image results in a high mean gray value then it is considered as a non-background image, i.e. there is a possibility of intrusion of an object in the image. So, in the difference image the white stop-line will also become black if it is not occluded by the vehicle. For detection of this occlusion five lines have been hypothetically drawn on the stop-line parallel to the longer edges of the stop-line, with a vertical gap of three pixels between them. The technique successfully identifies vehicle images in 92% cases. Major reason behind erroneous tracking of vehicle images is occlusion on the stop-line by pedestrians, speeding vehicles over the stop-line, erratic changes in outdoor lighting conditions and possible vibrations of the surveillance cameras.[16]

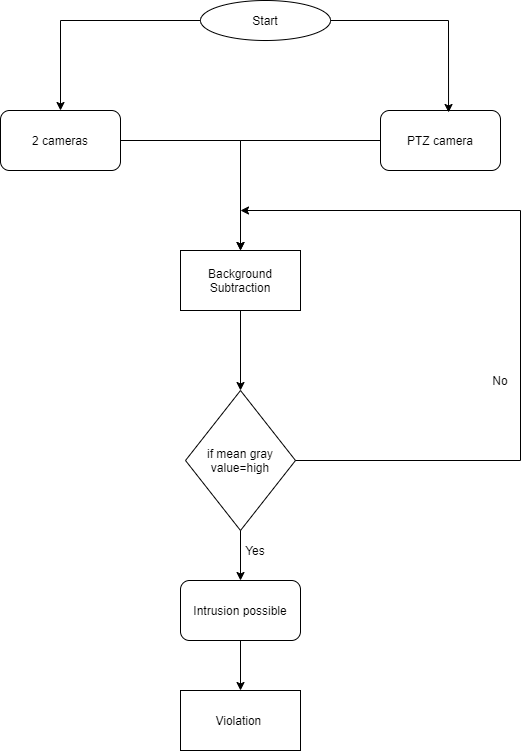


Figure 3. Flowchart of RLVDS

A video-based traffic violation detection system proposes 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. This paper proposes video streams as the input, compares five of the prevalent methods in detection of a vehicle in the video stream. Background Difference Method, which is already described above; subtracting the current image from the background image, binarization which means to take a grayscale image and conversion of the same to black and white image. This method needs to update the background image in real time when the environment changes. Inter-frame Difference Method, which subtracts the current frame from the previous frame, and then changing area can be found by setting the threshold value. Edge Detection Method, applying edge detection and denoising on the input image for result image. The result image is compared with a template image, after matching. This method is not suitable for real-time image processing. Optical Flow Method, detects moving objects by the change in the time domain of pixel intensity of an image sequence, and the relationship between the structure of objects and movement. This Method is, however, computationally intensive and is susceptible to noise. Block Matching Method, splits an image into M×N macro blocks. A motion vector is evaluated by searching for optimal matching of a macro block of the current frame in the next frame. A moving vehicle is composed of many macro blocks.

The paper discusses about the use of dynamical background update based on the wavelet transform, that is combined with background difference and feature-based tracking for detection of moving vehicle. A weighting parameter α is used to adjust the speed of background update within the current image. This method has the merit of avoiding the interference of the foreground, when updating the background. This is required because if the background updates too fast, there are extra holes in the foreground image otherwise, the movement affects detection performance. 2-D discrete wavelet transform is performed that eliminates trajectory and solves the problem. The advantage of OpenCV is realized in this by use of various image processing functions such as Gaussian filtering function GaussianBlur(), image space mapping function cvtColor()etc. The results are tested for different values of α (0.01, 0.03, 0.05 and 0.1) and it performs best on the α=0.03. Feature based tracking tracks then vehicle by the center, and decide whether they violate or not by thresholding. For every violation, the threshold is compared to the threshold. For e.g.: if Speed < Threshold, it is speeding. [6]

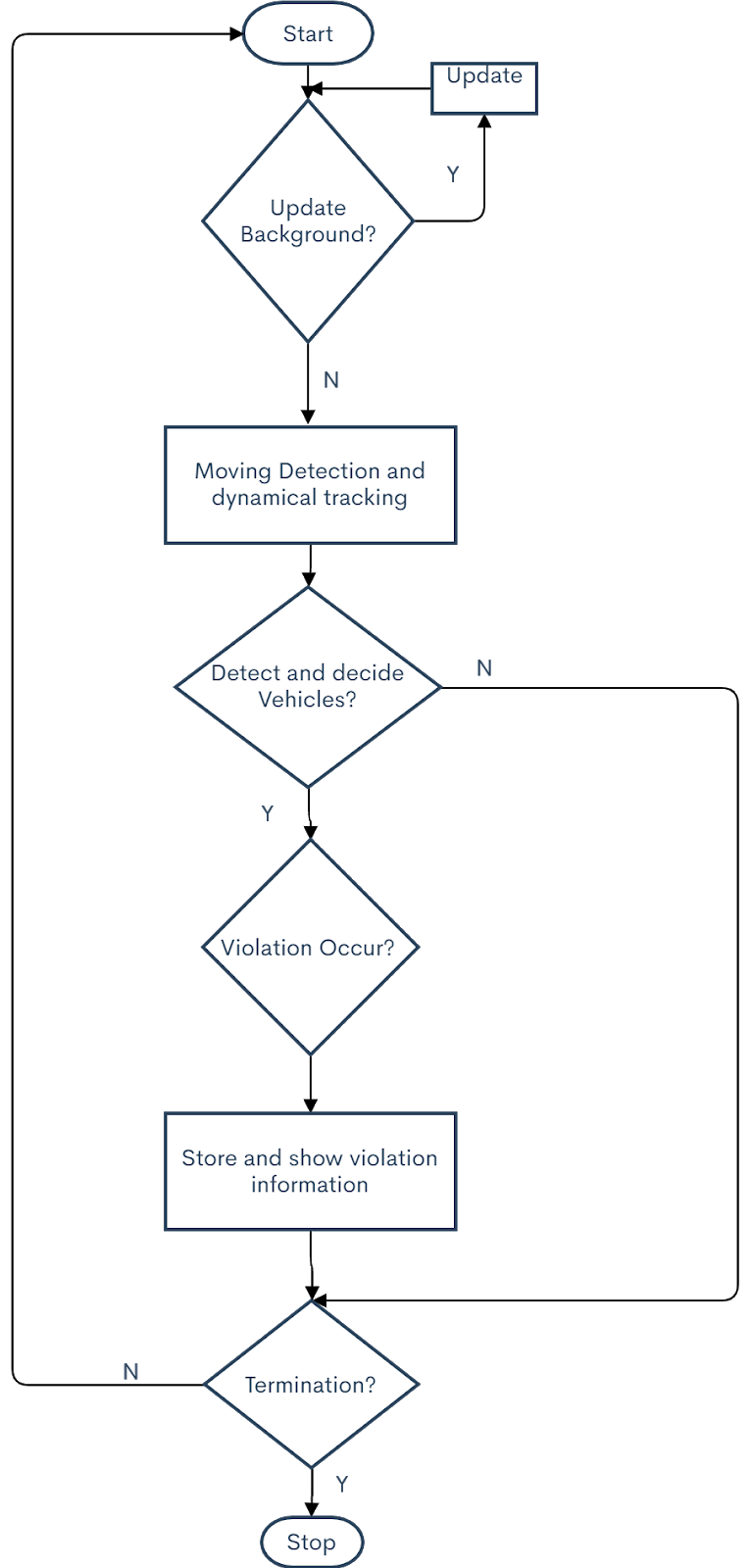


Figure 4. Flowchart of A Video based traffic violation detection system

High-Level Traffic-Violation Detection for Embedded Traffic Analysis on the other hand uses background segmentation and a novel road model for obtaining candidate traffic participants. A region-based tracking system equipped with static occlusion-reasoning, tracks the positions of the objects in the scene. Experiments show that they have achieved the processing rate of 63-150 Hz, with high average correct road detection, object-type classification rates of 93-94% and event detection accuracy of 85%.This paper emphasizes a road model used with the background difference algorithm. Also, a series of consistency tests are performed in-order to prevent false foreground regions i.e., false vehicle objects in the scene. Earlier proposed systems couldn’t prevent vehicles that enter the scene and stop for a relatively long period of time from identifying as a default background object. They don’t know whether the visible object in the background is correct. A Road Detection Algorithm for estimation of models for pixels in the background that belong to the traffic roads is used, vehicles dominantly moving on the road, forms the training set and the pixels thus are classified as a road, forming a road model. Previously, static objects occlusions were just implicitly handled using heuristics. For e.g.: a tree that is occluding the background is just implicitly by simple formulae. The system presents an approach for handling it by obtaining the regions in the background that frequently occlude. For Thin-Structure Occlusions (TSO) by a thin object such as a thin light pole, sign pole, by splitting the vehicle, e.g. when the driver moves behind, the vehicle gets split into two parts. A vehicle is enclosed by a bounding box, one side (back) of the tracked object remains static, while the opposite side moves in the direction of the object motion. These two observations are combined to obtain occluded regions. A naïve Bayesian classifier is used for the classification of objects. Different other 5 semantic elements such as vehicle or object length, width, object’s maximum speed, etc. The authors added the possibility of enhancement by inclusion of shadow handler for splitting more falsely merged objects. [17]

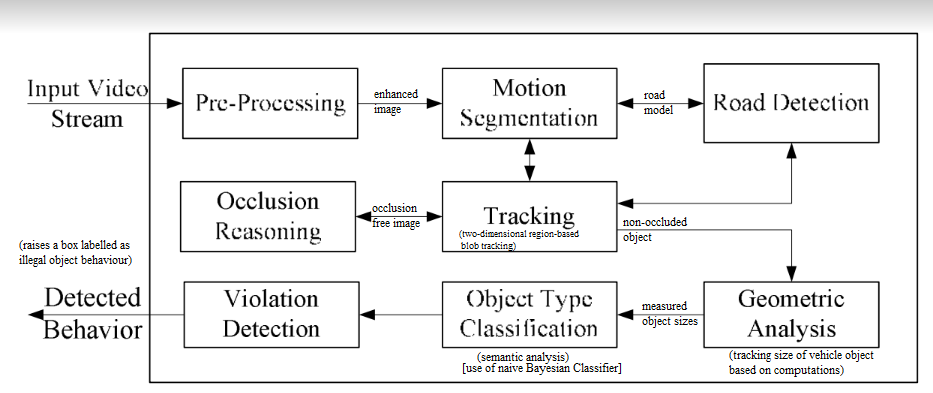


Figure 5. Flowchart of High-Level Traffic-Violation Detection for Embedded Traffic Analysis

Machine Vision for Traffic Violation Detection System through Genetic Algorithm, this paper has presented a machine vision algorithm to detect traffic violations specifically swerving and blocking the pedestrian lane. The proposed solution consists of a background difference method, and focuses on the genetic algorithm of the system to detect these violations. The general process is that a capture picture is to be subtracted first by the reference image, then the genetic algorithm is run to find the violator, and finally a display is outputted by cropping the image with the corresponding type of violation. The genetic algorithm chooses the best value from the initially random population and converges to a single solution. The algorithm is well-suited for real time implementation in traffic detection systems. The system inputs were captured photos from a CCTV camera and the outputs were cropped pictures of the car that was detected to have such violations mentioned earlier. At first the input image is subtracted from the reference image and a genetic algorithm is performed to give output. It is implemented through MATLAB in a computer with a clock processor of 3.4GHz and an 8GB of RAM, to have a nearly optimal program runtime. The system has detected both the swerving and pedestrian lane blocking. The system was faster for detecting swerving than pedestrian lane blocking detection. The system can only process one data at a time and runtime is also slow. The system “Traffic Signal Violation Detection System using Computer Vision” was developed by Abu Noman Md. Sakib, Pias Roy students at Department of Computer Science and Engineering of Khulna University of Engineering & Technology in Bangladesh. This system has used YOLOv3 and tkinter to detect traffic signal violations. The system uses computer vision, GUI with python library Tkinter and basic OpenCV. The main idea of the project is to detect and track the traffic signal violators. The System consists of two main components: vehicle detection model and graphical user interface (GUI). When the video footage is selected, the moving objects are detected from the input footage using YOLOv3 object detection model to classify vehicles into respective classes. OpenCV and machine learning software library which is used in this project for image processing purposes. It improved the accuracy. Tracking the activity of vehicles, the system determines if there is any violation or not. The GUI makes the system interactive for the user to use. Users can monitor the traffic footage and get the alert of violation with the detected bounding box of the vehicle. [18] A traffic line is drawn over the road in the preview of the given video footage by the user. The line specifies that the traffic light is red. Violation happens if any vehicle crosses the traffic line in red state. The detected objects have a green bounding box. If any vehicle passes the traffic light in red state, violation happens. After detecting a violation, the bounding box around the vehicle becomes red. This system can only detect the signal violation using computer vision. It uses OpenCV on python and machine learning to detect the object, classify and detect the violation but does not issue the fine or detect speeding vehicles. [19]

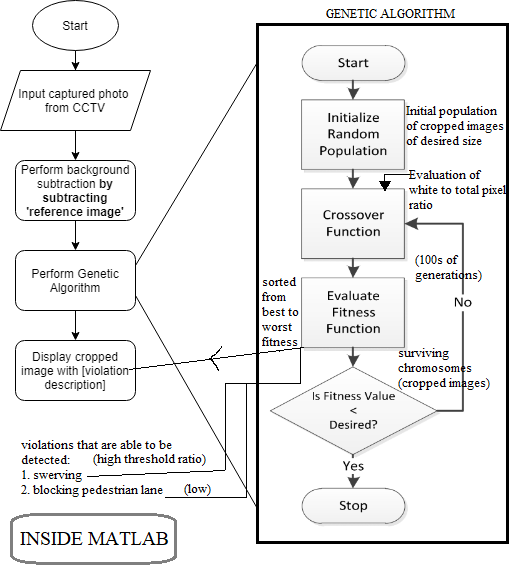


Figure 6. Flowchart of Machine Vision for Traffic Violation Detection System through Genetic Algorithm

Traffic Violation Detection System based on RFID features an RFID system. This system contains antennas, readers, tags and software & is specialized for detecting violations in the sections of highways. The genetic algorithm of artificial intelligence or feedback is used. Light poles equipped with RFID readers are placed on the for violations. RFID tags are attached to highways on entrance and they are taken off while they exit from the highway. By the use of genetic algorithms, offences including No-parking in the parking zone, vehicle retrogress, speeding, and also creates exceptional cases for emergency vehicles. While moving along the highway, the vehicle constantly reports itself through using communicative signals between the tags and the present readers in the intelligent light poles.[7]

Automatic Real Time System for Traffic Security and Violation Detection, uses video processing technique. This paper helps to overcome all the problems related to over speeding, illegal lane changes and red-light violation. Notification of violation is issued to both the traffic control department and traffic violator with proper evidence. A high-resolution camera is used to capture the video continuously. If violations are detected, then the number plate of the vehicle is captured using a camera & it is processed to obtain the number. Candidate Plate regions are extracted using a pre-processing function, accuracy increases while decrease of computational time. A tree of Local Binary Pattern (LBP) based cascade classifiers is used for classifications of plate regions into one; LBP is a type of visual descriptor used for classification in computer vision. A violator is considered when the base profile of the blob intersects the solid line considerably; a high-resolution image is used to grab a snapshot of the violating vehicle. For lane detection, a background model is used to segment the foreground from the background of the scene. In this violation region is a solid line region on the road and region of interest specifies the area of the image that will be processed for violations. Background model is created initially using a single frame and is then updated for every new image frame acquired from the camera. The Gaussian mixture model is used for every pixel in the image. For shadow detection it is performed per pixel for the non-background pixels using the rationale that shadows have similar chromaticity but lower brightness than the background model. The foreground image extracted from the background subtraction module is further processed for shadow pixels using the shadow detection algorithm. RGB vector distances are used for identifying shadow pixels.

Parts of the base profile of a blob resulting from stripping out the part that intersects with the violation area, a blob is considered to be involved in illegal solid line crossing. For red light violations, a similar stop line technique is used, which is already discussed earlier. Background difference method is used for calculating Threshold, which is in turn used for detection of the violation. In this paper’s approach, speeding is detected simply using two sensors separated at a distance and speed is calculated on the basis of time taken by the vehicle to cover that distance. The classification is LBP but extended one by another paper which proposes Multi-Scale Block Local Binary Patterns MB-LBP is used for license plate recognition and classification. The computation is done based on average values of block sub regions, instead of individual pixels. MB-LBP is more robust than LBP. On experimentation, it was found that there were few false detections and the system was able to achieve an average accuracy of 85%.[20]

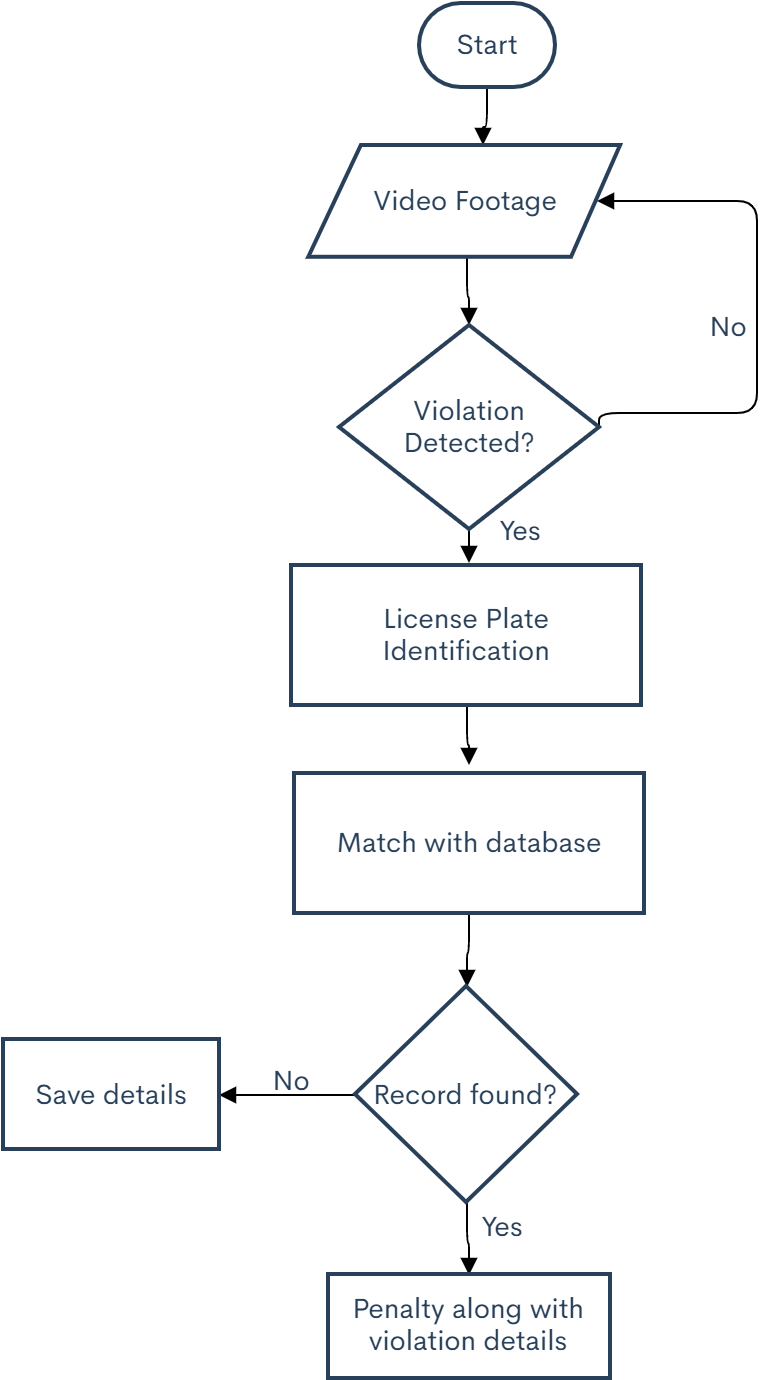


Figure 7. Flowchart of Automatic Real Time System for Traffic Security and Violation Detection

Traffic Violation Detection Demo, is a video analysis software capable of detecting some types of traffic violations by analyzing video streams from traffic cameras. The software is able to detect and store traffic violations that are related to intersections of vehicles with solid lines and traffic light violations. The project was developed using Python 3, extensive use of SciPy, OpenCV and UI is based on GTK. First, models of the road and of vehicle movements are obtained by processing the video stream. Second, these models are combined and analyzed to detect traffic violations.

To analyze the structure of the road a manual approach is taken. In this project, straight road marking lines are analyzed. Solid straight lines play a key role in structuring the road. Such lines are stop lines, edge lines, double lines and others. First, the background image of the video is extracted by applying a background difference algorithm. Second, the background image is analyzed to detect straight lines of the road by an edge detector algorithm and reduction of noise by the approach of Hough Transform. Implementations of these algorithms are available in OpenCV library. But the problem was other sections of the image, which contain a lot of edges, were also identified as lines. So, with further preprocessing of the image, e.g. with the removal of small non-straight edges, a better result could be achieved, but another approach is chosen. LSD algorithm was used for the detection of lines. In order to identify whether several segments are on the same line, a graph is composed out of detected segments, which is performed by using an algorithm for finding the strongly connected components of the graph whose implementation is available in SciPy. To detect the vehicles, YOLO neural network processes frames of video stream. To identify the vehicles and to smoothen fluctuations of predicted positions in a sequence of detections, SORT algorithm is applied to the detections. This algorithm uses rudimentary combination of techniques such as the Kalman Filter and Hungarian algorithm for tracking components. After detection of the lines, they are shown in the window of the application and the user can set any line type (Front, Parallel) or leave it without setting a type. Front lines are intended to be used across the carriageway, and parallel ones along the carriageway. Crossing of the lines is considered to be a violation.[21]

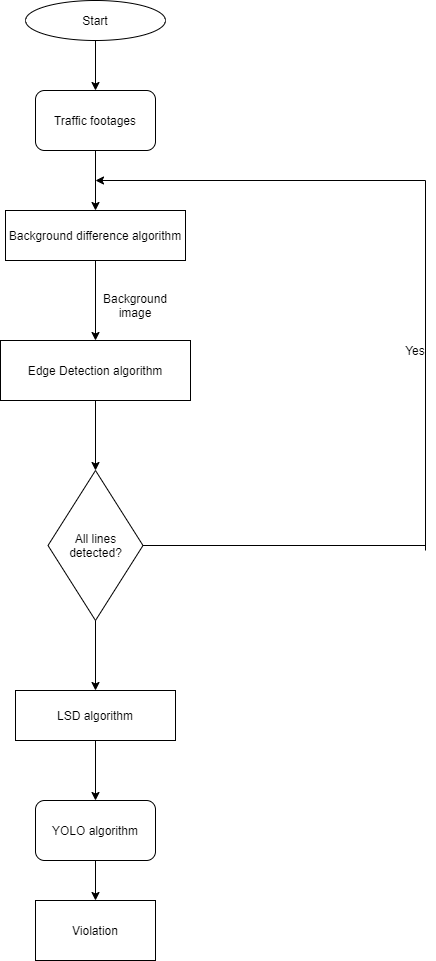


Figure 8. Flowchart of Traffic Violation Detection Demo

Traffic Management and Violation Detection Systems: An Open Challenge, this is a compilation of different TVDS systems and their comparison. This paper emphasizes the use of an IRV technology using cameras along with the RFID technology, database server and GSM technology to reduce violation. This also includes a lot of Traffic Violation Detection Systems, where some above discussed systems have also been mentioned. This paper discusses about sacat which is a Traffic Sign Detection and Recognition and Traffic Violation Management (TVM).[22]

Traffic Signal Violation Detection using Artificial Intelligence and Deep Learning, uses computer vision implemented using YOLOv3 object detection for traffic violation detection which included signal jump and speeding. On the basis of location of the vehicle during the frames and region of interest, they obtain an accuracy of 97.67% for vehicle count detection and an accuracy of 89.24% for speed violation detection.

The proposed architecture featured a surveillance system with intelligent detection and tracking of multiple vehicles from the surveillance input video using YOLOv3 as the object detection algorithm. This is done through a neural network and a classification architecture. Next, from the same given video footage, traffic lights, zebra crossing, different lanes, and traffic signs are classified this comes under environment awareness. Their system operation speed is dependent on the density of the traffic. Future scope for parallel computing was also discussed which seems to solve the problem of high computational time for high traffic volume roads. Use of GPUs and high-end FPGA kits. [23]

Another similar system was Computer Vision Based Approach for Traffic Violation Detection. CCTV footage was also the primary input in this system. The methodological pipeline was similar in the previous papers which included several common steps; vehicle (object) detection, vehicle identification, tracking and traffic violation detection.

They selected Haar Cascade detection as object detection technique due to its performance and accuracy. They first of all have trained a Haar cascade classifier using a set of positive and set of negative images. Positive images implied the vehicular images while the negative images implied the non-vehicular images. The results were returned as bounding boxes. They have used KalmanBoxTracker in order to track the data objects in the frame. Few cases of partial and complete occlusions are handled with the help of the tracker itself. The violation detection comprised of identification using the angles of the edges that connect the points of the vehicle’s path. The angles between two consecutive previous positions of the vehicles are used in purpose of disobeying of lane laws. If the angle exceeds a certain value, then the system detects it as violation. The overall system was able to identify 2 out of 3 lane rule violations committed by the vehicles it tracked. Due to the pipelining methodology, accuracy in one stage affects the accuracy of another phase. [24]

# Methodology

## Frontend

Our frontend is backed upon by vue.js. Different types of requests are created to the server. Vue.js is what allows the website to be dynamic in nature and website needing minimum reload.

## Backend

### Core Backend

Our backend is backed upon by python’s Django framework. We used Django rest framework to create the API which returns various types of data. We haven’t focused much on security of the API. API is configured to return various types of data upon requirement specified by the backend.

For example;

# fetch all products

if data[‘required’] == ‘all\_products’:

return Product.objects.all()

# fetch featured products

if data[‘required’] == ‘featured\_products’:

return FeaturedProduct.objects.all()

Above example is a replica code snippet of products section of API. Upon changing required attribute of the data, the API is able to either return all product or only featured products.

Furthermore, models were used to store various types of data such as products, customers, orders etc which was fetched using Django functions.

### Filter, Search and Sort Implementation

For filter, search and sort implementation, the filtering data such as price range, subcategory etc. or searching query was sent from the frontend and according to needed attribute of the data similar to previous was used to specify where we wanted to perform either filter, sort or search or hybrid operation.

The filtering and searching were implemented with django’s builtin in Objects.filter() was used to filter out based on price range or subcategory or search term sent from frontend.

Similarly, sort also depended upon either sorting by price or name, ascending or descending data received from the frontend. Furthermore, here – operator was added inorder to sort the objects in descending order.

We currently support four recommended products.

### Recommendation

The recommendation part of the website has two major scenarios.

#### Scenario 1: **F**or a guest

For a guest, the trending items are shown.

#### Scenario **2**: When a guest has logged in and now became a ‘user’ and has more than 2 views

When there is more than views in two consecutive subcategories then from the two most viewed subcategories, the two products are taken from the most viewed one and the other two products are taken from the second most viewed subcategory.

Figure 9. The Scenarios of Not Enough Ratings

**Scenario 3: When a user has added one subcategory product to the wishlist:**

In this case, two products are shown from the most viewed subcategory and two from the subcategory of which the product is added to the wishlist.

**Scenario 4: When a user has added two subcategories product to the wishlist:**

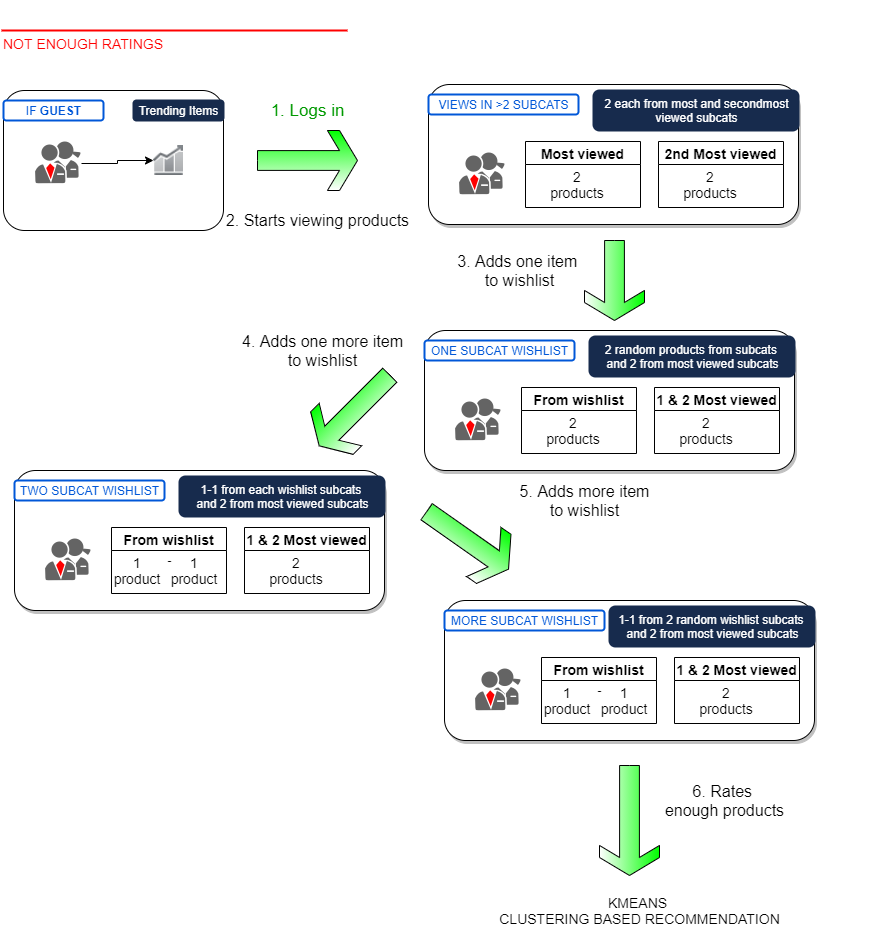
In this case, two products are shown from the most viewed subcategory and two are shown from one each of each subcategory of which the product is added to the wishlist.

**Scenario 5: When a user has added more than two subcategories product to the wishlist:**

In this case, two products are shown from the most viewed subcategory and two are shown from one each of two random subcategories selected from ‘n’ subcategories of which the product is added to the wishlist.

**Scenario 6: When a user has rated just enough products for k-means based recommendation:**

In this case, the kmeans clustering is applied and according to the products that are rated by the user, the users are kept into different clusters. In a cluster, the products that are rated by one user and not rated by user, but rated by the users in the same cluster are shown to the users.

The scenario is described in the following figure:

# Discussions

Our project was able to detect about 6-7 violations of about 12 videos which is an accuracy of 50-60% in case of lane and retrogress violation, considering the concept of bounding box based violation detection is new, the accuracy seems to be reasonable. While in Parking module, the program seems to work fine if it goes in sequence and there is only one vehicle or only one vehicle arrives in our designated parking spot.

Other methods have also performed lane violation system but not this bounding box approach we have tried. This method has been preferred over other methods because other methods such as using angle to detect the violation is impractical in case of a lot of traffic and incorrect camera position despite it’s 90%+ accuracy. Other methods with 80% to 90% accuracy using methods such as keeping a track left by it’s track also is impractical in case of our country where the neither the mode of traffic is box packed like other countries and the road condition gets hectic in some periods of day.

The reasons behind our results being low accuracy is current unpredictability of the YOLO object detection. This affects both parking and lane violation modules. For Parking, it’s not of a big issue as detection once is enough for the program to work. But, still sometimes the vehicle may go undetected. The huge affect occurs in the lane violation module where detection is required almost in every frame for the tracker to work properly. The position of bounding box almost changes in every frame due to how YOLO works. This causes sometimes false detections to occur or even real violations to go undetected.

Another issue is in the tracker, the tracker used in this is DEEP SORT Tracker. Tracking is a complex field of computer vision because of varying properties of the objects in a frame. The specific issue in our program is the tracker recognizing the same object and identifying it as a distinct one in other frames. Due to this multiple instances for the same violation may occur. However, a potential solution to this could be saving of the violator or the objects being tracked temporarily and using some kind of mechanism to find the percentage of similarity between the two.

Next weakness of the program subsides in Hough Transform. Hough Transform is also as unpredictable. It gives different outputs for even the same set of images. The line detections by Hough Transform depends on how the region of interest is selected. For the same image, the detection of the lines depends upon in which way the region of interest is selected. Not properly detecting the lines can also be a problem in a real world application. Also, the weather conditions and visibility of a day may also be of a issue for Hough Transform.

The lane violation detection is only possible where lane lines are at least visible. The program when started, the hough transform is just run once. What may happen is the lanes of the road of the same place may go faint as time passes and the program goes on being less efficient.

For program to work on even pan tilt zoom or camera position changing due to external factors or camera placed at another position needs an another approach for at least lane line detection and separation. For program to work in night conditions, night vision cameras and YOLOv3 model trained with night pictures need to be also added. The YOLO model needs to be updated from time to time because of changing of vehicle fashion in every few years.

To make this application feasible for real world, many strategies should be applied throughout the program. The bounding box approach is pretty feasible in case of proper detection and tracking of the image. Furthermore, a powerful system with high end GPUs and/or dedicated FPGAs may also be required for high volume traffic conditions. With computers and GPU getting powerful, vehicle positions for even every vehicle when the camera frame is full of vehicles may also be achieved. With the trackers being advanced, the approach seems to be feasible. Even more complex lane violation logics can be incorporated and existing could be modified in order to make the system more accurate and really working.

# Conclusions

Though, this bounding box approach we followed produces good results and many cases are detected, the method is still behind application due to unpredictability of detector, tracker and lane line detector, though all of them being decent to the present day. It lies into hands of us or any other interested candidates who can solve the flaws in the project in order to generate even more accurate and reliable output. Few cases of violation completely go undetected due to lack of bounding box datas. As well as, many detections of the same violator is shown due to incompetence of tracker in recognizing the same vehicle. Still many of the violation are detected. If some further improvements can be engulfed into the project, then, the bounding box approach is quite practical and can be used for real world purposes and detect violations. The further works may be addition and improvement of violation logic, implementation of even more powerful detectors and more accurate trackers, 2 lane line system or basically 1 solid line logic may be incorporated, the multiple violation tracking problem may be resolved and also many more problems some of which discussed above or others may be incorporated in the future. Thus, parking and lane violation can improve further and the bounding box approach can be used more practically than our current implementation

# References

|  |  |
| --- | --- |
| [1] | "The Himalayan Times," [Online]. Available: thehimalayantimes.com. |
| [2] | "TowardsDataScience," [Online]. Available: towardsdatascience.com. |
| [3] | "PapersWithCode," [Online]. Available: paperswithcode.com. |
| [4] | "Metropolitan Traffic Division," [Online]. Available: traffic.nepalpolice.gov.np. |
| [5] | "Google," [Online]. Available: google.com. |
| [6] | B. Zhang, X. Wang, L.-M. Meng and K.-L. Du, "A video-based traffic violation detection system," Hangzhou, December 2013. |
| [7] | S. Hajeb, M. Javadi, S. M. Hashemi and P. Parvizi, "Traffic Violation Detection System based on RFID," *International Journal of Mechanical, Aerospace, Industrial, Machatronic and Manufacturing Engineering,*vol. 7, no. 2, pp. 290-293, 2013. |
| [8] | R. Zaman and S. Reza, "Traffic Rules Violation Detection with Computer Vision," Published in GitHub, Khulna, Bangladesh, 2018. |
| [9] | "Intersection Violation Detection System," Hikvision Digital Technology Co., Ltd., Hangzhou. |
| [10] | R. Silwal, Interviewee, *Traffic Violation in Nepal.*[Interview]. 02 12 2019. |
| [11] | M. T. Division, "Twitter," 18 12 2019. [Online]. Available: (https://twitter.com/valleytraffic/status/1207457875645222912). [Accessed 19 12 2019]. |
| [12] | G. Gartaula, "Nepali Times," 10 Jan 2019. [Online]. Available: https://www.nepalitimes.com/banner/nepals-deadly-roads-take-their-toll/. [Accessed 20 Dec 2019]. |
| [13] | J. Smith, *Answered on 'Why should people follow traffic rules?',*Quora, 2018. |
| [14] | R. Little, *Traffic violations are everyone's problem,*Ho Chi Minh, Vietnam: Vietnam News, 2011. |

|  |  |
| --- | --- |
| [15] | A. N. M. Sakib and P. Roy, "Traffic Signal Violation Detection System using Computer Vision," Published in GitHub, Khulna, Bangladesh, 2020. |
| [16] | S. Saha, S. Basu, M. Nasipuri and D. K. Basu, "Development of an automated Red Light Violation Detection System (RLVDS) for Indian vehicles," in *IEEE National Conference on Computing and Communication Systems (COCOSYS-09) CS11*, Kolkata, India. |
| [17] | J. A. Vijverberg, J. Han, P. H. de With, D. Cornelissen and A. N. de Koning, "High-Level Traffic-Violation Detection for Embedded Traffic Analysis," in *Acoustics, Speech, and Signal Processing. ICASSP-88., 1988 International Conference*, Eindhoven, The Netherlands, 2007 May. |
| [18] | "python awesome," [Online]. Available: http://pythonawesome.com/traffic-signal-violation-detection-system-using-computer-vision/?fbclid=IwAR326SsJL4WRUPOhzvXr6PMQly\_WTyi8d0mwgk0J41\_yoCr-Ux\_h9tOyqkQ. |
| [19] | A. Christian, R. A. Bedruz, A. P. Uy and A. R. P. Quiros, "`Machine Vision for Traffic Violation Detection System through Genetic Algorithm," in *8th IEEE International Conference Humanoid, Nanotechnology, Information Technology Communication and Control, Environment and Management (HNICEM)*, Cebu, Philippines, Jan 2016. |
| [20] | P. Raj, D. Dhormare, S. Singh, S. Nawade and P. R. U. Yawale, "Automatic Real Time System for Traffic Security and Violation Detection," *International Journal of Trend in Scientific Research and Development,*vol. 2, no. 3, pp. 116-121, Mar-Apr 2018. |
| [21] | M. Gevorgyan, "Traffic Violation Detection Demo - Medium," 13 Dec 2017. [Online]. Available: https://medium.com/@partus/traffic-violation-detection-demo-7937c14ded71. [Accessed 12 Jan 2020]. |
| [22] | G. P. Kaur, G. Prakash, S. D. Juneja and T. Kumar, "Traffic Management and Violation Detection Systems: An Open Challenge," *International Journal of Emerging Technologies in Engineering Research (IJETER),*vol. 5, no. 1, pp. 37-41, Jan 2017. |
| [23] | R. J. Franklin and Mohana, "Traffic Signal Violation Detection using Artificial Intelligence and Deep Learning," in *Fifth International Conference on Communication and Electronics Systems*, Bengaluru, Karnataka, India, 2020. |

[24] A. Peiris, E. Edirisuriya, C. Athuraliya and I. Jayasooriya, "Computer Vision Based Approach for Traffic Violation Detection," in *13th International Research Conference, General Sir John Kotelawala Defence University*, Colombo, Sri Lanka, 2020.

[25] "Traffic Police, the servers of nation," Traffic Police, 2017. [Online]. Available: https://traffic.nepalpolice.gov.np/. [Accessed 20 Dec 2020].