Traffic Rule Violation Detection System using ML

Amit Rangwal [1], Shakib Deshmukh [2], Hrishikesh Atole [3], Balaji Kokil[4] , Shipra Saraswat[5]

[1] *Student, Computer Engineering, Jayawantrao Sawant College of Engineering, Maharashtra, India*
[2] *Student, Computer Engineering, Jayawantrao Sawant College of Engineering, Maharashtra, India*
[3] *Student, Computer Engineering, Jayawantrao Sawant College of Engineering, Maharashtra, India*
[4] *Student, Computer Engineering, Jayawantrao Sawant College of Engineering, Maharashtra, India*
[5] *Teacher, Computer Engineering, Jayawantrao Sawant College of Engineering, Maharashtra, India*

ABSTRACT

Real time identification systems are very important and needful for safety, security rule following and socialism and also for own safety concerns. Traffic rules are important for safety as traffic laws are to prevent drivers of vehicles from causing accidents or hitting pedestrians. They are also to help control the flow of traffic so that it is more efficient. Traffic Rule Violations are leading cause of accidents, according to WHO India is a leading country in casualties occurring on road. The current system uses human interaction for rule violation detection, as it is a manual process it has some limitations, on multiple occasions we find the system gets corrupt. An alternative solution would be AI-developed System. With our system, we can detect multiple rule violations, for example, Vehicle crossing signal during red light or driving without a helmet, etc. Basic idea is to detect these violations through preinstalled cameras. We can do it by ML based algorithm where we can detect the violators by Image-Processing, getting the number pate, categorizing violation accordingly and issuing fine. Which will help increase the efficiency of traffic rule enforcement

Keywords- *Firebase GC, CNN, ML,TCP/IP,WWW,BaaS,YOLO.*

1. INTRODUCTION:

The Idea of the System we have is using the infrastructure of these high surveillance systems and integrating them with Deep Learning to identify the violations. Through this System we will eliminate the human errors and system limitations.

Real time identification systems are very important and needful for safety, security rule following and socialism and also for own safety concerns. Traffic rules are important for safety as traffic laws are to prevent drivers of vehicles from causing accidents or hitting pedestrians. They are also to help control the flow of traffic so that it is more efficient.

The severity of different kinds of punishment depends upon the nature of the offence committed with regards to breaking traffic rules citizens have to pay the fine, serve the jail term or be banned from driving any vehicle.

It detects vehicles that do not obey traffic rules, such as breaking signal, driving in the wrong direction, making illegal turns, not wearing a helmet, and other violations.

Basically, due to human errors or technical errors these violators escape and sometimes there are also chances of accidents occurring.

Traffic rules are important for safety as traffic laws are to prevent drivers of vehicles from causing accidents or hitting pedestrians. They are also to help control the flow of traffic so that it is more

efficient. The severity of different kinds of punishment depends upon the nature of the offence committed with regards to breaking traffic rules citizens have to pay the fine, serve the jail term or be banned from driving any vehicle. According to WHO report 2015, more than 1.2 deaths occur across the globe on road each year, where in India has even crossed those numbers.[1] To handle these violations, we have hard working traffic police monitoring all road activities and fining challans accordingly. All work is done manually, which leads to some limitations.

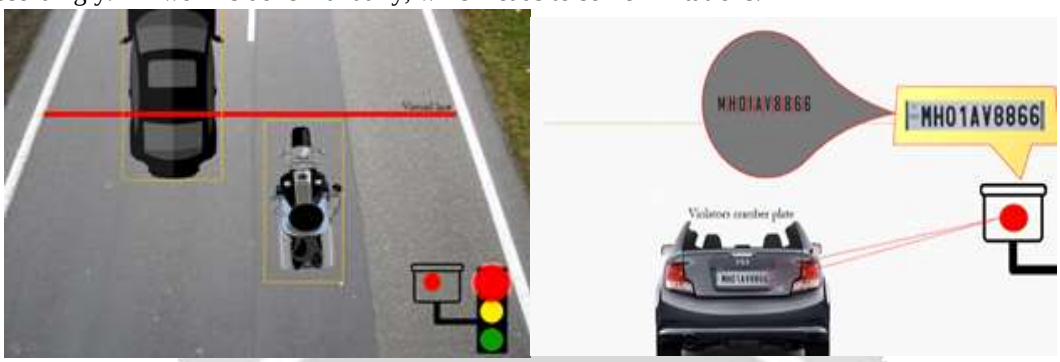


Fig.1: Violation Occurs

2. GUIDELINES FOR MANUSCRIPT PREPARATION:

2.1 Abbreviations and Acronyms:

1. Internet:

The Internet is the global system of interconnected computer networks that uses the Internet protocol suite (TCP/IP) to communicate between networks and devices. It is a network of networks that consists of private, public, academic, business, and government networks of local to global scope, linked by a broad array of electronic, wireless, and optical networking technologies. The Internet carries a vast range of information resources and services, such as the inter-linked hypertext documents and applications of the World Wide Web (WWW), electronic mail, telephony, and file sharing.[3]

Fig.1: Internet

2.Google Cloud (Firebase):

Firebase is a Backend-as-a-Service — BaaS — that started as a YC11 startup and grew up into a next-generation app-development platform on Google Cloud Platform.

Firebase frees developers to focus crafting fantastic user experiences. You don't need to manage servers. You don't need to write APIs. Firebase is your server, your API and your datastore, all written so generically that you can modify it to suit most needs. Yeah, you'll occasionally need to use other bits of the Google Cloud for your advanced applications. Firebase can't be everything to everybody. But it gets pretty close.[4]

Fig.2: Cloud Storage

3. MATH:

Three main algorithms are being used, first algorithm is edge detection in an image is to find object boundaries, which in turn can reduce the amount of data to be analyzed in an image, second algorithm CNN is used for object detection so as to detect the object using a software named YOLO. The last algorithm K-Nearest Neighbour is used to recognize the segmented characters which are performed on extracted license plate images. The three algorithms are mentioned namely:

1.Edge Detection:

Edge detection is an image processing technique for finding the boundaries of objects within images. It works by detecting discontinuities in brightness. Edge detection is used for image segmentation and data extraction in areas such as image processing, computer vision, and machine vision.[5]

Our edge detection method in this workshop is Canny edge detection, created by John Canny in 1986. This method uses a series of steps, some incorporating other types of edge detection. The skimage skimage.feature.canny() function performs the following steps:

1. A Gaussian blur (that is characterized by the sigma parameter, see introduction) is applied to remove noise from the image.
2. Sobel edge detection is performed on both the x and y dimensions, to find the intensity gradients of the edges in the image. Sobel edge detection computes the derivative of a curve fitting the gradient between light and dark areas in an image, and then finds the peak of the derivative, which is interpreted as the location of an edge pixel.
3. Pixels that would be highlighted, but seem too far from any edge, are removed. This is called non-maximum suppression, and the result is edge lines that are thinner than those produced by other methods.

Fig.3: Edge Detection

2.Convolutional Neural Network (CNN):

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConvNets have the ability to learn these filters/characteristics.[6]

The architecture of a ConvNet is analogous to that of the connectivity pattern of Neurons in the Human Brain and was inspired by the organization of the Visual Cortex. Individual neurons respond to stimuli only in a restricted region of the visual field known as the Receptive Field. A collection of such fields overlap to cover the entire visual area.

A ConvNet is able to successfully capture the Spatial and Temporal dependencies in an image through the application of relevant filters. The architecture performs a better fitting to the image dataset due to the reduction in the number of parameters involved and reusability of weights. In other words, the network can be trained to understand the sophistication of the image better.

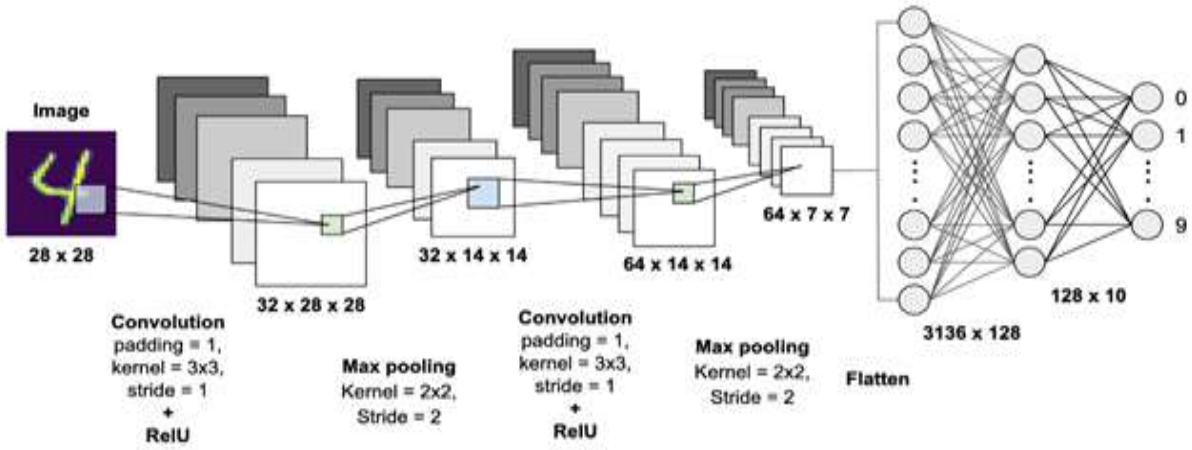


Fig.4: Convolutional Neural Network

3. K-Nearest Neighbour:

The k-nearest neighbors (KNN) algorithm is a simple, easy-to-implement supervised machine learning algorithm that can be used to solve both classification and regression problems. The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other. As we would need to in any machine learning problem, we must first find a way to represent data points as feature vectors. A feature vector is our mathematical representation of data, and since the desired characteristics of our data may not be inherently numerical.[7]

4. SOFTWARE DESIGN:

4.1 Flowchart:

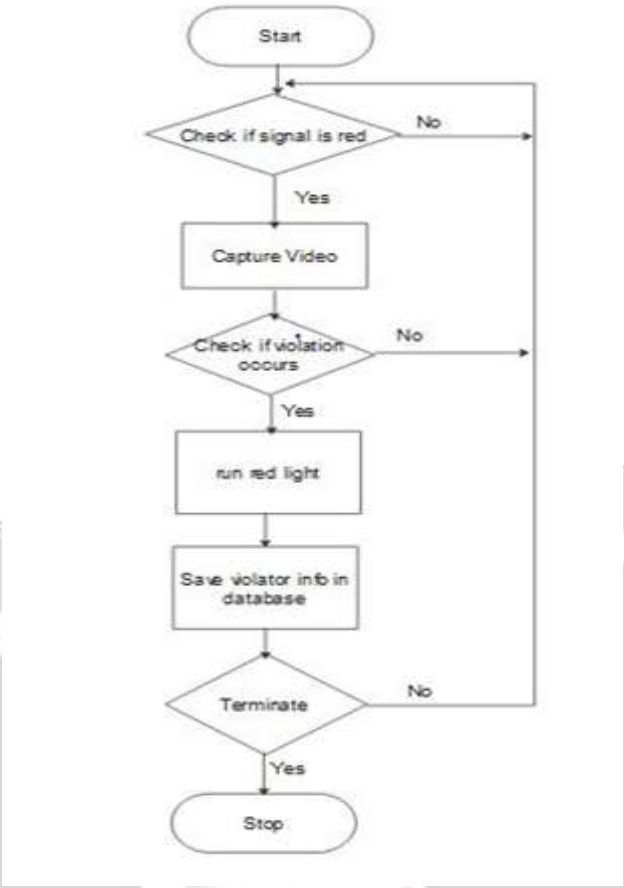


Fig.6: Flowchart

The process followed by python and yolo for the project is shown in flowchart:

I. Check the Signal: If signal is red then start capturing the video to search for any violators. If not then do not capture anything

II. Pre-processing of Vehicle Image: Convert the color (rgb) image to grayscale image. Applying Prewitt's edge detection method finds the edges of the vehicle image. Apply the morphological function "imitate" to dilute the image, so that edge lines become thinner and unwanted small lines will be washed out.

III. Check for Violation: If any violation occurs the only save those vehicle images who had done violation into the database for further processing

IV. Extract Number Plate from Vehicle Image: Once we get the plate area, it is cropped out from the vehicle image and made a separate image. This image is converted to optical image format by complimenting its black and white image.

V. Convert Character Image to Text: Using correlation function a match between extracted image and templates is found. The character in each extracted image is converted to text and stored with a variable.

4.2 Mathematical Model:

Definition: L = Flux (vehicles pass through)

q is set of vehicles

(x,t) = parallel to x axis, determining position on y axis

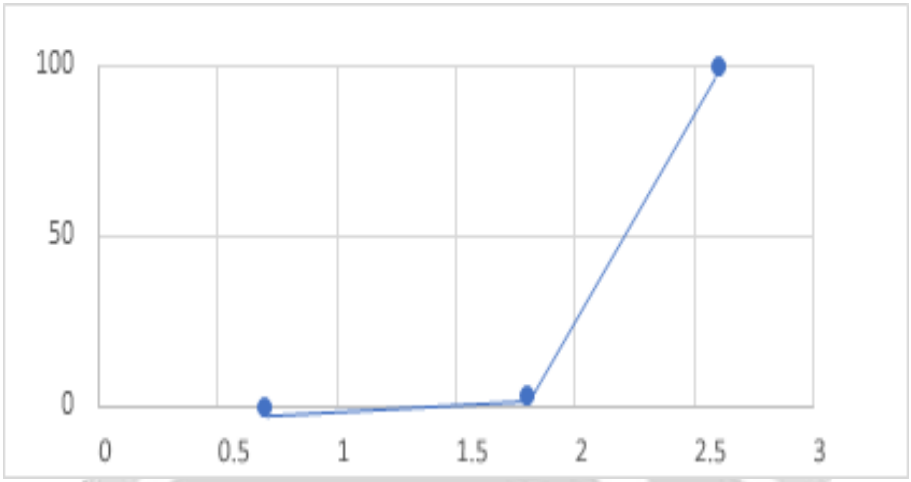
Assuming minimum violation on traffic flow

$$\int_{x1}^{x2} L(x,t)dx = \int_{x1}^{x2} \left(\frac{dL}{dt}\right) dx$$

Resultant= Total vehicle stopping at light – vehicles exit before light turns green

$$\int_{x1}^{x2} \left[\left(\frac{dL}{dt}\right) + \left(\frac{dq}{dx}\right) dx\right] = R$$

No of vehicles on Line A and B on Virtual line as X-axis



Results and finding:

1. Our system detects any traffic rule is violated or not like signal breaking.
2. If violated, vehicle is detected and number plate is extracted of the violated vehicle
3. And at last send to the admin for further process where he can view the details of the violator.

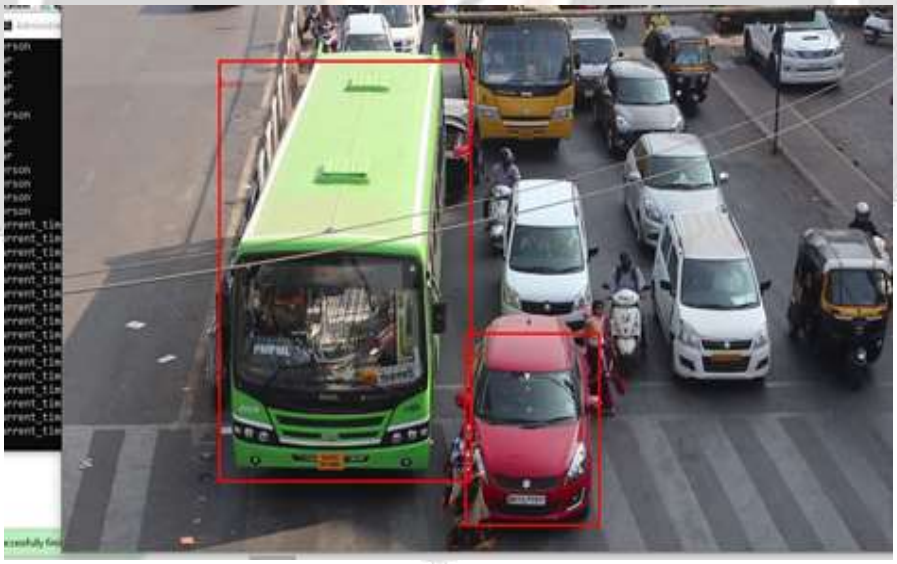


Fig. 7: Object Detection

Violator Information:

Once the process is done the image of the vehicle which violated the rule will be taken and displayed onto the app which is created using android studio.

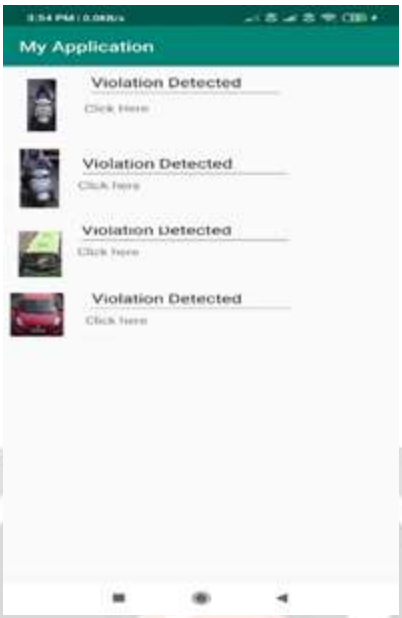


Fig.8: Violated Vehicle

5. CONCLUSIONS:

With implementation of this system the manual work will be reduced, even the human limitation will be effectively overcome. Now the window of getting away from the violators due to negligence will be narrowed. More violators can be scanned. We will be working towards an automated environment which reduces the workload on police officers too, now the workflow will become efficient. With inculcation of this new system rule will be enforced more actively and decrease the road accidents and casualties.

Thus far the system will be automated. Workflow will be efficient. Rule enforcement will decrease the road accidents and casualties.

REFERENCES:

[1] Padmaja B, Sunil M P Dynamic Traffic Rule Violation Monitoring System Using Automatic Number Plate Recognition with SMS Feedback.

[2] Wikipedia: https://en.wikipedia.org/wiki/Raspberry_Pi

[3] Techterms: https://techterms.com/definition/internet

[4] R Shreyas, Department of Electronics and Communication Engineering School of Engineering and Technology, Jain University, Bangalore, India.

[5] Adithya H B, Comparison of feature extraction techniques to recognize traffic rule violations using low processing embedded system Manish Kumar Purohit Department of Electronics & Communication Engineering Parul Institute of Engineering and Technology Vadodara, Gujarat, India mrpurohit@gsfcltd.com

[6] Pradeep Kumar B Comprehensive Traffic Management System: Real-time traffic data analysis using RFID

[7] Meghana B S, Department of Computer Science and Engineering, Ramaiah University of Applied Sciences Bengaluru, India meghanabs52@gmail.com

[8] YOLO-https://pjreddie.com/darknet/yolo/

[9] https://ieeexplore.ieee.org/document