Vision based Traffic analysis

Aim- The goal of the project is to automate the traffic signal violation detection system and make it easy for the traffic police department to monitor the traffic and take action against the Owner of vehicles violating rules in a fast and efficient way. Detecting and tracking the vehicle and their activities accurately is the main priority of the system.

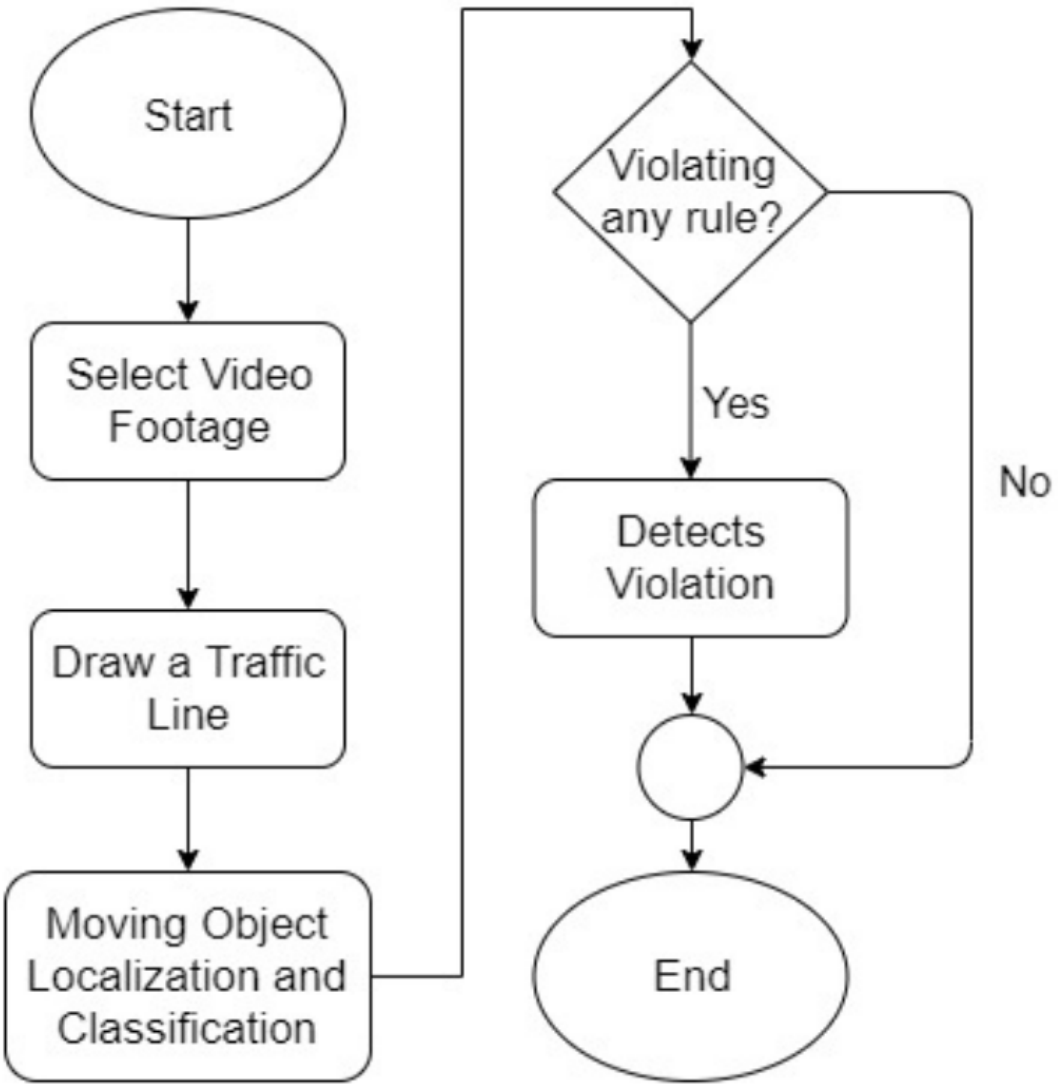
System Overview

The System consists of two main components -
• Vehicle detection model
• A graphical user interface (GUI)

Here we use the saved videos from roadside cameras.Vehicles are detected from the footage. Tracking the activity of vehicles, the system determines if there is any violation or not. System Overview shows how the system works.

The Graphical User Interface (GUI) makes the system interactive for the user to use.User can monitor the traffic footage and get the alert of violation with the detected bounding
box of vehicle. User can take further action using the GUI.

System Overview



Methodology Used

1) Vehicle Classification

From the given video footage, moving objects are detected. An object detection modelYOLOv3 is used to classify those moving objects into respective classes. YOLOv3 is the object detection algorithm in the YOLO (You Only Look Once) family. It improved the accuracy with many tricks and is more capable of detecting objects. The classifier model is built with Darknet-53 architecture.

Features

a). Bounding Box Predictions:

YOLOv3 is a single network the loss for objectiveness and classification needs to be calculated separately but from the same network. YOLOv3 predicts the objectiveness score using logistic regression where 1 means complete overlap of bounding box prior over the ground truth object. It will predict only 1 bounding box prior for one ground truth object and any error in this would incur for both classification as well as detection loss. There would also be other bounding box priors which would have objectiveness score more than the threshold but less than the best one. These errors will only incur for the detection loss and not for the classification loss.

b) Class Prediction:
YOLOv3 uses independent logistic classifiers for each class instead of a regular softmax layer. This is done to make the classification multi-label classification. Each box predicts the classes the bounding box may contain using multilabel classification.

c) Predictions across scales:
To support detection YOLOv3 predicts boxes at 3 different scales. Then features are extracted from each scale by using a method similar to that of feature pyramid networks. YOLOv3 gains the ability to better predict at varying scales using the above method. The bounding box priors generated using dimension clusters are divided into 3 scales, so that there are 3 bounding box priors per scale and thus total 9 bounding box priors.

d) Feature Extractor:
YOLOv3 uses a new network- Darknet-53. Darknet-53 has 53 convolutional layers, its deeper than YOLOv2 and it also has residuals or shortcut connections. Its powerful than Darknet -19 and more efficient than ResNet-101 or ResNet-152.

	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
	Convolutional	32	1 × 1	
1×	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
	Convolutional	64	1 × 1	
2×	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
	Convolutional	128	1 × 1	
8×	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
	Convolutional	256	1 × 1	
8×	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
	Convolutional	512	1 × 1	
4×	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

Darknet-53 architecture

2) Violation Detection

The vehicles are detected using YOLOv3 model. After detecting the vehicles, violation cases are checked. 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 violation, the bounding box around the vehicle becomes red.

Implementation

1) Computer Vision

OpenCV is an open source computer vision and machine learning software library which is used in this project for image processing purposes. Tensorflow is used for implementing the vehicle classifier with darknet -53.

2) Graphical User Interface (GUI)

The graphical user interface has all the options needed for the software. The software serves administration and other debugging purposes. We don't need to edit code for any management. For example, if we need to open any video footage, we can do it with the Open item

Requirements to run the system- GPU is required to run the system to make it faster.

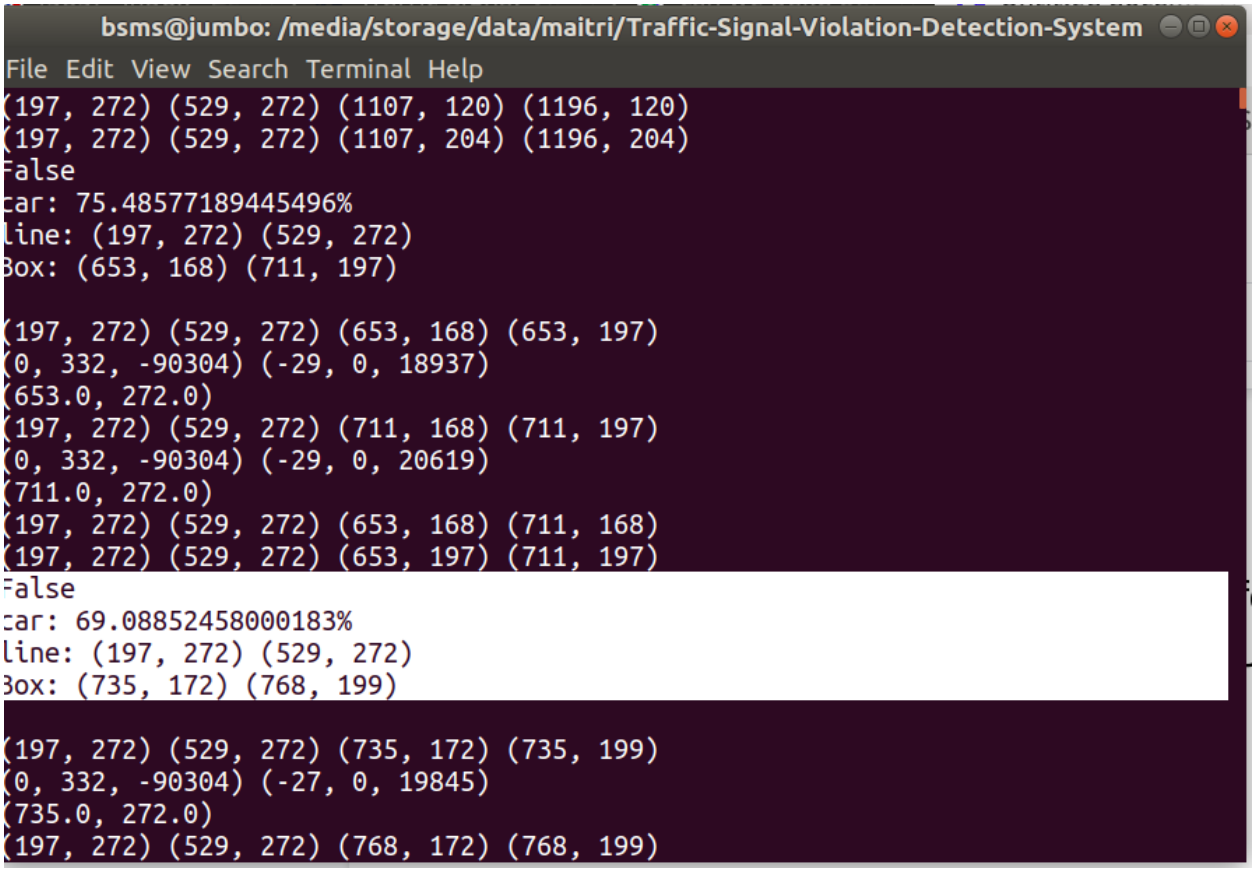
Steps to run the system

1)Primarily, for the start of the project usage, the administrator needs to open a video footage using ‘Open’ item that can be found under ‘File’

2) Select any video file from input.

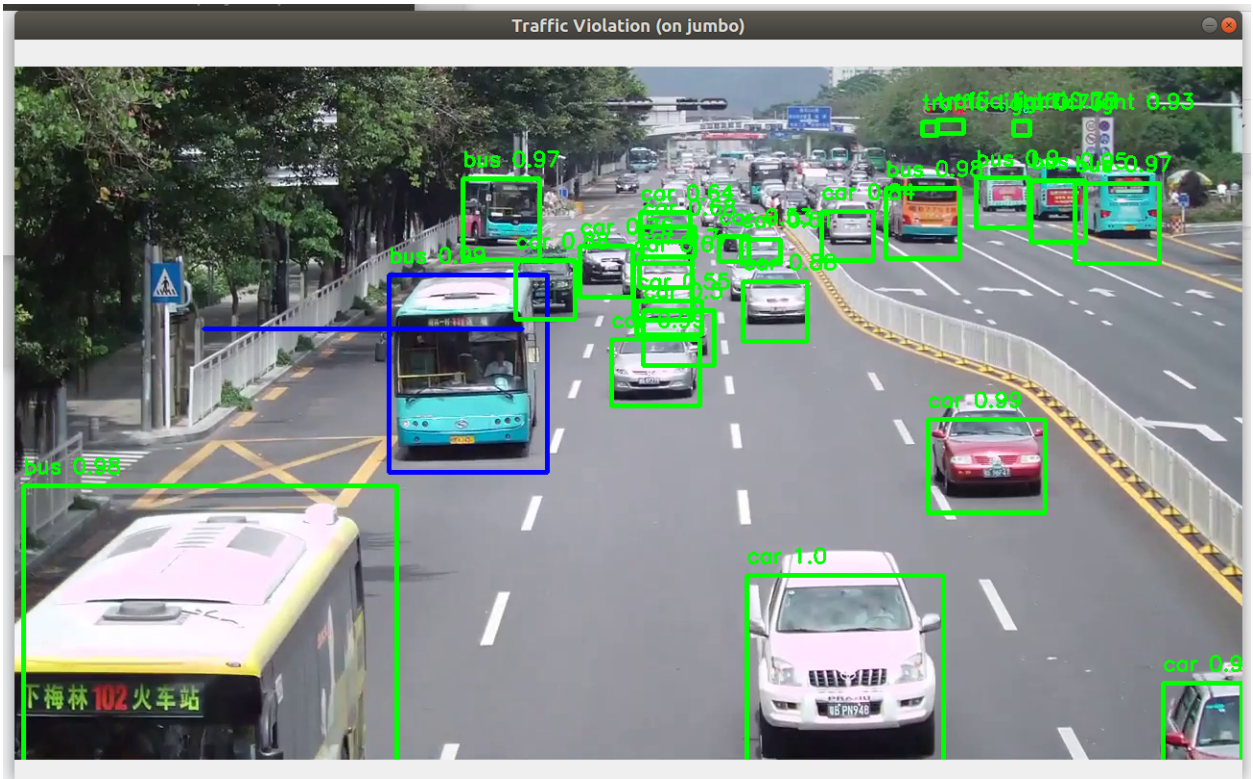
3)After opening a video footage from storage, the system will get a preview of the footage. The preview contains a frame from the given video footage. The preview is used to identify roads and draw a traffic line over the road. The traffic line drawn by the administrator will act as a traffic signal line. To enable the line drawing feature, we need to select the ‘Region of interest’ item from the ‘Analyze’ option.After that administrator will need to select two points to draw a line that specifies the traffic signal.

4)Selecting the region of interest will start the violation detection system. The coordinates of the line drawn will be shown on console. The violation detection system will start immediately after the line is drawn. At first the weights will be loaded. Then the system will detect objects and check for violations.



Coordinates of detected points will be visible on the console.

5)The output will be shown frame by frame from the GUI.



Libraries used for Graphical User Interface:
Tkinter.

Conclusion & Recommendation

The designed algorithm was effectively able to detect the type of violation specified on this project which is denying traffic signals. The convergence of detection for the traffic violation mentioned is dissimilar, since it has a different threshold condition. The system provides detection for traffic signal violation. Further, the system is able to process one data ata time. Also, the program runtime is somewhat slow, and can be improved by using a computer

with high speed processor specifications or GPU.
Future research about the application of the designed algorithm for other advanced image processing techniques. Since, this may improve the program runtime of the system by neglecting other unnecessary steps done in a background difference method. A computer vision algorithm may be done instead to provide more intelligence in the system.

Scope of improvement

To implement the number plate detection with OCR support to make this system more robust.

References

[1] G. Ou, Y. Gao and Y. Liu, "Real Time Vehicular Traffic Violation Detection In Traffic Monitoring System," in 2012 IEEE/WIC/ACM, Beijing, China , 2012.

[2] X. Wang, L.-M. Meng, B. Zhang, J. Lu and K.-L.Du, "A Video-based Traffic Violation Detection System," in MEC, Shenyang, China, 2013.

[3] Joseph Redmon and Ali Farhadi, "YOLOv3: An Incremental Improvement".

-https://github.com/anmspro/Traffic-Signal-Violation-Detection-System.