VEHICLE MONITORING FOR VIOLATION AND TRAFFIC DENSITY ANALYSIS

Akhil Abraham
akhilabraham34@gmail.com

Jaideep Gowda
jdgowda75@gmail.com

Benson P Sabu
benzps01@gmail.com

Dept. of Information Technology,
Fr. Conceicao Rodrigues Institute of Technology,
Navi Mumbai, India

***Abstract* - Vehicles have become an important aspect in all human's everyday life. It has become the source of livelihood, source of travel, transportation, etc. But as the population is vastly growing, so is the number of vehicles. Hence it becomes a real challenge for managing this heavy vehicular population. It becomes difficult to find a vehicle in this huge thick dense metal forest only by mere naked eye especially when a robbery is in place. Current systems take pictures of the vehicle number plates and use the images to detect the vehicle owner. This system takes a lot of time and effort and the number plates can be switched with vehicles with different models. The aim of this project is to reduce this time and effort required to identify the vehicles by creating a centralized system and also identify the density of the vehicular population. This centralized system will automatically monitor the vehicles based on their number plates and vehicle make model and verify this with the database consisting of all the vehicle owners. Also this system will include vehicle counting to correctly identify the vehicle density of an area. This system also helps to analyze traffic in a region and verifies the details of owners like insurance validity, any tickets, toll bills, etc.**

Keywords: Vehicle Detection, License Plate, YOLOV3, Traffic data.

I. Introduction

Vehicles have become an integral part of human life [2]. It has become the source of livelihood, source of travel, transportation. But as the population is vastly growing, so is the number of vehicles. Hence it becomes a real challenge for managing this heavy vehicular population. It becomes difficult to find a vehicle in this huge thick dense metal forest only by mere naked eye especially when a robbery is in place. Multiple vehicle detection is a promising and challenging role in intelligent transportation systems and computer vision applications [3]. The major research issues were found in the recent kinds of literature in the ITS sector which is closely related to the real-time traffic environmental problems such as occlusions, camera oscillations, background changes, sensors, cluttering, camouflage, varying illumination changes in a day- and sunny and at night time vision.[3] Before deep learning is widely applied in the field of computer vision, researchers conduct object detection by establishing some mathematical models based on certain empirical knowledge. The traditional object detection the method is generally divided into three stages: firstly, select candidate areas on a given image by using sliding windows of different sizes, then extract the characteristics of these candidate regions, finally, the trained classifier is used for classification. The existing works on number plate recognition essentially segment the number of plate characters and use optical character recognition (OCR) techniques to identify all the segmented characters. But we completely neglect the character segmentation stage and pass the number plate region of interest (ROJ) to YOLO directly for the characters to be recognized. Our ANPR system investigates an input image and outputs the potential characters present in the number plate [4]. Although there are plenty of ANPR algorithms proposed in the past few years, till date ANPR is a complex and challenging task to recognize the characters in a number plate accurately in the wild with multiple instances with partial occlusion or from the arbitrary capturing viewpoint. One of the most important issues in traffic control and vehicle identification in metropolitan areas as well as recording theft and detection of stolen vehicles automatically using security and control cameras in cities and roads, is to recognize vehicle types and car plates in order to identify and differentiate them. In recent years, the accuracy and speed of these control systems has developed significantly. However, there are still ways to circumvent these systems by counterfeiting and modifying number plates. With advancements and provision of rapid and accurate machine vision methods and using these methods to identify vehicle type and model, and then integrating these two methods, these violations can be largely prevented. In this study, a new approach was proposed to detect the type, model and location of vehicles found in images. In this approach, using the images of vehicles available in dataset, an object detection network was trained using YOLO [1] (You Only Look Once) algorithm in order to identify vehicle type and model along with their location in the image. Because of the time-consuming and bulky implementation of multiple combined deep networks, a CNN (convolutional neural network) classifier was used to identify the model, and an object detection netw

II. LITERATURE SURVEY AND ANALYSIS

A. Related Work

Various studies have been conducted on modelling vehicles, and various deep learning-based and image processing methods have been used in the literature to extract the features and classify them. In [6], a convolutional network is trained using the set of introduced images to identify the location of vehicles and extract features from that location. Then, using a SVM classifier, feature vectors of vehicles and their classes are categorized. In the method proposed in

[7], the front part of vehicles is identified and given the details such as the distance between lights and size of the axle, as well as the location of lines and margins of the image and extracted features using SIFT operator, a KNN classifier was trained. Some errors of this algorithm include manipulation or changing of the vehicle logo which creates errors in detection. In [8], by measuring lines and margins with certain ratios proportional to the number plate, the location was identified and was used as the vehicle image. Then using the SIFT operator, a KNN classifier was trained. Some errors of this algorithm include manipulation or changing of the vehicle logo which creates errors in detection. In [8], by measuring lines and margins with certain ratios proportional to the number plate, the location was identified and was used as the vehicle image. Then using the SIFT operator, features were extracted in several sections and then, were categorized by calculating Euclidean distance using KNN algorithm. In [9] a vehicle type classification system based on deep learning is proposed. The system uses Faster R-CNN to solve the task. Experimental results show that the method is not only time- saving, but also has more robustness and higher accuracy. In [10], a combination of CNN network and an SSD object detection network with MobileNet v2 architecture was used to identify vehicle model and its location in the image. One of the disadvantages of the methods mentioned above is the low number of vehicles that the system can identify. In addition, most of the proposed methods cannot identify more than one vehicle or increase the time to determine if there is more than one vehicle. The aim of this study is to propose a method to identify all vehicles available in the dataset from all angles with various optical and visual conditions. In [11], is proposed a novel deep learning approach for Make and model recognition using the SqueezeNet architecture. The frontal views of vehicle images are first extracted and fed into a deep network for training and testing. The SqueezeNet architecture with bypass connections between the Fire modules, a variant of the vanilla SqueezeNet, is employed for this study, which makes this system more efficient.

III. PROPOSED SYSTEM

.

A. Design Goal

1: Real Time Detection of Vehicles and License Plate
2: Vehicle Classification based on the colour and logo
3: License Plate Localization with character Extraction
4: Comparing the Characters that was Extracted with the Main Database
5: A Robust Vehicle Counting System
6: Analysis of the traffic violation and count for better understanding and Predictions

B. System Model

Figure 1.0 As Mentioned our Model includes 6 Major Functionalities including the Detector that generates the bounding boxes of the vehicle Where all the filtration takes place. The Data then send to the main database where it checks whether that owner has violated any traffic rule. The counter is responsible for vehicle counting. These would help in the Management of traffic data, improving traffic flow across the city and Traffic forecasting is an integral part of the process of designing road facilities, starting from investment feasibility study to developing of working documentation.

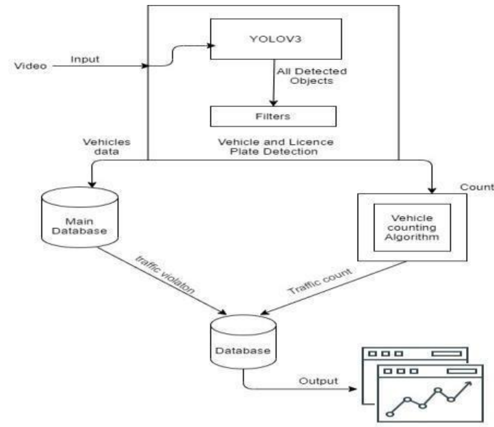


Fig1: System Architecture

The main challenge was to create a robust car number plate data-set. The dataset contains 6,500 of Indian car number plate images of moving and static cars. The dataset contains images captured on both day and night time. The dataset is split as 90% for training and 10% for testing. Every single image has their corresponding annotations which are available in a text format (.txt).

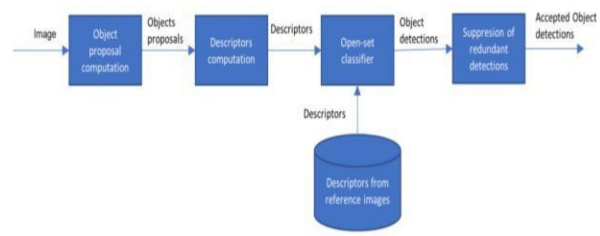


Fig2: Illustrates the object Detection pipeline, which is explained in this section.

GUI-software called YOLO Mark was used for marking bounding boxes of objects and generating annotation files [23].propose the number plate recognition system based upon the YOLO which works efficiently under different conditions. General rule for the training dataset is to include set of relative sizes of objects that you want to detect:

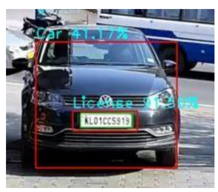
train_network_width *train_obj_width / train_image_width -
= detection_network_width * detection_obj_width / detection_image_width

train_network_height *train_obLheight / train_image_height -
= detection_network_height * detection_obj_height / detection_image_height

Parameters	Value
Batch	64
Subdivision	16
width	320
height	320
channels	3
Momentum	0.9
Decay	0.0005
Angle	0
Saturation	1.5
Exposure	1.5
Hue	1
Learning Rate	0.001
Burn_in	1000
Max_batches	8000
Policy	400000,450000
Scales	0.1,0.1

Table I. Parameters Configurations used while training the Dataset

A single YOLO model is used for both number plate detection and recognition. We trained a 37-c1ass CNN to recognize all the characters in the number plate and detect the number plate itself (i.e., Number Plate, A-Z (except 0), 0-9). Since 0 and 0 are recognized as the same, we neglect 0 for training. In order to use the YOLO algorithm, the filter size present in the final convolutional layer should be tweaked so that, it matches the number of objects. [5]



The filter size is calculated by filters = (classes + 5) * K (I)
where k=number of masks in one yolo layer (k=3) classes = number of classes to be predicted (C =4). Therefore, filters = (4 + 5) * 3. So, then the number of filters used in our model is 27. YOLO algorithm uses anchor boxes A to predict the output bounding boxes where each of the boxes contains four coordinates (x, y, w, h), confidence score and also their class probabilities C [5]. Fig. 3 shows the number plate recognition pipe line where an input image is fed into the YOLO model, then if the number plate is detected, the corresponding number plate region of interest (ROI) is extracted and this image is again fed into the YOLO model to get recognized. Finally, the recognized output is sorted from left to right so that it's in the correct order as in the number plate. We achieve a 100 % accuracy in Number Plate Detection and 91% accuracy in number plate recognition. Fig.3 shows the number plate recognition using the proposed approach.

Stages	Accuracy
Vehicle Detection	90%
License Plate Detection	95%
Score of the Output	92%

Table II. Results on Object

Detection License Plate Localization and Character

Extraction

1. License Plate Localization: The first step is to detect the License plate from the car. After Detection And Localization takes place.

cropped = img [y:y + h, x:x + w]

here the coordinates of the Image captured which would Further Help in Image Processing.

2. Processing of License Plate

Second Step Resize the image to the required size and then grayscale it. Gray scaling is common in all image processing steps. This speeds up other following process since we no longer have to deal with the color details when processing an image. Every image will have useful and useless information, in this case for us only the license plate is the useful information the rest are pretty much useless for our program. This useless information is called noise. Normally using a bilateral filter (Blurring) will remove the unwanted details from an image.

destination_image = cv2.bilateralFilter(source_image, diameter of pixel, sigmaColor, sigmaSpace).

You can increase the sigma color and sigma space from 15 to higher values to blur out more background information, but be careful that the useful part does not get blurred. The output image is shown below, as you can see the background details (tree and building) are blurred in this image. Further we found the threshold value and finally using Bitwise not Operation to mask or subtract the background. And get the output

thresh = cv2.threshold(plate, 0, 255, cv2.THRESH_BINARY | cv2.THRESH_OTSU) [1]

roi = cv2.bitwise_not(thresh)

3. Character Recognition: Now, the new image that we obtained in the previous step The Final step in this Number Plate Recognition is to actually read the number plate information. We used the pytesseract package to read characters from image

config = ("--psm 10")
text = pytesseract.image_to_string(roi, lang="eng",config=config)

4. Color Recognition of the Vehicle:

Inorder to Detect the Color of the vehicle We Used KNN Algorithm. To define or extract the color pallets and create the hexadecimal value the method that will be used is KMeans Clustering. K-Means clustering is a type of unsupervised learning, which is used when you have unlabeled data (i.e., data without defined categories or groups). The goal of this algorithm is to find groups in the data, with the number of groups represented by the variable K. First, we convert the image to points that our clustering algorithm can use. Next, the color distance is calculated using the Euclidean distance formula, which is

$$d(p,q) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2}$$

After that, we find the center for a set of points by adding the values for each dimension and divide by the number of points. Then, the clusters are sorted and the value will be converted into hexadecimal form. To get the color name of the hexadecimal value the method that will be used is K-Nearest Neighbors (KNN) algorithm. K-nearest neighbors is a supervised classification algorithm that needs labelled data to train on. With the given data, KNN can classify new, unlabeled data by analysis of the 'k' number of the nearest data points. Thus, the variable 'k' is considered to be a parameter that will be established by the machine learning engineer. This KNN algorithm will classify the color using the color distance value.

Vehicle Counting.

To start with vehicle counting the first thing we have to do is to track the object and assign unique IDs. We used Alex Bewley's SORT algorithm.It is basically a real time tracking for 2D multiple tracking in video sequence.This algorithm is designed for online tracking applications where only past and current frames are available

IV. SUMMARY AND FUTURE SCOPE

In this paper, the types and models of vehicle detection along with its license, and the vehicular density in the image are studied. The project involves reducing the time and effort required to identify the vehicles by creating a centralized system. This centralized system will automatically monitor the vehicles based on their number plates and vehicle make model and verify this with the database consisting of all the vehicle owners. This system also helps to analyse traffic in a region and verifies the details of owners like insurance validity, any tickets, toll bills, etc.

To increase accuracy of detection, varied dataset including images of vehicles in different situations, lighting conditions, different angles, etc. Also in addition, we intend to present an application for street level detection like this system can be installed on Police Vehicles.

References

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, ''You only look once: Unified, real-time object detection,'' in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2016, pp. 779–788.

[2] Mitesh Rathod; Ranjana Gite; Anushka Pawar; Shoomiren Singh; Pranav Kelkar
An air pollutant vehicle tracker system using gas sensor and GPS

[3] D. Sudha J. Priyadarshini an intelligent multiple vehicle detection and tracking using modified vibe algorithm and deep learning algorithm

[4] R Naren Babu V Sowmya K P Soman Indian Car Number Plate Recognition using Deep Learning

[5] https://github.com/AlexeyABlYolo mark

[6] L. Yang, P. Luo, C. C. Loy, and X. Tang, "A large-scale car dataset for fine-grained categorization and verification," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2015, vol. 07-12-June-2015, pp. 3973–3981.

[7] A. Psyllos, C. N. Anagnostopoulos, and E. Kayafas, "Vehicle model recognition from frontal view image measurements," in Computer Standards and Interfaces, 2011.

[8] A. P. Psyllos, C. N. E. Anagnostopoulos, and E. Kayafas, "Vehicle logo recognition using a sift-based enhanced matching scheme," IEEE Trans. Intell. Transp. Syst., vol. 11, no. 2, pp. 322–328, Jun. 2010.

[9] X. Wang, W. Zhang, X. Wu, L. Xiao, Y. Qian, and Z. Fang, "Real-time vehicle type classification with deep convolutional neural networks," in Journal of Real-Time Image Processing, 2019, vol. 16, no. 1, pp. 5–14.

[10] B. Satar and A. E. Dirik, "Deep learning-based vehicle make- model classification," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2018, vol. 11141 LNCS, pp. 544–553.

[11] H. J. Lee, I. Ullah, W. Wan, Y. Gao, and Z. Fang, "RealTime vehicle make and model recognition with the residual squeezenet architecture," Sensors (Switzerland), vol. 19, no. 5, Mar. 2019.