International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249 – 8958 (Online), Volume-9 Issue-5, June 2020

Overspeed Detection using Image Processing for Indian Highways

Punith B Kotagi, Punith Kumar N, Parishwanath Madakari, Omkar V H, Vishwanath

Abstract: Over speeding of vehicles is the major cause of accidents in recent times. Monitoring of such over speeding vehicles especially on highways is of prime importance. Many manual methods are being used to detect the over speeding vehicles by traffic control team. However, these methods require lot of man power and continuous monitoring by traffic personnel. In this study, an attempt is made to develop an automatic speed monitoring system, which provides a simple way to monitoring speeds of all the vehicles from a centralized control room. This system calculates the instantaneous speed of vehicle with help of sensors and the over speeding vehicle is detected using an image processing technique using Python programming language. The developed model is validated with real world traffic data and comparative analysis of speeds obtained by manual method and developed model shows that model truly represents the field condition.

Keywords: Image processing, Speed detection, Traffic management, Traffic operation

I. INTRODUCTION

In developing nations such as India, the vehicular growth rate is increasing exponentially which is worsening the traffic operations. Most of the urban cities in India are facing traffic related problems such as congestion, accidents, pollution, etc. during peak hours. The main cause for traffic congestion in such cities is mainly due to uncontrolled urbanization and extensive usage of private vehicles. The traffic congestion leads to many problems like increase in travelling time, health disorders and accidents. Road accidents in India claimed over 1.5 lakh lives in the country in the year 2018, with over-speeding of vehicles being the major cause. Ministry of Road Transport and Highways report on Road accidents in India stated that road accidents increased by a rate of 0.46 % in the year 2018 when compared to 2017 [1] [2]. Over-speeding accounted for 66 % in total road accidents due to traffic rules violation as shown in Fig. 1.

Revised Manuscript Received on 25 June 2020.
* Correspondence Author
Punith B Kotagi*, Assistant Professor, Department of Civil Engineering, The National Institute of Engineering, Mysuru, India. Email: punithbkotagi@nie.ac.in
Punith Kumar N, Undergraduate Student, Department of Civil Engineering, The National Institute of Engineering, Mysuru, India. Email: punithkumar2798@gmail.com
Parishwanath Madakari, Undergraduate Student, Department of Civil Engineering, The National Institute of Engineering, Mysuru, India. Email: parishwanathmadakari98@gmail.com
Omkar V H Undergraduate, Student, Department of Civil Engineering, The National Institute of Engineering, Mysuru, India. Email: omkarvh1998@gmail.com
Vishwanath, Undergraduate Student, Department of Civil Engineering, The National Institute of Engineering, Mysuru, India. Email: vishwaharish70@gmail.com

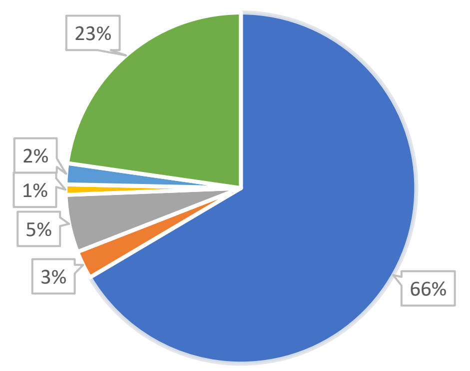

Fig. 1. Road accidents by type of traffic rules violation in the year 2018

In spite of well-equipped traffic personnel with night vision speed guns it is a tedious task to identify over speeding vehicles. Indian Government increased the over speed fine by amending of the Motor Vehicles Act. Although over speeding is the main cause for accidents in India. In order to mitigate over speeding, continuous monitoring of highways is essential. But in densely populated nations such as India, the vehicular growth is at a faster rate and monitoring speeds of all the vehicles is a complex task for traffic authorities. The manual method of monitoring speeds (such as speed guns) requires lot of man power and continuous patrolling by traffic personnel. Hence an attempt is made in this study to develop an automatic speed detecting system by using Python Programming Language. A prototype model is developed to detect the speed and type of over speeding vehicle and the developed model is thoroughly validated using real world traffic data.

II. LITERATURE REVIEW

In this section several literatures were reviewed to study the current methodologies adopted to detect speed and vehicle type for mixed traffic conditions. C. Mallikarjuna et al. developed a system to automatically analyse traffic videos and extract macroscopic traffic data such as classified vehicle flows, average vehicle speeds and average occupancy [3].

Ranga H P et al. proposed an algorithm on the basis of image processing using dimensions of vehicles. The algorithm consists of various techniques such as image differencing, thresholding, edge detection and binary morphological process by using MATLAB [4]. Chandrashekhar. M et al. placed a camera alongside of the traffic light to capture image sequences. These images were processed to control the state change of the traffic light in order to decrease the traffic congestion and to avoid the wastage of time in a green light during empty road [5]. Khanke and Kulkarni studied traffic flow monitoring based on computer vision techniques installed in the SQL server using MATLAB. Images are scanned to detect the presence of an object ad information of the registered object can be tracked [6]. Omkar et al. analysed heavy traffic congestion using the MATLAB in order to prevent causalities. A new approach was adopted to count the number of vehicles present on the road at a given particular time with the help of area of the vehicles [7]. Uke and Thool proposed a system to convert video into frames and extract references background and perform detection of moving objects using image processing through python programming [8]. Gupta et al. calculated vehicle density using camera sensors and computational technology to regulate the traffic light [9]. Vidhiya et al. developed a model using Raspberry pi, IR transmitter and IR receiver to detect the traffic density. The model was mounted on either sides of road and aimed to reduce possible traffic jams and provide clearance for emergency vehicle whenever required [10]. Narkhede et al. developed an automatic registration plate recognition technique using image processing. In this model IR sensors were used to detect vehicles and by using image processing algorithm the registration number of vehicles are identified. SMS may be sent to traffic rules violators if any [11]. However, these models require thorough validation and also manual measures require a lot of man power for monitoring traffic manoeuvres throughout the day and even during night. Considering these aspects an attempt has been made in this study to detect over-speeding vehicles using image processing technique using Python programming language.

III. DATA COLLECTION

Video graphic method is adopted to collect the data [12] [13]. The video is recorded by placing the camera at the suitable position on typical weekdays during peak hours. The recorded video is then processed in python to extract various data like speed, composition, volume and also for manual extraction of speed and counting. In order to collect the data suitable site has to be selected. Following are some of the criteria's to be satisfied to select the study section.

- It should be two-lane or four-lane divided highway.
- The study stretch should be atleast 50 to 60m away from bus stop.
- There should be no cross road in the selected stretch.
- There should be no speed breakers and side interference like footpath, shops.
- There should be minimum pedestrian activities.

Reconnaissance survey is conducted in various parts of Mysuru city. The Reconnaissance Survey is done by walking beside of a stretch of 500 m, on the different identified locations, in order to select the best site. Reconnaissance Survey is needed because it guides us to choose the best location and also to identify the peak hour. The preliminary traffic survey is conducted throughout the day to identify the peak hours and is observed that the morning peaks our ranges from 8:00 am to 10:00 am and evening peak hour ranges from 4:00 pm to 6:00 pm.

From the Reconnaissance Survey the location has been identified that satisfy the criteria's mentioned above and other reason for selecting this location is, because of aqueduct which helped in better positioning of camera to record the video graphic data. The details of the selected study stretch are as follows.

- Road type: Four lane divided National Highway (NH-766)
- Road Name: Mysuru-Nanjangud Road.
- Land mark: Varuna Canal near Gudumandanahalli.
- Latitude: 12.2465645.
- Longitude: 76.6591946.
- Width of each lane: 3.5m.
- Length of study stretch: 50m.

The video data is collected from the above-mentioned location. On a typical week days, morning and evening peak hours has been identified by Reconnaissance Survey. Based on this information the traffic data was recorded in the evening peak hour (4:00 pm to 5:00 pm) through video graphic method. Since in morning time the sun light was affecting the video clarity and camera lens evening peak hour was selected. Before recording the video record, two reference lines has been marked on the road showing the entry and exit point. So that manual calculation of speed can be done by running the video at later stage. After marking the lines, the camera is placed on the top of the Varuna canal with swing angle of zero degree and video graphic data is recorded for one hour.

IV. MODEL DEVELOPMENT

The proto type model is developed in order to check the practical feasibility and real time implementation of the code developed. So that if the results obtained from the model and the results obtained from the manual method are almost near with permissible amount of errors the model can be implemented in real time traffic. It also reduces the time and man power required for classification and counting vehicles and also monitoring of over speed vehicles.

A. Components used

The type of hardware components used and there specifications are listed below (Fig. 2.).

1. Raspberry Pi 4 Model B with 4 GB ram.
2. Web cam of minimum 5mp.
3. Memory card class 10 of 32 GB.
4. Power bank of 3.1 Amp output.
5. OLED display.
6. Jumper wires.
7. Fan and Raspberry pi casing.

B. Development procedure

The coding which has been done in computer using Python software and which is tested using the prerecorded video for classification and counting of vehicles and also for speed identification, until satisfactory results are obtained. After that following steps are as followed.

1. The code is transferred to memory card and it is inserted to the Raspberry pi.
2. Then the camera is connected to Camera Serial Interface (CSI) camera port and power bank is connected to Micro USB power supply port.
3. The OLED (Organic Light-emitting Diode) connected to Display Serial Interface (DSI) display port.
4. The Raspberry pi is covered with casing along with fan mounted on it in order to avoid excess heating.
5. After all this setup model is placed on a highway such that there is obstruction for entire road width visibility and there is no over lapping of vehicles.
6. The results will be displayed on the OLED display.

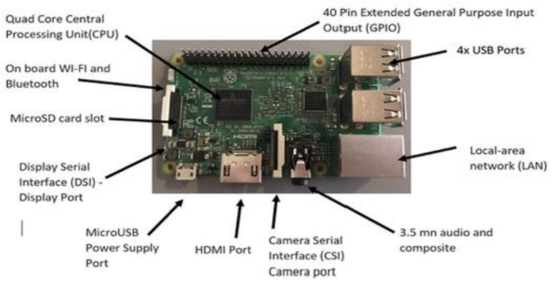

Fig. 2. Components of the model

C. Detailed working procedure of code

The steps involved in model development are shown in Fig. 3. and explained as follows:

Step1: Upload Video: The video graphic data collected from the selected site is used as input video to the application.

Step2: Frames Extraction: Frames is extracted from the input video. The number of frames per seconds (fps) depends on the quality of the video if higher the quality greater will be the number of frames.

Step3: Object Detection and classification: "YOLO" or "You Only Look Once" is a massive Convolutional Neural network for object detection and classification. YOLO's sorts its output by the 49 grid cells (7x7) that it divides the image into. The cells are represented in this array left to right, then top to bottom. The data is organized into three parts which I will describe below.

YOLO Output: Probability: This is the simplest one, there are 20 classes that the network can predict. 20 classes x 49 cells = the 980 numbers that are in this part of the vector. So 0 through 19 of the vector is the relative probabilities that the top left cell is categorized as each class.

YOLO Output: Confidence: Each cell in the grid is responsible for predicting two bounding boxes. Thus there is one number in this section of 98 of each of those bounding boxes in each cell (49 x 2). Each "confidence score" is the probability that there is an object within that bounding box

YOLO Output: Box Coordinates: This is where the boxes are predicted. Each box is represented with four numbers (x, y, width, height). These numbers are relative to the cell they're in and have to be scaled to be represented on the entire image. Once again, each cell predicts two bounding boxes so 2 boxes x 4 number per box x 49 cells = 392 numbers.

Artificial Intelligence has been witnessing a monumental growth in bridging the gap between the capabilities of humans and machines. Researchers and enthusiasts alike, work on numerous aspects of the field to make amazing things happen. One of many such areas is the domain of Computer Vision.

The architecture of a convolution network 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 overlaps to cover the entire visual area. An image is nothing but a matrix of pixel values, so we have to flatten the image (e.g. 3x3 image matrix into a 9x1 vector) and feed it to a Multi-Level Perceptron for classification purposes. A convolution network 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.

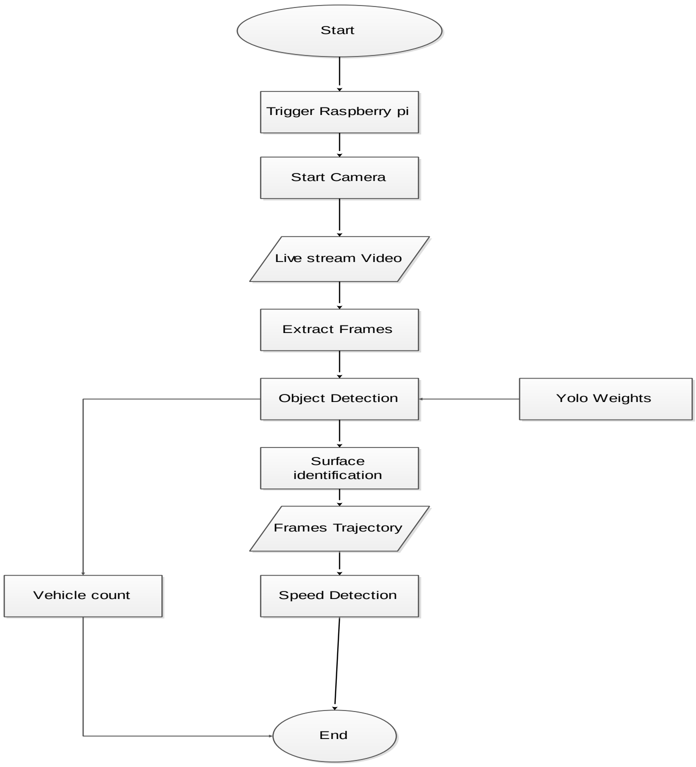

Fig. 3. Flow of model development

Step4: Road surface segmentation: This section describes the method of road surface extraction and segmentation. We implemented surface extraction and segmentation using image processing methods, such as Gaussian mixture modelling, which enables better vehicle detection results when using the deep learning object detection method. The road surveillance video image has a large field of view.

Step 5: Vehicle Trajectory Identification: Analysis of the trajectories of moving objects and the counting of multiple-object traffic information. Most of the roads are driven in two directions, and the roads are separated by isolation barriers. According to the direction of the vehicle tracking. We delete the trajectory that is not updated for ten consecutive frames, which is suitable for the camera scene with a wide-angle of image collection on the road under study. In this type of scene, the road surface captured by the camera is distant. In ten consecutive video frames, the vehicle will move farther away. Therefore, when the trajectory is not updated for ten frames, the trajectory is deleted. At the same time, the vehicle trajectory and the detection line will only intersect once, and the threshold setting thus does not affect the final counting result. If the prediction box fails to match in consecutive frames, the object is considered to be absent from the video scene, and the prediction box is deleted.

Step 6: Speed detection: The speed of the vehicle in each frame is calculated using the position of the vehicle in each frame, so the next step is to find the spots bounding, and the centre of gravity. Bubble centroid distance is important to understand the moving vehicle in consecutive frames and therefore is known as the frame rate for motion capture, the speed calculation becomes possible. This information must be recorded in a continuous array cell in the same size as the camera image captured because the distance travelled by the centroid is needed is a pixel with a specific coordinate on the image to determine the vehicle speed.

V. DATA EXTRACTION

The recorded video is processed in order to obtain the vehicle speed, composition and volume (Fig. 4.). The details of each component to be extracted by manual method is discussed below.

- **Vehicle Speed:** Speed is considered as a quality measurement of travel as the drivers and passengers will be concerned more about the speed of the journey than the design aspects of the traffic. It is defined as the rate of motion in distance per unit of time. Mathematically speed or velocity 'v' is given by.

$$v = \frac{d}{t} \quad (1)$$

Where, v = Speed of the vehicle in m/s.
d = Distance travelled in m.
t = Time in seconds.

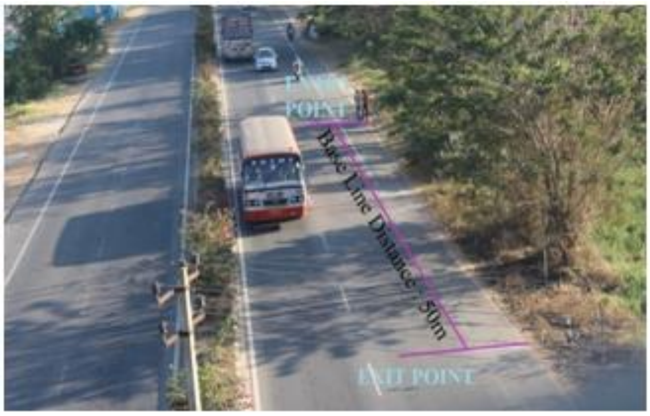

Fig. 4. Screen shot of site along with Base line

The Table I provides details of manually extracted speeds for different types of vehicles. It is observed that two wheelers are having an average speed of 61.5 kmph, Cars are having an average speed of 63.0 kmph, Buses are having an average speed of 56.2 kmph and Heavy vehicles are having an average speed of 67.5 kmph. The speeds obtained from manual extraction will be compare with speeds obtained from developed model in later stage.

Table-I: Speeds of different vehicle type

Vehicle type	Speed (kmph)		
	Minimum	Maximum	Mean
Two-Wheeler	30	90	61.5
Car	36	90	63
Bus	45	90	56.2
Heavy Vehicle	4	90	67.5
Considering all vehicles	30	90	62.05

Composition and volume: In this step the number of vehicles of same compositions are identified and classified from the video graphic data, by adopting manual method of data extraction. The extracted data is given in the Table II

Table-II: Vehicle composition and numbers

Vehicle type	Volume	Vehicle composition (%)
Two-Wheeler	642	54.13
Car	408	34.40
Bus	66	5.56
Heavy Vehicle	70	5.90
All vehicles	1186	100.00

From the Fig. 5. it is clear that Two-Wheeler is having higher composition of 54.13%, followed by Car which is having composition of 34.40, Heavy vehicle having composition of 5.90% and Bus having composition of 5.56%.

Retrieval Number: E1123069520/2020©BEIESP
DOI: 10.35940/ijeat.E1123.069520
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

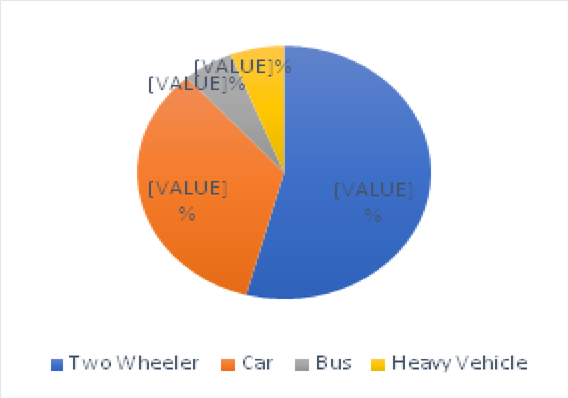

Fig. 5. Vehicle composition distribution

VI. VALIDATION OF MODEL

The validation of model is process of comparing field observed values and values from developed model [14]. In this process the average speed of different vehicle type obtained from manual extraction and developed model are compared and show in Table III. From the table it can observed the percentage error of two-wheeler is 6.30%, percentage error of Car 36.54%, percentage error of Bus 8.88% and percentage error of Heavy vehicle 9.52%. From the Table III it observed that the percentage error is less than 20%. The developed model is truly detecting the speed values and can be used for further development of porotype model.

Table-III: Data of average speed and percentage error.

Vehicle type	Average speed obtained from manual extraction (kmph)	Average speed obtained from Model (kmph)	Percentage error
Two-Wheeler	60.60	56.78	6.30
Car	63.92	40.56	36.54
Bus	45.00	49.00	8.88
Heavy Vehicle	42.00	38.00	9.52
All vehicle	52.88	42.52	19.59

VII. RESULTS AND CONCLUSION

A. RESULTS

The classified vehicular speeds from the selected study stretch is extracted by three methods: manual extraction, developed model, speed gun. The major outcomes of the study are listed below.

1. Manual extraction

- The average speed values of Two-Wheeler are 61.5 kmph.
- The average speed value of Car is 63 kmph.
- The average speed value of Bus is 56.2 kmph.
- The average speed value of Heavy vehicle is 42 kmph.

2. Developed model

- The average speed values of Two-Wheeler is 56.78 kmph.
- The average speed value of Car is 40.56 kmph.
- The average speed value of Bus is 49 kmph.
- The average speed value of Heavy vehicle is 38 kmph.

3. Percentage error

- The percentage error in speed calculation of Two-Wheeler is 6.30%
- The percentage error in speed calculation of Car is 36.54%
- The percentage error in speed calculation of Bus is 8.88%
- The percentage error in speed calculation of Heavy vehicle is 9.52%

The speed obtained from the prototype and field observed speed gun values will be compared.

VIII. CONCLUSION

The detailed study was conducted in order to detect the over speeding vehicles in the selected study stretch. An automatic speed detection model was developed using Python programming language. The developed code was also used to build a prototype model which detects type of vehicle along with the speed. The comparative analysis of speeds obtained manual and model shows that the developed model truly represents the field condition. The external validation of the developed prototype using a speed gun will enhance accuracy of the developed model. This prototype will be helpful to monitor the speeds automatically by reducing the requirement of manual traffic personnel.

ACKNOWLEDGMENT

Authors thank “The National Institute of Engineering – Center for Research and Development (NIE-CRD)” for financial support for conducting the study.

REFERENCES

1. Road Accidents in India-2018 published by Government of India Ministry of Road Transport & Highways Transport Research Wing (MORTH) in 2019.
2. Road Transport Year Book 2016-17 published by Government of India Ministry of Road Transport & Highways Transport Research Wing (MORTH) in 2019.
3. Mallikarjuna, C., Phanindra, A., & Rao, K. R. (2009). Traffic data collection under mixed traffic conditions using video image processing. Journal of transportation engineering, 135(4), 174-182.
4. Ranga, H. P., Kiran, M. R., Shekar, S. R., & Kumar, S. N. (2010). Vehicle detection and classification based on morphological technique. In 2010 International Conference on Signal and Image Processing.
5. Chandrasekhar, M., Saikrishna, C., Chakradhar, B., Kumar, P. P., & Sasanka, C. (2013). Traffic control using digital image processing. International Journal Of Advanced Electrical and Electronics Engineering, 2(5), 96.
6. Khanke, P., & Kulkarni, P. S. (2014). A technique on road traffic analysis using image processing. International Journal of Engineering Research and Technology, 3, 2769-2772.
7. Ramdas, G. O. (2014). Image Processing Based Traffic Light Control. International Journal of Science, Engineering and Technology Research (IJSETR), 3(4).
8. Uke, N., & Thool, R. (2013). Moving vehicle detection for measuring traffic count using opencv. Journal of Automation and Control Engineering, 1(4).
9. Gupta, P., Gadre, D. V., & Rawat, T. K. (2014). Real Time Traffic Light Control System (Hardware and Software Implementation). International Journal of Electronic and Electrical Engineering.

10. Vidhyia, M., Elayaraja, S., Anitha, M., Divya, M., & Divya Barathi, S. (2016). Traffic light control system using Raspberry-pi. Asian J. Electr. Sci.(AJES), 5(1), 8-12.
11. Narkhede, A., Nikam, V., Soni, A., Sathe, A., & Chiddarwar, G. G. Automatic Traffic Rule Violation Detection and Number Plate Recognition.
12. Punith B Kotagi, Likhitha M, Rohineesha A S (2019). Study of Traffic Flow Characteristics on Urban Divided Roads", Infra Road Tech 2019 - Fourth National conference on highways & Infrastructure Management.
13. Kotagi, P. B., & Asaithambi, G. (2017). Simulation framework for modeling bidirectional mixed traffic. In 2017 9th International Conference on Communication Systems and Networks (COMSNETS) (pp. 443-447). IEEE.
14. Kotagi, P. B., Raj, P., & Asaithambi, G. (2019). Modeling lateral placement and movement of vehicles on urban undivided roads in mixed traffic: A case study of India. Journal of Traffic and Transportation Engineering (English Edition).

AUTHORS PROFILE

Punith B Kotagi, Assistant Professor in Dept. of Civil Engineering, The National Institute of Engineering, Mysuru, India. Research area includes traffic simulation and modeling.

Punith Kumar N, final year student of Civil Engineering at The National Institute of Engineering Mysuru, Karnataka. His research area is Traffic Engineering which includes detection of over speeding vehicles using image processing techniques.

Parishwanath Madakari, final year student of Civil Engineering at The National Institute of Engineering Mysuru, Karnataka. His research area is Traffic Engineering which includes detection of over speeding vehicles using image processing techniques.

Omkar V H, final year student of Civil Engineering at The National Institute of Engineering Mysuru, Karnataka. His research area is Traffic Engineering which includes detection of over speeding vehicles using image processing techniques.

Vishwanath, final year student of Civil Engineering at The National Institute of Engineering Mysuru, Karnataka. His research area is Traffic Engineering which includes detection of over speeding vehicles using image processing techniques.

Retrieval Number: E1123069520/2020©BEIESP
DOI: 10.35940/ijeat.E1123.069520
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication