Mini Project Report On

Vehicle Detection and Counting Interface for Traffic Analytics

for the course

IT416 : Computer Vision

Submitted by

Abhishek Kaswan(181IT201)
Yash Parakh(181IT253)

VII SEM B.Tech (IT)

Under the guidance of

Professor Dinesh Naik
Dept of IT, NITK Surathkal

in partial fulfillment for the award of the degree

of

Bachelor of Technology
in
Information Technology

at

Department of Information Technology

National Institute of Technology Karnataka, Surathkal.

November 2021

Vehicle Detection and Counting Interface for Traffic Analytics

Abhishek Kaswan
181IT201
Information Technology Department
National Institute of Technology Karnataka
Surathkal, India
abhishek.181it201@nitk.edu.in

Yash Parakh
181IT253
Information Technology Department
National Institute of Technology Karnataka
Surathkal, India
yashparakh5@gmail.com

***Abstract*—Vehicles have become the basic necessity of the people for their daily transport also. Due to this, more and more vehicles are running on the roads which increases the density of automobiles on the road. This project can be used in many areas like crowd counting, traffic analysis and many more. Amongst some of the important components of a smart city is the automated traffic control system. On integrating this system near traffic lights, you could easily track a number of useful things simultaneously. Most of the traditional methods start with Background Subtraction methods and Thresholding methods, but they vary in the method of implementing the Region of interest (ROI) and noise removal techniques. Source video is read one frame at a time with OpenCV. Every frame read is then pre-processed by SSD with Mobilenet model, which is developed on tensorflow. The detected vehicle image is used to predict the size using image area and also is used for color recognition by KNN is trained with R,G,B color histogram.**

Keywords—Computer Vision, Vehicle, Detection, Recognition, Frame, Counting, Classification, Color

I. INTRODUCTION

A. OVERVIEW

Many places are equipped with surveillance systems that combine AI with cameras. These technologies are utilised by government organizations to various private facilities. These AI-based cameras help in many ways, and one of the main features is to count the number of vehicles. It can be used to count the number of vehicles passing by or entering any particular place.

Vehicles have become the basic necessity of the people for their daily transport also. Due to this, more and more vehicles are running on the roads which increases the density of automobiles on the road. So, keeping track of the number of vehicles that are passing through the lane/roads has become one of the problems. Due to which the detection of the vehicle and its classification became one of the interesting problems of computer vision.

B. MOTIVATION

The thought of automated smart energy systems, electrical grids, one-touch access ports – it's an enthralling concept! One of the core components of a smart city is automated traffic management. If we could integrate a vehicle detection system in a traffic light camera, you could easily track a number of useful things simultaneously.

II. LITERATURE SURVEY

Many works for this problem have been implemented by using Traditional methods as well as Machine Learning Techniques. Most of the traditional methods start with segregating background and then thresholding, but they vary in the method of implementing the region of interest and removal of noise.

"Vision-based vehicle detection and counting system using deep learning in highway scenes" by Huansheng Song; Haoxiang Liang; Huaiyu Li; Zhe Dai; Xu Yun: The object size of the vehicle depends on the viewing angle and affects vehicles far away. This paper proposes a vision-based vehicle detection and counting system. A new high definition highway vehicle dataset is published in this study.

"Vehicle Detection and Classification using Image processing" by R.Roopa Chandrika; N.S.G. Ganesh;A. Mummoorthy; K. M. Karthick: The paper proposes a method to classify vehicles by processing of frames. The task of vehicle detection and counting is broken down into six steps. It proposed to develop an unique algorithm using Gaussian mixture model and blob detection methods.

"Research on visual vehicle detection and tracking based on deep learning" by Yaoming Zhang; Xiaoli Song; Mengen Wang; Tian Guan: Based on the structure of YOLOV3 convolutional neural network, the vehicle video detection experiment is carried out under the framework of TensorFlow, and the detection results are evaluated by the detection accuracy and error detection rate.

"Vehicle detection and counting under mixed traffic conditions in Vietnam using YOLOV4" by Vuong Xuan Can; Phan Xuan Vu; Mou Rui-fang; Vu Trong Thuat: Most of those researches do not focus on vehicle detection in the mixed traffic flow like Vietnam where on-road motorcycles are used frequently. Because of non-lane-based movements of vehicles on roads, traffic flow is very complex in comparison with homogeneous traffic like developed countries.

III. Problem Statement

To develop a vehicle detection system using opencv and tensorflow object counting API. It should be able to detect vehicles and try to extract relevant data like speed, color and size of the vehicle.

IV. Objectives

- To detect and classify vehicles (car, truck, bicycle, motorcycle, bus) along with direction of travel and their approximate color.
- To develop vehicle counting systems capable of performing speed estimation and traffic flow estimation along with vehicle detection, tracking, counting, classification (into light medium and heavy).
- The focus is on development of the vehicle counting system and it's comparative analysis on datasets.

V. Methodology

Dataset - The training data has a huge importance in classification accuracy, maybe the accuracy can be higher with more suitable training data.

We have implemented our working model on different recorded videos which were taken from Davis king`s dib library. It includes image files along with two XML files, training.xml and testing.xml. Each image element in the file includes the file path along with a number of bounding box objects. For each image, all visible front and rear views of vehicles are labeled as such.

Another important thing is lightning and shadows. In the test images, the images which were taken under bad lighting conditions and with shadows should be classified right because some filtering algorithm would be implemented before the test images are sent to KNN classifier Thus, accuracy can be improved.

Data Preprocessing -For Tensorflow object detection API, data points for object detection are needed such as height and width of the image, list of bounding boxes coordinates, list of class labels and also Tensorflow encoded image. To achieve this, raw image dataset will be converted into tensorflow format dataset with corresponding class label files. After that, it can be used to train Tensorflow object detection API.

A. Vehicle Detection

For real-time computer vision , OpenCV is a library of programming functions, it is an open source software library. OpenCV (Open Source Computer vision Library) is aimed to give a common platform for computer vision applications. Also it is used to speed up the use of machine perception in the sales product. In the beginning, read input video frames using OpenCV. Now the resulting frames would be processed by the SSD mobilenet model.

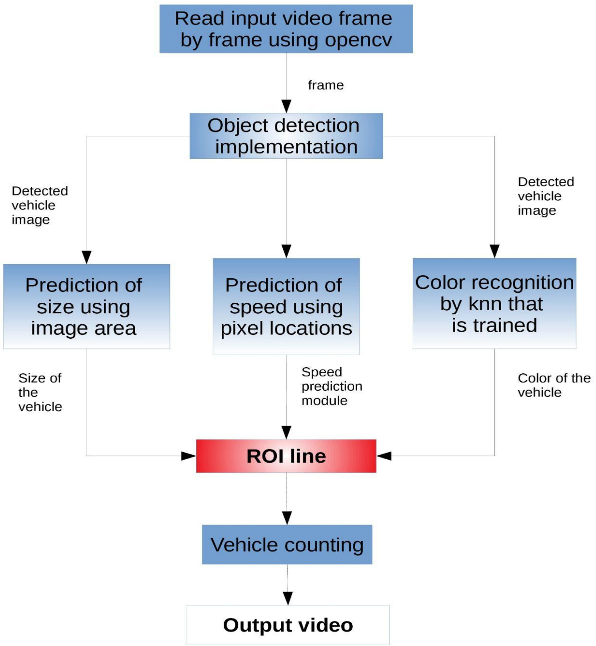

Fig 1. Architecture

The model will detect multiple detection from the video frames. As the detector did not perform well on test images, to fine tune this SSD detector mentioned dataset was used. We trained our models by splitting them into a training dataset 80% and validation dataset 20% .After this using transfer learning , we set up the Tensorflow Object Detection API configuration file for training.

We will receive a detected vehicle image and pixels which will be used to predict several outputs such as pixel would help in predicting speed while the image will be used for color recognition where it would be able to differentiate color for a particular vehicle. And at last vehicle counting is captured when any vehicle passes through the ROI (region of interest)line. At last along with csv file output video will be received. There is data for counting, color, size and speed of the vehicle.

This is divided into three main tasks including

- Cleaning of Frame
- Vehicle Detection
- Vehicle Counting & Classification

Finding the contour of the vehicle in the frame is the important part, to map the location as well as defining the shape of that vehicle and to get the precise flow and location of the vehicle.

By the size of the bounding box, classification of the vehicle is done, and then finally counting of the vehicle can be done by creating a line and finding the number of centroids which passes through that line. When the vehicle passes over the line and counts, the color of the region of interest line changes to green. But this method leaves space for a lot of errors. And one such major error is the distorted object detection resulting from the shadows which overlap the vehicles. Vehicle counting and classification(meaning vehicle size/type) is also implemented here.

For color of the vehicle - The detected vehicle image also is used to color recognition by KNN is trained with R,G, B color histogram, this module is to distinguish color in detected vehicle image frames. It can classify White, Black, Red, Green, Blue, Orange, Yellow and Violet. To classify more colors or improve the accuracy work on the training data is suggested or learn more about other color features such as Color Moments or Color Correlogram. The general workflow is given below:

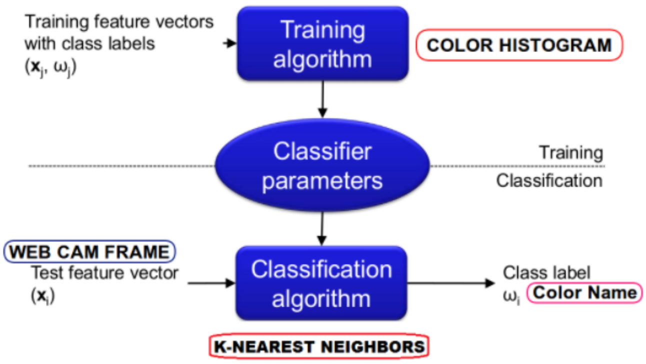

Fig 2. Color Classification

There are 3 main steps to understand basic workflow used for color classifting:

- Feature Extraction: Its simple explanation is, It compares features found in the image with other available features. Now feature extraction is performed here for getting the R, G, B Color Histogram values of training images.
- Training K-Nearest Neighbors Classifier: K nearest neighbors is a simple algorithm .It stores all given cases and also using similarity measures like distance functions ,classifies new cases. Train KNN classifier by R, G, B Color Histogram values here.
- Classifying by Trained KNN: Read camera per frame and perform feature extraction on each frame. At last classify the mean color of it by using a trained KNN classifier.

For size/type of the vehicle - The image received from API is used to predict the size using the image area, there is generally a size prediction module in order to detect the size of the vehicle. Computer vision system is designed to collect timestamps of cars to measure speed (with a known distance). scale_constant is used for manual scaling because camera calibration is not performed during speed estimation.

There were functions available for visualizations where it would get an image and perform some visualization on it. These functions do not return any value but they modify the image itself. Apart from that Label map utility functions are also available which returns categories list compatible with evaluation.

The code for the following project is in the following repository: "https://github.com/YashParakh5/CV_Project"

VI. Results and Analysis

- For vehicle counting and classification, as shown in the workflow, after capturing the video we implemented object detection.
- Recognition of approximate vehicle color is also completed. Vehicle color prediction has been developed using OpenCV via K-Nearest Neighbors Machine Learning Classification Algorithm is Trained Color Histogram Features.
- Prediction of the speed of the vehicle is implemented using OpenCV via image pixel manipulation and calculation.
- Detection of vehicle direction of travel is also shown in the results.The images of detected vehicles are cropped from video frames and they are saved as new images.
- The program gives a .csv file as an output which includes vehicle type/size, vehicle color, movement direction and vehicle speed (km/h), after the end of the process for the source video file.

Figure 4: Vehicle Counting Passing through ROI Line

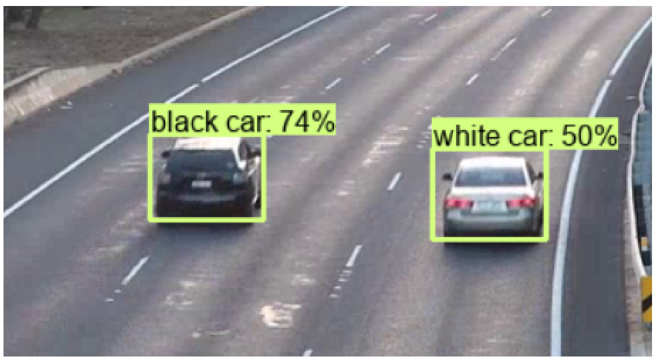

Figure 5: Color Prediction with Score

Tensorboard visualization was used during the implementation and it helped to understand if the training was succeeding or not. Table 1 and Table 2 below show the evaluation of the model including precision and recall of detection boxes. As we can see, the SSD model achieves 65.71% mAP @ 0.5 IoU.

Mean average precision of a model is not able to explain how tight the bounding boxes of the model are. The reason behind is, because that info is actually combined with the correctness of predictions. To know the box`s tightness IoU should be directly evaluated.

Across Scales	Small	Medium	Large
mAP	0.058	0.21	0.56
AR@100	0.11	0.29	0.648

Table 1: mAP and AR across scales like small,medium and large.

Intersection over Union (IoU)	Average Precision
0.5	0.652
0.75	0.416
0.50:0.95	0.39

Table 2: Precision evaluation of the model.

VII. Innovation

- Trained SSD on the Davis King's vehicles dataset to differentiate the front and rear views of the vehicles.
- Predict the vehicle's size using the image area and vehicle's speed and direction using pixel locations.

VIII. Conclusion

This is a project done for IT416 computer vision project. It uses various technologies to detect, track and count the vehicles in the camera's view. Compared with the traditional method of monitoring vehicle traffic by hardware, the method of this paper is low in cost and high in stability and does not require large-scale construction or installation work on existing monitoring equipment.

The detected vehicle image also is used to color recognition by KNN is trained with color histogram, this module is to distinguish color in detected vehicle image frames. There is also detected vehicle image pixel location to predict the speed and directions using pixel location. After all detection passthrough the vehicle image will be detected in counting detected vehicles after the vehicle drives through the region of interest line. Lastly the video will be output with extract data from the counting, size, color, speed and direction of the vehicle.

Authors and Affiliations

This report is a part of a mini-project for IT 416 Computer Vision. Following are the members who contributed: (Team No. 59)

- Abhishek Kaswan, Roll No. 181IT201
- Yash Parakh, Roll No. 181IT253

References

[1] Song, H., Liang, H., Li, H. et al. Vision-based vehicle detection and counting system using deep learning in highway scenes. Eur. Transp. Res. Rev. 11, 51 (2019).

[2] Chandrika, R. Roopa et al. "Vehicle Detection and Classification using Image processing." 2019 International Conference on Emerging Trends in Science and Engineering (ICESE) 1 (2019): 1-6.

[3] Zhang, Yaoming & Song, Xiaoli & Wang, Mengen & Guan, Tian & Liu, Jiawei & Wang, Zhaojian & Zhen, Yajing & Zhang, Dongsheng & Gu, Xiaoyi. (2020). Research on visual vehicle detection and tracking based on deep learning. IOP Conference Series: Materials Science and Engineering. 892. 012051. 10.1088/1757-899X/892/1/012051.

[4] Vehicle detection and counting under mixed traffic conditions in Vietnam using YOLOV4 by Vuong Xuan Can; Phan Xuan Vu; Mou Rui-fang; Vu Trong Thuat: (2021)

[5] Detection and classification of vehicles for traffic video analytics by Ahmad Arinaldi; Jaka Arya Pradana; Arlan A.Gurusinga: (2018)

[6] Research on the Cascade Vehicle Detection Method Based on CNN by Jianjun Hu; Yuqi Sun; Songsong Xiong: (2021)

[7] Anandhalli, Mallikarjun & Baligar, Vishwanath. (2015). Improvised approach using background subtraction for vehicle detection. Souvenir of the 2015 IEEE International Advance Computing Conference, IACC 2015. 303-308. 10.1109/IADCC.2015.7154719.

[8] Bahnsen, Chris H. and Moeslund, Thomas B., "Rain Removal in Traffic Surveillance: Does it Matter?", IEEE Transactions on Intelligent Transportation Systems, 2018, pp. 1-18, doi: 10.1109/TITS.2018.2872502

[9] A. Prati, I. Mikic, M. Trivedi, R. Cucchiara, "Detecting moving shadows: algorithm and evaluation", IEEE Trans. Pattern Anal. Mach., Vol. 25, Iss. 3, pp. 918–923, 2003.

[10] M. Won, T. Park, and S. H. Son, "Toward mitigating phantom jam using vehicle-to-vehicle communication," IEEE Trans. Intell. Transp. Syst., vol. 18, no. 5, pp. 1313–1324, May 2017.

[11] FHWA. The 2016 Traffic Monitoring Guide. Accessed: Apr. 12, 2019. [Online]. Available: https://www.fhwa.dot.gov /policyinformation/ tmguide/tmg_fhwa_pl_17_003.pdf

[12] M. Won, S. Sahu, and K.-J. Park, "DeepWiTraffic: Low cost WiFi-based traffic monitoring system using deep learning," 2018, arXiv:1812.08208. [Online]. Available: http://arxiv.org/abs /1812.08208

[13] H. Lee and B. Coifman, "Using LiDAR to validate the performance of vehicle classification stations," J. Intell. Transp. Syst., vol. 19, no. 4, pp. 355–369, Oct. 2015.

[14] FHWA. The 2016 Traffic Detector Handbook. Accessed: Apr. 12, 2019. [Online]. Available: https://www.fhwa.dot.gov /publications/research/ operations/its/06108/06108.pdf

[15] Á. Virginás-Tar, M. Baba, V. Gui, D. Pescaru and I. Jian, "Vehicle counting and classification for traffic surveillance using Wireless Video Sensor Networks," 2014 22nd Telecommunications Forum Telfor (TELFOR), Belgrade, Serbia, 2014, pp. 1019-1022, doi: 10.1109/TELFOR.2014.7034580

[16] Dabiri, S., Heaslip, K., 2018c. Transport-domain applications of widely used data sources in smart transportation: A survey. arXiv preprint arXiv: 1803.10902

[17] Hallenbeck, M.E., Selezneva, O.I., Quinley, R., 2014. Verification, refinement, and applicability of long-term pavement performance vehicle classification rules. Technical report.