A Project Report on

Autonomous Traffic Monitoring and Controlling

Submitted in partial fulfillment of the requirements of

Bachelor of Engineering

in

Computer Engineering

by

Yuvraj Yadav(16102040)
Ashwin Shenolikar(16102037)
Tanmay Sule(16102032)

Under the Guidance of

Prof. Amol Kalugade

Department of Computer Engineering
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI

Academic Year 2019-2020

Approval Sheet

This Project Report entitled ***"Autonomous Traffic Monitoring and Controlling"*** Submitted by ***"Yuvraj Yadav"(16102040), "Ashwin Shenolikar"(16102037), "Tanmay Sule"(16102032)*** is approved for the partial fulfillment of the requirenment for the award of the degree of ***Bachelor of Engineering*** in ***Computers*** from ***University of Mumbai***.

Prof. Amol Kalugade
Guide

Prof. S. H. Malave
Head Department of Computer Engineering

Place:A.P.Shah Institute of Technology, Thane
Date:

CERTIFICATE

This is to certify that the project entitled ***"Autonomous Traffic Monitoring and Controlling"*** submitted by ***"Yuvraj Yadav" (16102040),"Ashwin Shenolikar" (16102037),"Tanmay Sule" (16102032)*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Computer***,to the University of Mumbai,is a bonafide work carried out during academic year 2017-2018.

Prof. Amol Kalugade
Guide

Prof. S. H. Malave
Head Department of Computer Engineering

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane
Date:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Yuvraj Yadav 16102040)

(Ashwin Shenolikar 16102037)

(Tanmay Sule 16102032)

Date:

Abstract

Road travel is an everyday part of most of the people living in urban areas. It is essential for each such commuter to have a hassle-free journey when they travel by road. Traffic rules are introduced in order to make roads traffic-free and convenient. However, the enforcement of these is not always as good. This implementation proposes a way to enforce traffic rules in a way that violators are actually held liable for breaking traffic rules. This implementation aims to accomplish this using a combination of image processing, machine learning algorithms and optical character recognition. The points of input are video feeds from CCTV cameras mounted on traffic signals, which will be processed and trained to identify and classify vehicles into categories such as two-wheelers, four-wheelers, logistic vehicles and public transport vehicles. The vehicles identified to be breaking traffic rules will be scanned for their number plates, using which the information of the owner will be extracted, and a fine will be generated and charged to the owner. Another application of this project is to gather data and train a model which can identify the density of traffic in a given area, and accordingly modify traffic signal timings to avoid occurrences of traffic jams.

Contents

List of Figures

Chapter 1

Introduction

The daily life of most people living in suburban and urban settlements today consists of a commute. The daily commute of each citizen usually consists of detrimental factors. Due to various reasons, delays are caused in the daily commute,leading to increased travel times which adds to the hectic schedule. Hence, the model aims to ease this part of the citizens life by enforcing traffic rules in an automated fashion. The usage of machine learning and image processing is done in order to achieve this.

1.1 Problem Definition

Daily commute of citizens in urban cities of India more often than not consists of facing the problem of slow travel due to various incidents.The root cause of this problem is as follows: Over speeding, lane cutting in dense traffic situations give rise to phantom traffic jams[1][2], vehicles crossing traffic signals irrespective of whether their lane had a green light or not, and other such reasons. The main issue faced while tackling these issues is the appropriate handling of recorded instances. The culprits may get away with breaking traffic rules without any repercussions. Minimization of such instances so that traffic rules are followed by everyone is the aim of the project.

1.2 Objectives

The objectives of current implementation are: Detection and categorization of vehicles from a live CCTV video feed Detection of violation of zebra crossing rule at traffic signals and enforcement of traffic rules Detection of lane cutting incidents and enforcement of traffic rules Data analysis done of vehicle density to avoid traffic jams and accumulation of vehicles at certain spots.

1.3 Scope

The scope of current implementation includes an autonomous system consisting of a vehicle classifier, which can distinguish classes of vehicles into two-wheelers, 4-wheelers, and transport vehicles in real time using CCTV cameras mounted at traffic signals. Enforcement and regulation of traffic rules such as lane cutting, zebra crossing violations can be implemented. The system will recognize those vehicles which are breaking traffic rules using deep learning algorithms. Recognition of traffic rule violation will lead to identification of vehicle number using Optical Character Recognition (OCR). After the recognition of these violators of traffic rules via their vehicles license plate numbers, the system will automatically generate a penalty bill for the violator and email it to their accounts. Finally, the data gathered at various traffic signals can be used to train the system to analyze traffic density at various points in an area in order to prevent traffic jams. To summarize, the scope of the project is divided into the following modules: Module 1:- Vehicle Identification Categorization. Module 2:- Enforcement of traffic rules. Module 3:- Data analytics training.

Chapter 2

Literature Review

1. Automatic Number Plate Recognition System for Vehicle Identification Using Optical Character Recognition
Muhammad Tahir Qadri , Muhammad Asif ISBN Num: 978-0-7695-3609-5

ABSTRACT: Automatic Number Plate Recognition (ANPR) is an image processing technology which uses number (license) plate to identify the vehicle. The objective is to design an efficient automatic authorized vehicle identification system by using the vehicle number plate. The system is implemented on the entrance for security control of a highly restricted area like military zones or area around top government offices e.g. Parliament, Supreme Court etc. The developed system first detects the vehicle and then captures the vehicle image. Vehicle number plate region is extracted using the image segmentation in an image. Optical character recognition technique is used for the character recognition. The resulting data is then used to compare with the records on a database so as to come up with the specific information like the vehicles owner, place of registration, address, etc. The system is implemented and simulated in Matlab, and it performance is tested on real image. It is observed from the experiment that the developed system successfully detects and recognize the vehicle number plate on real images.

This model draws from some established systems and develops on them in order to make a completely autonomous system. As explained in this paper, the model presented follows a structure as follows: The solution provided by the authors is divided into 3 major modules: Image Capture, Number plate recognition and extraction, and finally, Character Recognition. Similarly, the modules in this paper follows a similar structure, with key differences occurring in implementation methods, and to a lesser extent, in module structure. Features such as image extraction from camera feed, Image Filtering, and OCR application are used in this model in addition to modules not covered by the referenced paper.

2. You Only Look Once: Unified, Real-Time Object Detection

Joseph Redmon , Santosh Divvala, Ross Girshick , Ali Farhadi University of Washington , Allen Institute for AI , Facebook AI Research

ABSTRACT: We present YOLO, a new approach to object detection. Prior work on object detection repurposes classifiers to perform detection. Instead, we frame object detection as a regression problem to spatially separated bounding boxes and associated class probabilities. A single neural network predicts bounding boxes and class probabilities directly from full images in one evaluation. Since the whole detection pipeline is a single network, it can be optimized end-to-end directly on detection performance. Our unified architecture is extremely fast. Our base YOLO model processes images in real-time at 45 frames per second. A smaller version of the network, Fast YOLO, processes an astounding 155 frames per second while still achieving double the mAP of other real-time detectors. Compared to state-of-the-art detection systems, YOLO makes more localization errors but is less likely to predict false positives on background. Finally, YOLO learns very general representations of objects. It outperforms other detection methods, including DPM and R-CNN, when generalizing from natural images to other domains like artwork.

This model of general object detection is by far the best methods invented yet. This is why we have used this model and trained it using our trained weights to give highest accuracy for our project.

Chapter 3

Technology Stack

The technology stack used for this project can be explained by the module for which it is being used as follows:
Module 1:- Vehicle Identification Categorization:-
Libraries:
Keras, Numpy, OpenCV, YOLOv3 Deep Learning Model
Module 2:- Enforcement of traffic rules:-
1. Optical Character Recognition using Scikit-learn libraries.

Chapter 4

Project Design

4.1 Proposed System

The main aim of our project is to make travelling by the roads of urban areas a much more pleasant experience than it is today. This means that our system will help in strengthening of traffic rules established by the government and make it so that violators of these rules are reliably caught and sentenced the appropriate fine. The key feature of our project is that it is almost entirely automated - meaning that there is little to no human interaction involved when it comes to its usual functioning. This system can ensure that there is a robust framework through which surveillance of vehicles is done and that wrongdoings, when it comes to traffic rules, are punished. By targeting these core problems when it comes to redundancies in the idealistic road commute, we aim to make it so that the traffic rules are followed by everyone, and that there are fewer incidents such as traffic jams, road rage, or crashes.

4.2 Flow of Modules

The entire project is classified into three main modules: 1. Vehicle Identification and Classification, 2. Enforcement of Traffic Rules, and 3. Data Analytics Training. The flow of modules is such that the working of each module depends on the working of the other modules, that is, they are interdependent on each other. For example, the model cannot detect a violation of a traffic rule by a vehicle if it cannot ascertain what the vehicle is in the first place. The first step in the basic flow is to identify objects from a live CCTV video feed and to classify them as vehicle or not vehicle. After that, it has to classify these vehicles into categories based on the class of the vehicle. This module runs all the time. The second module is responsible to determine when a vehicle commits a violation, such as lane cutting, and then use OCR to scan its number plate, following which it can take appropriate action using the details of the vehicle owner. This module is always active in search of rule violations. However, the further process of the flow only begins when such a violation is actually detected. Finally, the third module is also dependent on the first two, in the sense that it works on the data collected by the two in real time. This module ensures that in case of a possible traffic congestion situation, signal duration's can be optimized in order to avoid it.

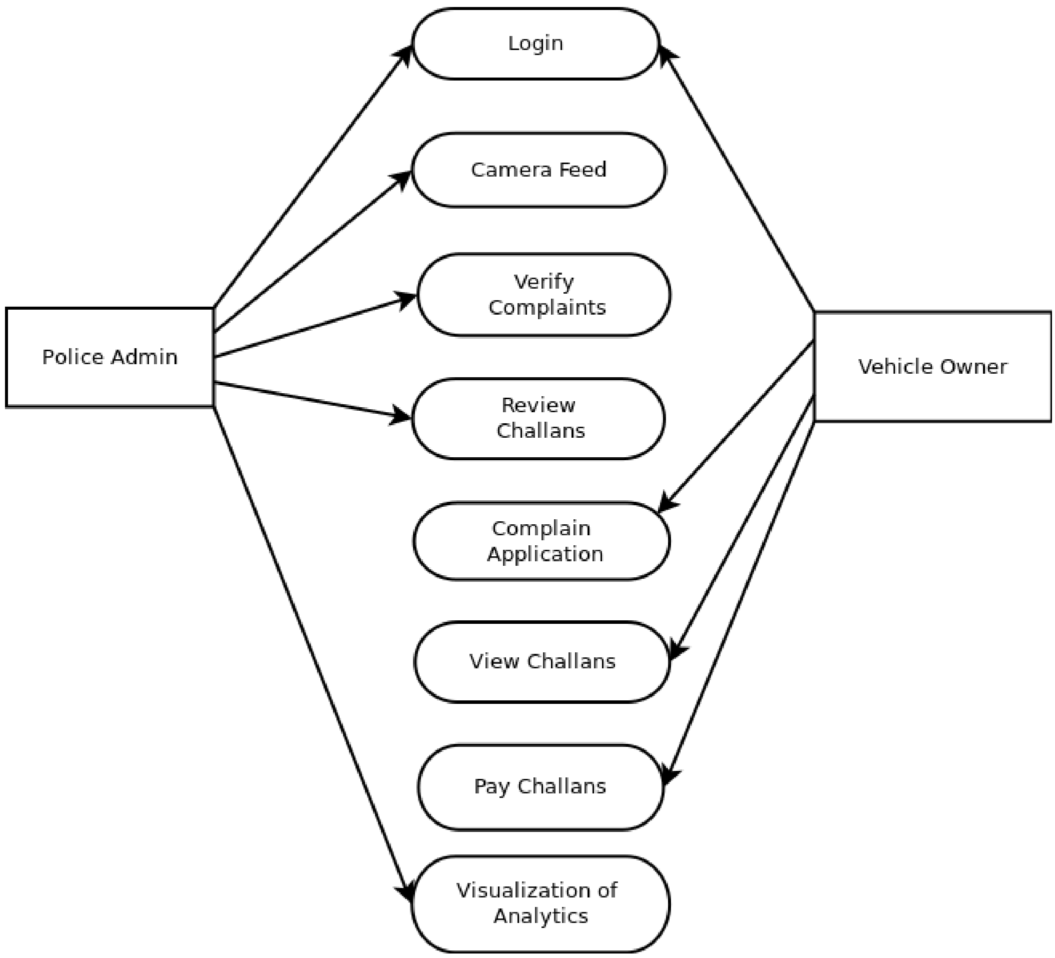


Figure 4.1: Use Case Diagram

4.3 Description of Use Case Diagram

The main working of the system is focused on automation. So, there are few to no human entities which are responsible to carry out any important actions. However, the system does require human personnel to ensure that it is working in a proper manner. The first entity is a traffic police admin, who has a unique login. This grants access to to the live camera feed, review challans, and complaint handling. On the other hand, a user login is given to other people who have obtained challans so that they may pay them, or in case of any complaints or criticisms, fill out an application as a complaint.

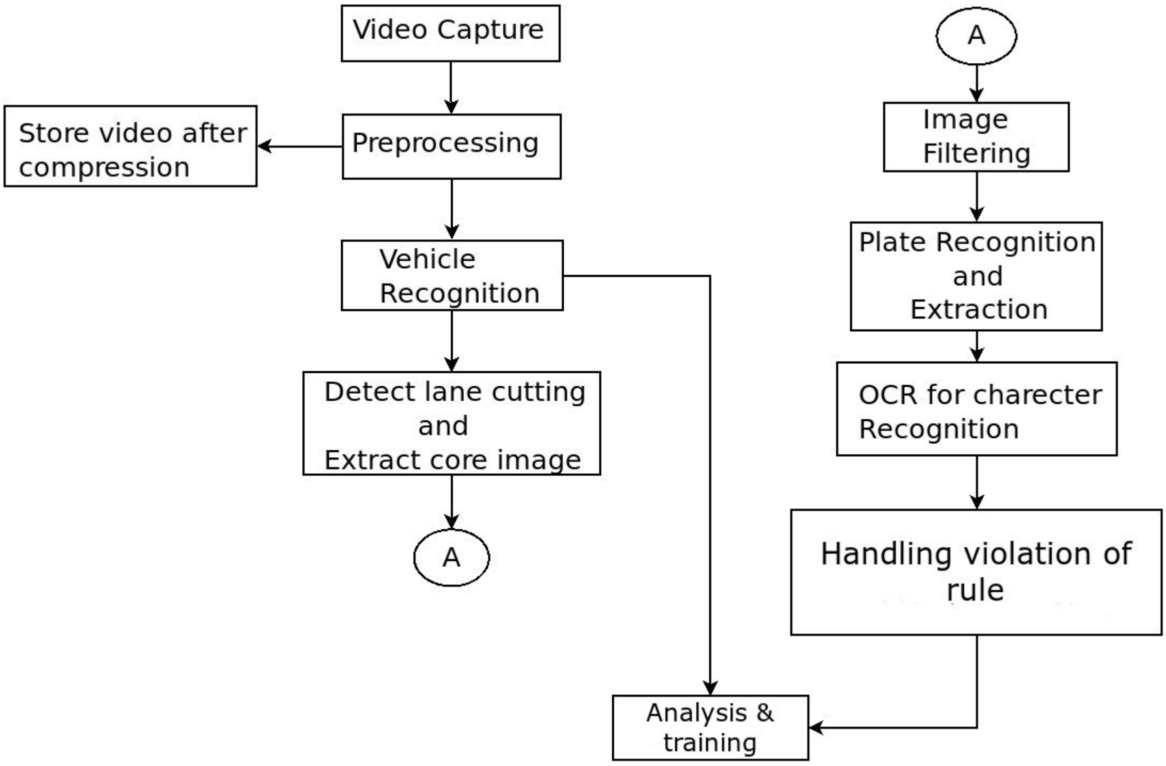


Figure 4.2: Activity Diagram

Chapter 5

CNN Models

Haar Cascade
Haar Cascade is a machine learning object detection algorithm used to identify objects in an image or video and based on the concept of features proposed by Paul Viola and Michael Jones in their paper "Rapid Object Detection using a Boosted Cascade of Simple Features" in 2001
It is a machine learning based approach where a cascade function is trained from a lot of positive and negative images. It is then used to detect objects in other images.

Alexnet
AlexNet was much larger than previous CNNs used for computer vision tasks (e.g. Yann LeCun's LeNet paper in 1998). It has 60 million parameters and 650,000 neurons and took five to six days to train on two GTX 580 3GB GPUs

Yolo
Introduction of YOLOv2 and YOLO9000, real-time detection systems took place recently. YOLOv2 is state-of-the-art and faster than other detection systems across a variety of detection datasets. Furthermore, it can be run at a variety of image sizes to provide a smooth tradeoff between speed and accuracy.

YOLO9000 is a real-time framework for detection more than 9000 object categories by jointly optimizing detection and classification. It uses WordTree to combine data from various sources and our joint optimization technique to train simultaneously on ImageNet and COCO. YOLO9000 is a strong step towards closing the dataset size gap between detection and classification.

Chapter 6

Modules and Implementation

6.1 Module 1: Vehicle Identification and classification

Initially, a video clip is read and segregating into number of frames. Each frame is then considered as an independent image, which is in RGB format and is converted into Gray scale image. In the proposed project, we assume a stationary background for all video sequences. The next phase is identifying the foreground dynamic objects (vehicle), which is obtained by subtracting background image from the given input video frame. The difference between the frames at certain intervals is computed to detect the moving object. The vehicle attributes (width, height, perimeter and area) are obtained by feature extraction technique of image processing. These features are feed into a classifier model to classify the vehicle as big or small by neural network architecture. The total architecture for vehicle classification system used in proposed project. We have used the Convolutional Neural Network Model 'YOLO' to identify vehicles. It is a fast and accurate model which gives a prediction using only a single iteration of the input. We have shown the procedure and the screenshots of the vehicle detections done:
Step 1 The initial step is to collect images upon which we can train our model. This is the dataset. It usually consists of a large collection of input data, which can be text,image,or even audio. However for our task of vehicle recognition, we need image data of vehicles. We have used the UA-Detrac dataset in order to train our model.
UA-DETRAC is a challenging real-world multi-object detection and multi-object tracking benchmark. The dataset consists of 10 hours of videos captured with a Cannon EOS 550D camera at 24 different locations at Beijing and Tianjin in China. The videos are recorded at 25 frames per seconds (fps), with resolution of 960540 pixels. There are more than 140 thousand frames in the UA-DETRAC dataset and 8250 vehicles that are manually annotated, leading to a total of 1.21 million labeled bounding boxes of objects.
Now that we have our dataset, we need to have a neural network model in order to actually train it. There are two options here: build a custom model from scratch and improve upon it; or use one of the pre-existing models which are widely used today. We decided to do the latter as the existing models are more than effective enough to do our task. Some famous models are R-CNN, Faster R-CNN, YOLO, etc. We opted for YOLO.
You Only Look Once (YOLO) is a popular algorithm because it achieves high accuracy whilst also being able to run in real-time. This algorithm only looks once at the image in the sense that it requires only one forward propagation pass through the network to make predictions. After non-max suppression, it then outputs recognized objects together with

the bounding boxes.

Model Details

Inputs and outputs

The input is a batch of images, and each image has the shape (m, 608, 608, 3). The output is a list of bounding boxes along with the recognized classes. Each bounding box is represented by 6 numbers pc, bx, by, bh, bw, c as explained above. If you expand c into an 80-dimensional vector, each bounding box is then represented by 85 numbers.

Anchor Boxes

Anchor boxes are chosen by exploring the training data to choose reasonable height/width ratios that represent the different classes. The dimension for anchor boxes is the second to last dimension in the encoding: m, nH, nW, anchors, classes. The YOLO architecture is: IMAGE (m, 608, 608, 3) -¿ DEEP CNN -¿ ENCODING (m, 19, 19, 5, 85).

Encoding

Lets look in greater detail at what this encoding represents.

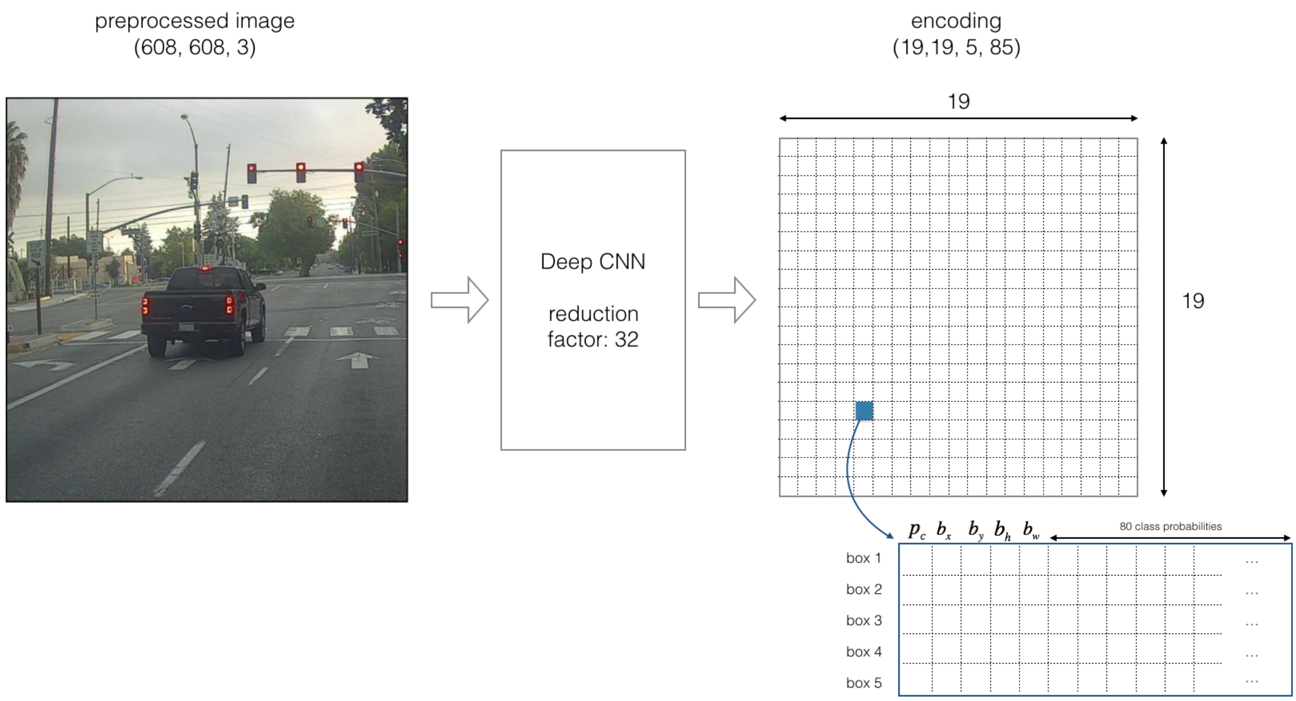


Figure 6.1: Encoding architecture for YOLO

If the center/midpoint of an object falls into a grid cell, that grid cell is responsible for detecting that object.

Since we are using 5 anchor boxes, each of the 19 x 19 cells thus encodes information about 5 boxes. Anchor boxes are defined only by their width and height. For simplicity, we will flatten the last two last dimensions of the shape (19, 19, 5, 85) encoding. So the output of the Deep CNN is (19, 19, 425).

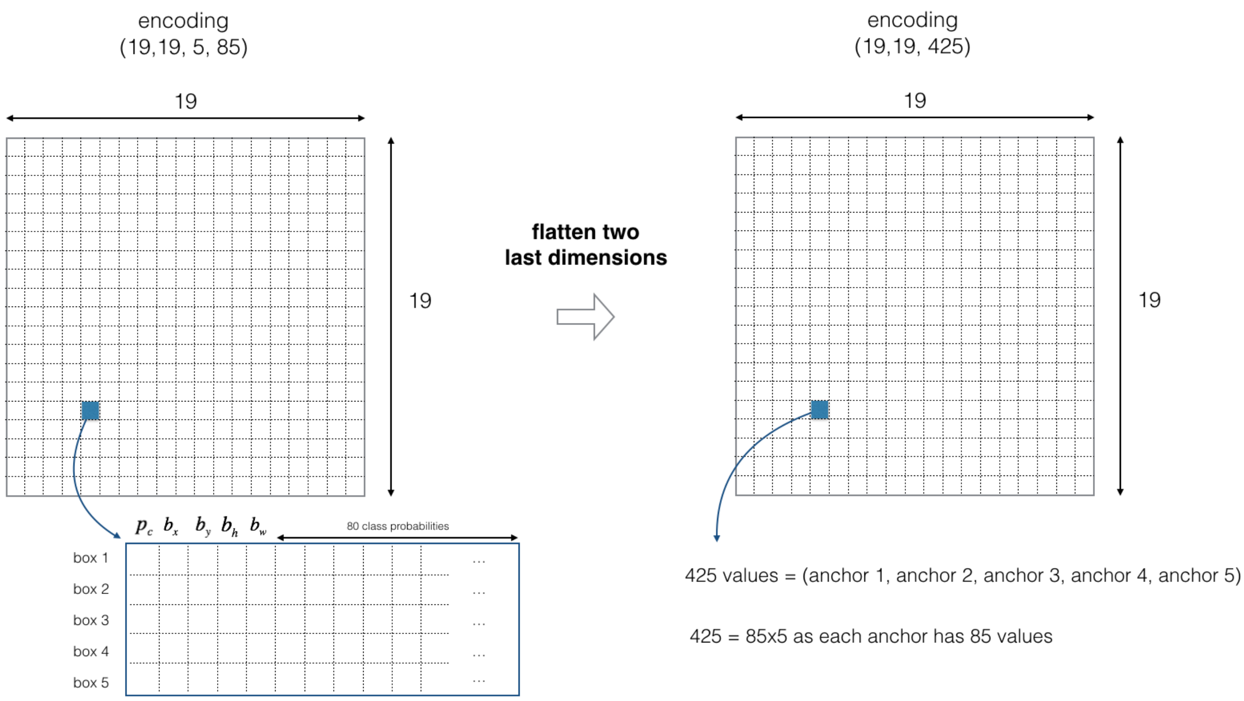


Figure 6.2: Flattening the last two last dimensions

Class score Now, for each box (of each cell), we will compute the following element-wise product and extract a probability that the box contains a certain class.
The class score is score c = p c * c the probability that there is an object p c times the probability that the object is a certain class c.

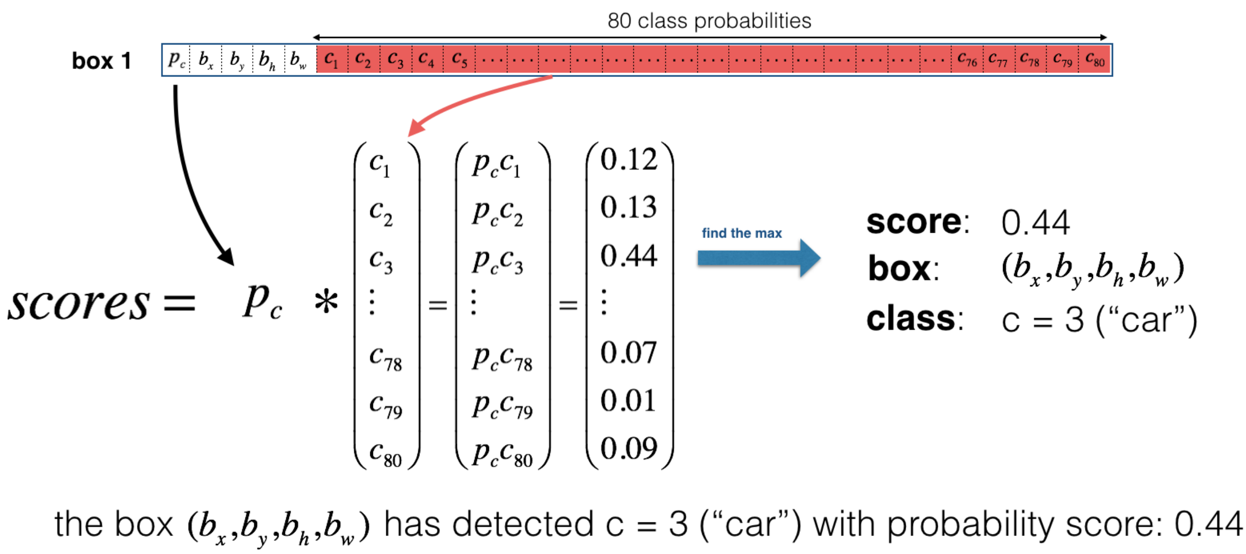


Figure 6.3: Find the class detected by each box

In Figure 6.3, lets say for box 1 (cell 1), the probability that an object exists is p = 0.60. So theres a 60 percent chance that an object exists in box 1 (cell 1).

The probability that the object is the class category 3 (a car) is c = 0.73. The score for box 1 and for category 3 is score c, = 0.60 * 0.73 = 0.44. Lets say we calculate the score for all 80 classes in box 1 and find that the score for the car class (class 3) is the maximum. So well assign the score 0.44 and class 3 to this box 1.

Visualizing classes

Heres one way to visualize what YOLO is predicting on an image:
For each of the 19 x 19 grid cells, find the maximum of the probability scores, that is, taking a max across the 80 classes, one maximum for each of the 5 anchor boxes).
Colour that grid cell according to what object that grid cell considers the most likely.
Doing this results in this picture:

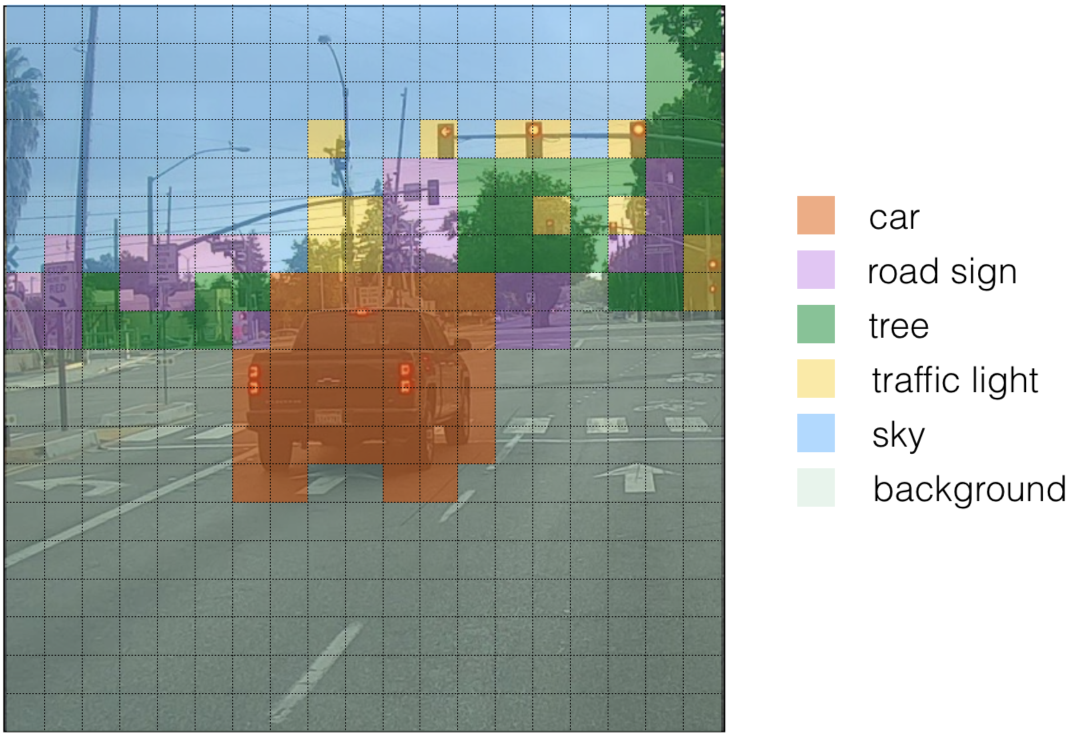


Figure 6.4: Each one of the 19 x 19 grid cells is coloured according to which class has the largest predicted probability in that cell.

The following image is an example of the model we trained using darknet or yolov1, as we can see it accurately shows some cars in the foreground but has some problems for detecting them in the distance.

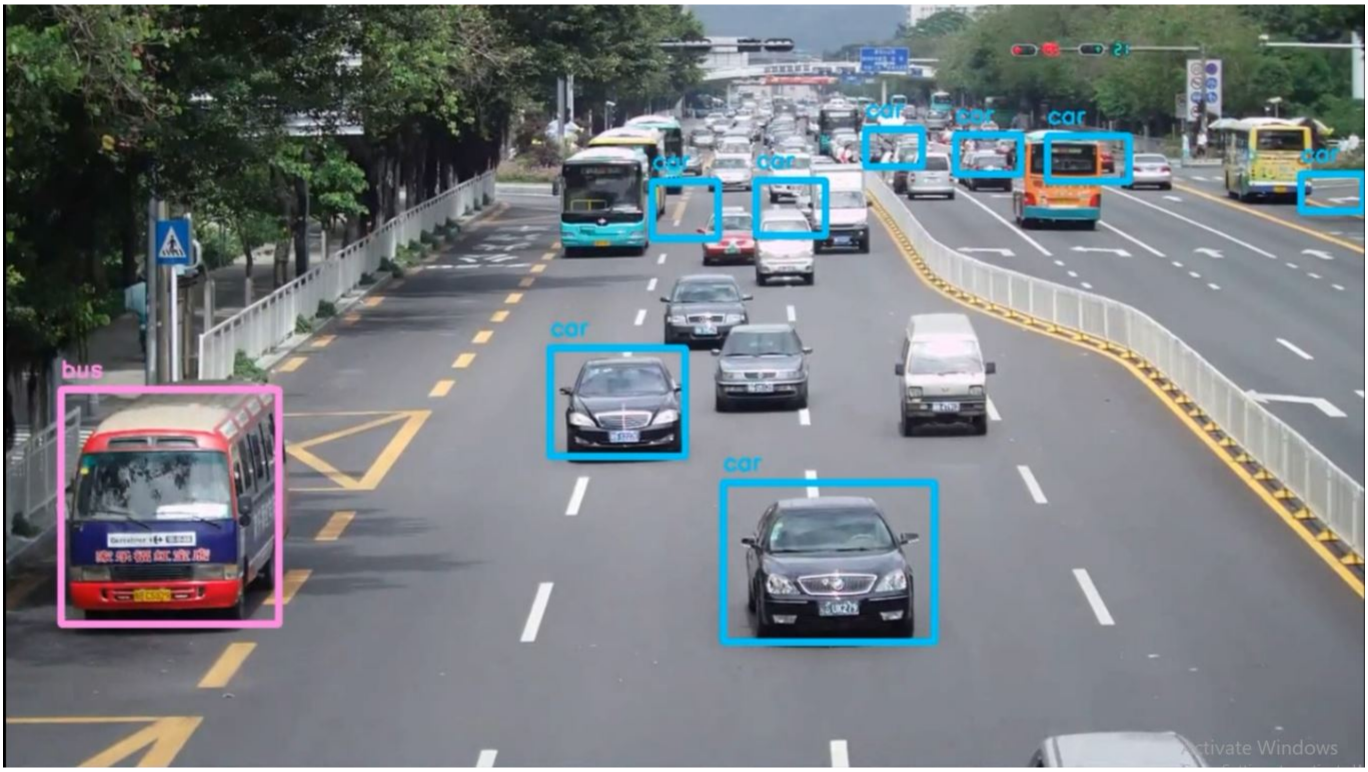


Figure 6.5: Vehicle Detections

6.2 Module 2: Automatic Number Plate Recognition

This module deals with the actual detection and handling of traffic rule violations. This is done using Optical Character Recognition using OpenCV, which identifies rule violators and detects vehicle and owner information using its number plate. With this information, appropriate action can be taken. Current scope includes violations such as crossing limit during red signal.
Here is the process whichs show how the number plate detection works

Step 1
This is the first stage and at the end of this stage, we should be able to identify the license plates position on the car. In order to do this, we need to read the image and convert it to grayscale. In a grayscale image, each pixel is between 0 255. We now need to convert it to a binary image in which a pixel is either complete black or white.

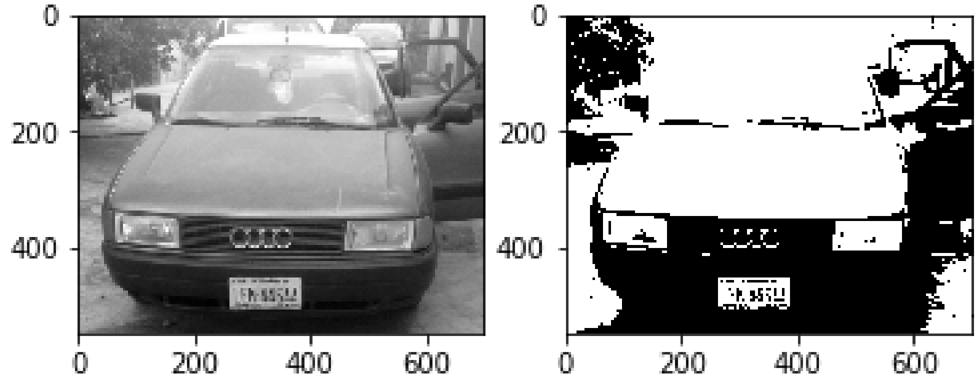


Figure 6.6: Image in Grayscale

Step 2
We need to identify all the connected regions in the image, using the concept of connected component analysis (CCA). Other approaches like edge detection and morphological processing can also be explored. CCA basically helps us group and label connected regions on the foreground. A pixel is deemed to be connected to another if they both have the same value and are adjacent to each other.
The measure.label method was used to map all the connected regions in the binary image and label them. Calling the regionprops method on the labelled image will return a list of all the regions as well as their properties like area, bounding box, label etc. We used the patches.Rectangle method to draw a rectangle over all the mapped regions.

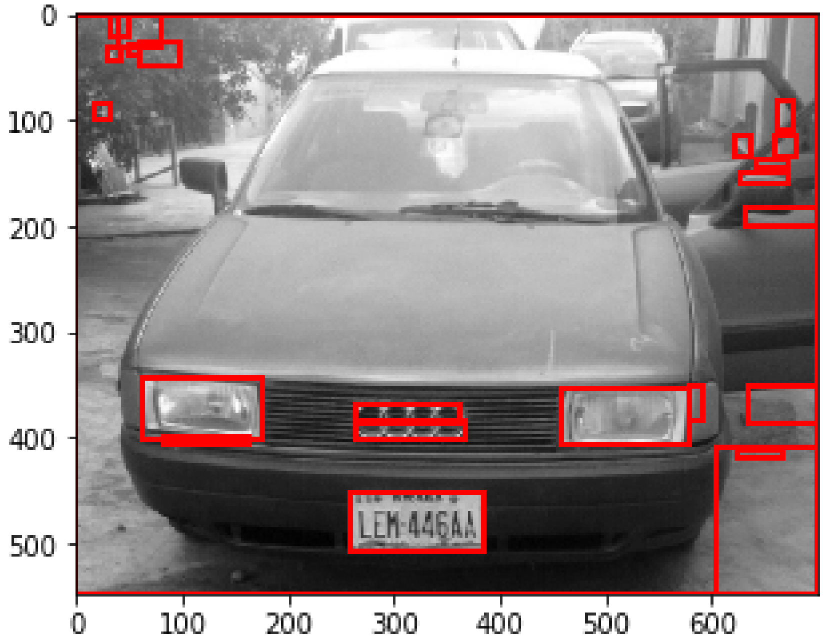


Figure 6.7: Detections for potential numberplates

Step 3

From the resulting image, we can see that other regions that do not contain the license plate are also mapped. In order to eliminate these, we will use some characteristics of a typical license plate to remove them.
They are rectangular in shape.
The width is more than the height.
The proportion of the width of the license plate region to the full image ranges between 15 and 40 percent.
The proportion of the height of the license plate region to the full image is between 8 20 percent.

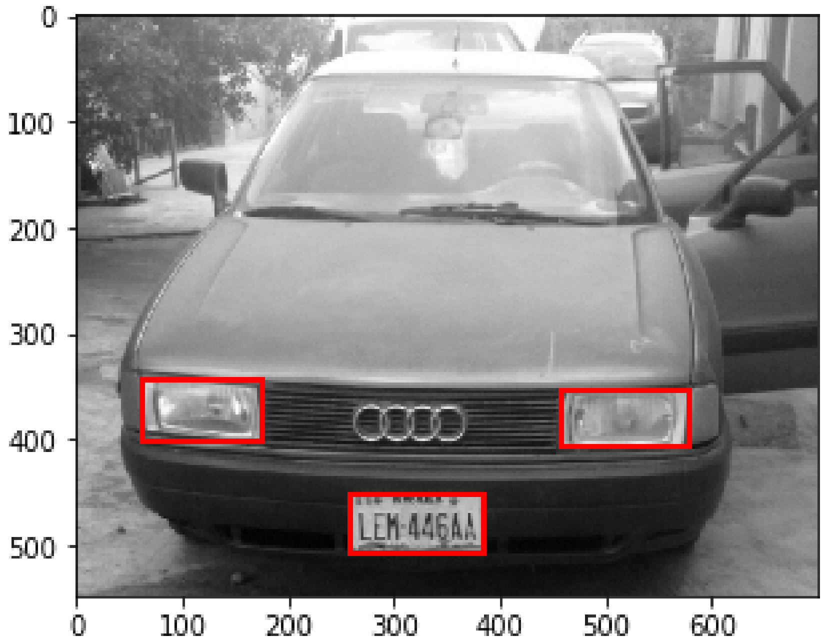


Figure 6.8: Filtered Detections

Step 4

The plate like objects is a list of all the regions on the car that look like a license plate. Then a CCA was done on the license plate and each character was resized to 20px by 20px. This was done because of the next stage that has to do with recognition of the character. In order to keep track of the order of the characters, the columnlist variable was introduced to take note of the starting x-axis of each region. This can then be sorted to know the correct order of the characters.

This is going to be the last stage, its at this stage we introduce the concept of machine learning. Machine learning can simply be defined as the branch of AI that deals with data and processes it to discover pattern that can be used for future predictions.

There are several classifiers we can use with each of them having its advantages and disadvantages. Well use SVC (support vector classifiers) for this task. We chose to use SVC because it gave me the best performance. However, this does not necessarily mean that SVC is the best classifier.

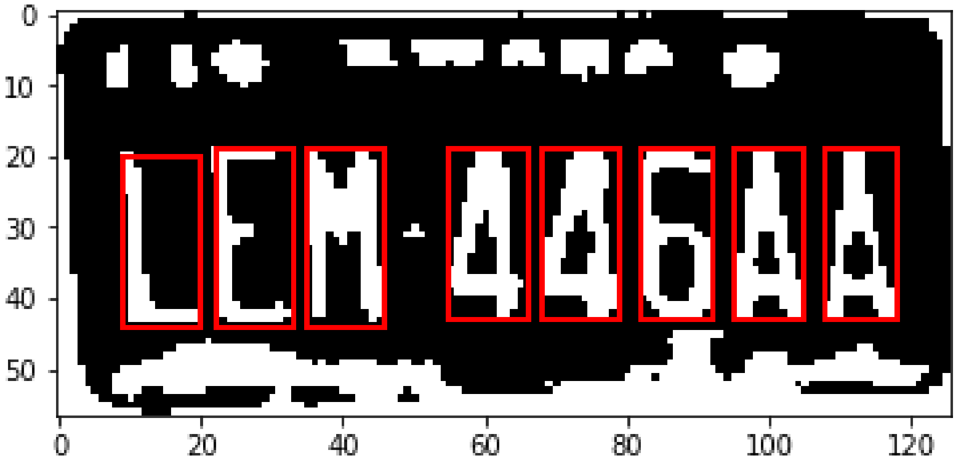


Figure 6.9: Character Recognition after preprocessing

Chapter 7

Benefits and Applications

7.1 Benefits

Smoke emission done by vehicles greatly harms the environment and makes up for 94.5% of the carbon footprint in the years 2003 - 2004 and has increased since then. With this system, it is possible to reduce occurrence of traffic jams by punishing those who cause them, and by extension reduce the time individual vehicles spend on roads.

Incidents of traffic violation are often mixed with those of road rage and traffic situations. A simple lane cutting incident can slow down an entire lane and subsequently, the entire commute. Reducing these incidents is one of the main goals of the project. This will hold liable only those people who have done the traffic violation, and not the bystanders who have nothing to do with it. Additionally, this can reduce occurrences of traffic jams which will save people a lot of time.

7.2 Applications

Classifying vehicles according to their type is a basic step in most projects which involve vehicular traffic. Achieving high accuracy in this is very important, and in doing so it can also be used for other projects based on it, such as centralized traffic regulation in an area. In India, some aspects of this project are being implemented in a partial way. That is, vehicle recognition for traffic violators may be done in an automated fashion, but the action to be taken is done manually and so is prone to human error, such as inability to actually charge a few vehicles. However, this project aims to remove such vulnerabilities in the system. Some classes of vehicles have restricted allowance in terms of transport, such as huge carrier vehicles which are only allowed to commute in the night as they are slow and bulky and extremely prone to cause traffic jams if allowed to travel during the day. In case this does happen, the system can catch those vehicles and penalize them accordingly. On rare occurrences such as criminal activity, training of the model can be done to keep track of a singular suspect vehicle. While this is not in the immediate scope, it may be developed at a later stage.

Chapter 8

Conclusions and Future Scope

To conclude, the main implementation of the project consists of reporting of traffic rule violators using deep learning convolutional neural networks in order to train an autonomous system to carry out traffic law enforcement. Future scope includes tracking of specific vehicles, such as those which are suspect. A centralized system for traffic regulation can be implemented using the live data from the CCTV Cameras. So, dynamic assignment of traffic signal durations can be done.

Bibliography

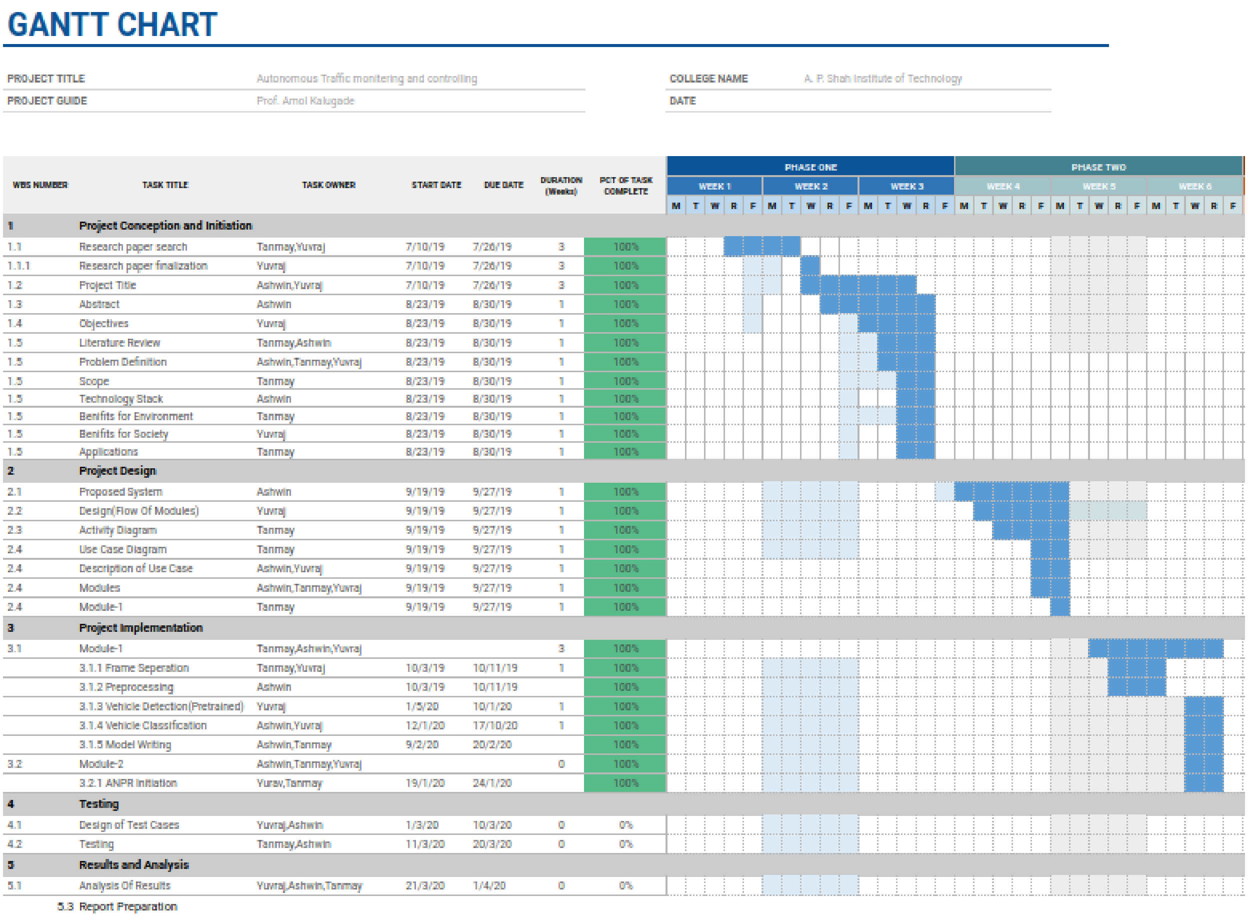
[1] Joseph Redmon, Ali Farhadi,University of Washington, Allen Institute for AI - "YOLO9000: Better, Faster, Stronger"

[2] Muhammad Tahir Qadri, Muhammad Asif, 2009 International Conference on Education Technology and Computer - "Automatic Number Plate Recognition System for Vehicle Identification Using Optical Character Recognition"

[3] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi,University of Washington, Allen Institute for AI, Facebook AI Research- "You Only Look Once:Unified, Real-Time Object Detection"

Annexure

Annexure-A: Gantt Chart



GANTT CHART

PROJECT TITLE	Autonomous Traffic monitering and controlling	COLLEGE NAME	A. P. Shah Institute of Technology
PROJECT GUIDE	Prof. Amol Kalugade	DATE	

WBS NUMBER	TASK TITLE	TASK OWNER	START DATE	DUE DATE	DURATION (Weeks)	PCT OF TASK COMPLETE
1	**Project Conception and Initiation**					
1.1	Research paper search	Tanmay,Yuvraj	7/10/19	7/26/19	3	100%
1.1.1	Research paper finalization	Yuvraj	7/10/19	7/26/19	3	100%
1.2	Project Title	Ashwin,Yuvraj	7/10/19	7/26/19	3	100%
1.3	Abstract	Ashwin	8/23/19	8/30/19	1	100%
1.4	Objectives	Yuvraj	8/23/19	8/30/19	1	100%
1.5	Literature Review	Tanmay,Ashwin	8/23/19	8/30/19	1	100%
1.5	Problem Definition	Ashwin,Tanmay,Yuvraj	8/23/19	8/30/19	1	100%
1.5	Scope	Tanmay	8/23/19	8/30/19	1	100%
1.5	Technology Stack	Ashwin	8/23/19	8/30/19	1	100%
1.5	Benfits for Environment	Tanmay	8/23/19	8/30/19	1	100%
1.5	Benfits for Society	Yuvraj	8/23/19	8/30/19	1	100%
1.5	Applications	Tanmay	8/23/19	8/30/19	1	100%
2	**Project Design**					
2.1	Proposed System	Ashwin	9/19/19	9/27/19	1	100%
2.2	Design(Flow Of Modules)	Yuvraj	9/19/19	9/27/19	1	100%
2.3	Activity Diagram	Tanmay	9/19/19	9/27/19	1	100%
2.4	Use Case Diagram	Tanmay	9/19/19	9/27/19	1	100%
2.4	Description of Use Case	Ashwin,Yuvraj	9/19/19	9/27/19	1	100%
2.4	Modules	Ashwin,Tanmay,Yuvraj	9/19/19	9/27/19	1	100%
2.4	Module-1	Tanmay	9/19/19	9/27/19	1	100%
3	**Project Implementation**					
3.1	Module-1	Tanmay,Ashwin,Yuvraj			3	100%
	3.1.1 Frame Seperation	Tanmay,Yuvraj	10/3/19	10/11/19	1	100%
	3.1.2 Preprocessing	Ashwin	10/3/19	10/11/19		100%
	3.1.3 Vehicle Detection(Pretrained)	Yuvraj	1/5/20	10/1/20	1	100%
	3.1.4 Vehicle Classification	Ashwin,Yuvraj	12/1/20	17/10/20	1	100%
	3.1.5 Model Writing	Ashwin,Tanmay	9/2/20	20/2/20		100%
3.2	Module-2	Ashwin,Tanmay,Yuvraj			0	100%
	3.2.1 ANPR Initiation	Yurav,Tanmay	19/1/20	24/1/20		100%
4	**Testing**					
4.1	Design of Test Cases	Yuvraj,Ashwin	1/3/20	10/3/20	0	0%
4.2	Testing	Tanmay,Ashwin	11/3/20	20/3/20	0	0%
5	**Results and Analysis**					
5.1	Analysis Of Results	Yuvraj,Ashwin,Tanmay	21/3/20	1/4/20	0	0%

5.3 Report Preparation

Figure 8.1: Gantt Chart

Acknowledgement

We have great pleasure in presenting the report on **Autonomous Traffic Monitoring and Controlling.** We take this opportunity to express our sincere thanks towards our guide **Prof. Amol Kalugade** Department of Computer, APSIT Thane for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of project.

We thank **Prof. Sachin H. Malave** Head of Department,Computer, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

We thank **Prof. Amol Kalugade** BE project co-ordinator, Department of Computer, APSIT for being encouraging throughout the course and for guidance.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

Yuvraj Yadav:
16102040:

Ashwin Shenolikar:
16102037:

Tanmay Sule:
16102032: