Road Traffic Prioritization using Image Processing

Brighton C. Berejena[1], **Fungayi D. Mukoko**[2]

School of Information Science And Technology, Harare Institute of Technology, Harare, Zimbabwe

Abstract: ***One of the problems that has affected most cities in recent years is the massive increase in number of vehicles, which leads to traffic congestion. Such a problem puts huge stress on various systems for decision making and infrastructure planning of many metropolitan areas. On top of this congestion complexity, there are motor vehicles that should be given first priority when approaching an intersection. These vehicles include the ambulances, VIP motorcades, fire-fighter trucks, police vans etc. Therefore, traffic control becomes the main concern for different systems to overcome this challenge as associated phenomena of the continuous growth of population and vehicles over inadequate road infrastructures and the need to give first preferences to certain types of motor vehicles. In this research, I assume each traffic signal alongside the road has multiple high resolution cameras to capture images of vehicles that are approaching and send the real-time images to a cloud-based High Processing Computer (H. P. C.) for image classification and providing a rating score for each of the intersecting roads.***

Keywords: Intelligent traffic light, Machine Learning, image detection, Motor Vehicle Classification, Traffic Management

1. Introduction

The increase in population of motor vehicles in the Metropolitan areas around the world has created a new set of problems. One of the major problems is traffic congestion which directly affects service delivery and free movement of traffic. Previous researches focused on detecting the best routes and advising the motorists on the best route. Other researches also focused on reducing traffic congestion by giving priority to roads that have a huge built up of traffic. These previous research do not give consideration to the fact that there are certain classes of vehicles that should be given first priority when approaching an intersection, thus most cities have to use the police to control traffic.

1.1. Background to the problem/study

A city is a complex system which consists of many interdependent subsystems where traffic system is one of its important subsystems. A study says; it is the cornerstone of the world's economy. Moreover, it is also declared as one of the major dimensions of the smart city. With the rapid growth of the population of the world, the number of vehicles on roadways is increasing consequently, the rate of traffic jams is also increasing in the same manner. These traffic jams can and are now seriously affecting the effective movement of emergency vehicles thereby affecting their service delivery and responses to emergency.

It is, therefore, identified that active traffic management is now very vital. In majority countries, traffic is managed through fixed time signals whereas, in large cities of some developed countries, traffic is managed through centrally controlled systems. The paradigm of the Machine Learning (ML) has been introduced in traffic management systems.

1.2. Statement of the problem

To the best of my knowledge, it is identified that till date the current traffic management systems are centralized. In case of networking issues, such systems may crash. In addition, there is less focus on fluctuations in traffic prioritization. Current systems do not give priority to different classes of vehicles. For example, emergency vehicles, police vehicles and Presidential motorcades are not given any due prioritization by the current systems. Therefore, the proposed system classifies vehicles based on the type of the vehicle and give a road on the intersection its weighting based on the different classes of vehicles approaching the intersection from that particular road. The system exploiting the concepts of Machine Learning and Artificial Intelligence together. The representation of traffic data in statistical form can also be helpful to authorities for real-time controlling and managing traffic. Moreover, it may also be helpful for future planning.

1.3. Research objectives

The long-term goal of this research is to develop a system that can control traffic lights by using an image processing algorithm by giving rankings of the different roads at an intersection at specific time intervals.

The aims and objectives of this research are to come up with a solution that enables unattended control of traffic lights within the central business districts and other busy road intersections. Particularly the research has the following sub-aims and objectives:

1) OB1 - To collect traffic images using high-resolution cameras and submit them to a data center equipped with high-performance computers for processing.
2) OB2-To analyze images on monitored road intersections using the Deep Convolutional Neural Network (DCNN) algorithm such as TensorFlow, OpenCV and Deep Java Library (DJL).
3) OB3-To produce a control signal that can be sent to the traffic lights.

1.4. Proposed Solution and Justification

The research is expected to produce a solution that enables classification of road traffic, and provide and ranking mechanism for each of the roads at a monitored intersection. The ranking will be used to determine a signal code that can be send to traffic lights at the intersection which can either be Green, Amber or Red.

The proposed system, shown in Figure 2, is designed to govern traffic at road networks, sensing through surveillance cameras which are embedded on roadsides. The system works in a distributed manner, it processes cameras' video and image data at the local server, determines the type of vehicles approaching the intersection and sends a control signal to the traffic lights.
In addition to this, it also tackles emergency vehicles such as ambulance, fire brigade.

The system is divided into three layers:
a) Data Acquisition and Collection layer.
b) Data Processing and Decision-making layer
c) Application and Actuation layer

A) Data Acquisition and Collection Layer

A surveillance camera is the most widely used source to detect the road traffic in this field due to efficiency and ease of maintenance [15] [16] [17] [18]. Blob detection algorithm is applied to the video stream at the local server due to its performance and capability of noise reduction [18]. After traffic detection, a local server sends the density measured through image processing to the respective micro-controller. Image acquisition refers to the process of acquiring the image from a suitable source. In this project, the source is a camera (preferably CCTV, to avoid additional setup). The image is obtained by converting a video into multiple frames and processing each frame at a time. The image resolution can vary from 720*480p to 1280*720p based on the camera used.

Figure 1a: Vehicle detection

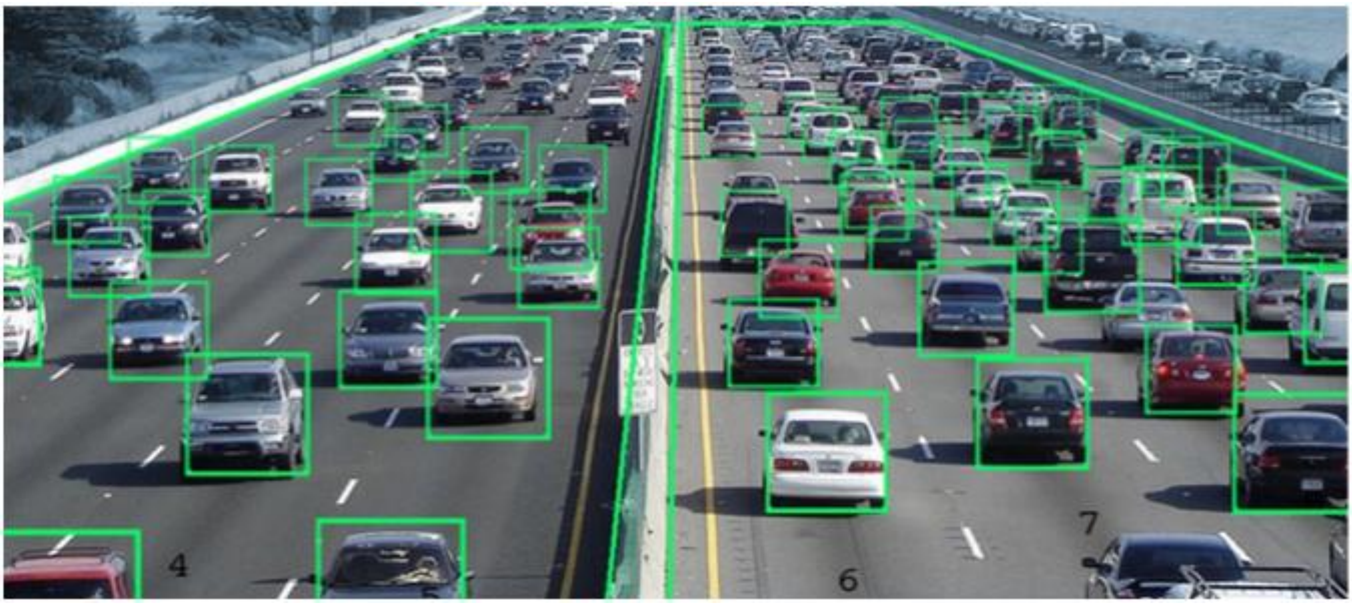


Figure 1b: Vehicle Detection Using JavaDL and Opencv

Figure 1: Below illustrates the vehicle detection process as the vehicles will be approaching the traffic intersection

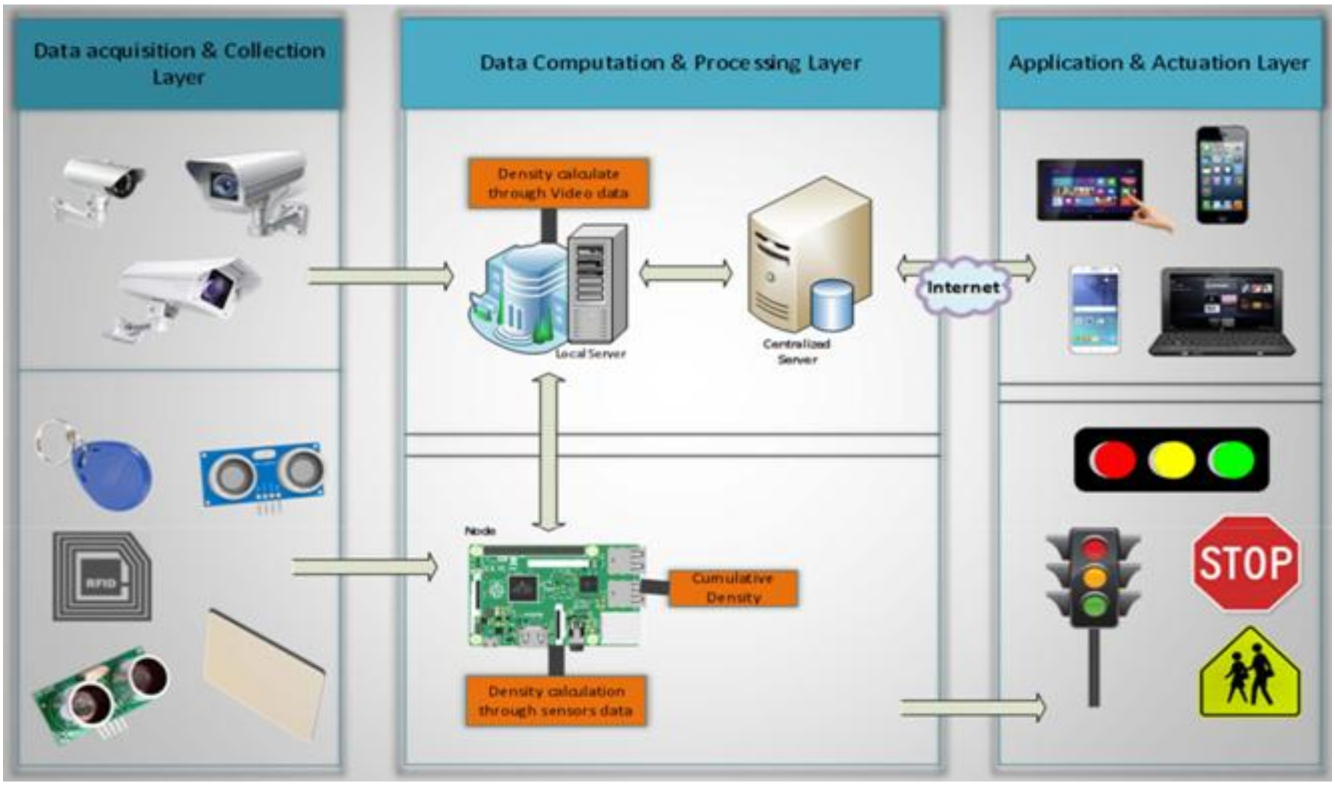


Figure 2: The System Model

B) Data Processing and Decision-making layer

Data processing and decision making layer involves identifying the type of traffic from the collected images and videos, giving a ranking on each identified vehicle and giving a score on each road. This is all done on a real-time basis. Figure 3 below shows the different types of vehicles that will be monitored and their corresponding scores.

The acquired image is scaled down from the original resolution into 400*300p. This increases the frame rate of the image acquisition and hence helps the system avoid latency issues. Image scaling brings a level of consistency to the images acquired by converting input from all camera sources into a single resolution. It reduces the number of edges detected by ignoring minute details. Image resizing is primarily required when the system's processor cannot handle a large density of pixels at high speeds. Based on the algorithm that is used, image scaling can produce highly differentiated results as compared to a non-resized image.
new height = Previous height/Previous width*new width
new width = Previous width/Previous height*new height

A grayscale image is one where each pixel depicts only its intensity. Hence the colors of each pixel are restricted to shades of gray ranging from white to black. The image is converted from RGB to grayscale during the acquisition process. Grayscaling refers to the process of converting an image from RGB into black and white. Multiple functions in OpenCV can work only if a grayscale image is passed as a parameter. Canny Edge detection also uses grayscale images. The edge detection algorithm returns white for all detected edges and black for the rest of the image. OpenCV uses the following formula to convert a pixel from RGB to gray-scale:

Y□ 0299R+0.587G+0.114B

R, G, B denote the values of the red, green and blue colors of a single pixel in an image.

Canny Edge Detection was chosen since it has optimal quality, is Simple and Single edge point response.
The Canny edge detection algorithm follows a set of basic steps:
Step 1: input image with a Gaussian filter to remove noise and unwanted details and texture.
Step 2: Calculate the gradient magnitude and angle images.
Step 3: A non-maxima based suppression is applied to the gradient magnitude image.
Step 4: Hysteresis.

Template matching is a technique in image processing which is used to find the regions of an image that match (are similar) a template image (patch). It is a brute force method where we compare two images pixel by pixel along the height and width of an image. Steps involved in this are:
Step 1. Obtain source and template image.
Step 2. Sliding: Move the template image pixel by pixel in both x and y directions.
Step 3. Calculate a metric at each location
Step 4. Using this metric, deduce how good or a bad of a match is formed between the images.
Step 5. The brightest location indicates the best matches.

Java and OpenCV algorithm was used to detect vehicles from the images. Java is a high-level Object Orientated Programming Language used for general-purpose programming. By making use of the existing Deep Java Library, and improving on its algorithm on decision-making, I was able to classify the images of the traffic.

	Vehicle Type	Score
	Ambulance (Red Beacon)	50
	Fire Truck (Red Beacon)	50
	Police Car (Blue Beacon)	40
	Motorcade (Blud Beacon)	30
	Bus	20
	Haulage Truck	10
	Taxi	5
	Private Car	3

Figure 2: Vehicle Priority Table

The table above is only for illustration purposes. The ranking are arbitrary values only. If the emergency vehicle is detected, the system stops its normal operation and immediately turns the respective signal green and it remains green until that particular vehicle passes to that intersection. In addition to this, if the fire is detected on the road, the micro-controller intimates to the respective local server through which it goes to the centralized server and then this information goes to the respective department through a mobile application. The flow, how the system calculates signal time, is presented in Figure 3.

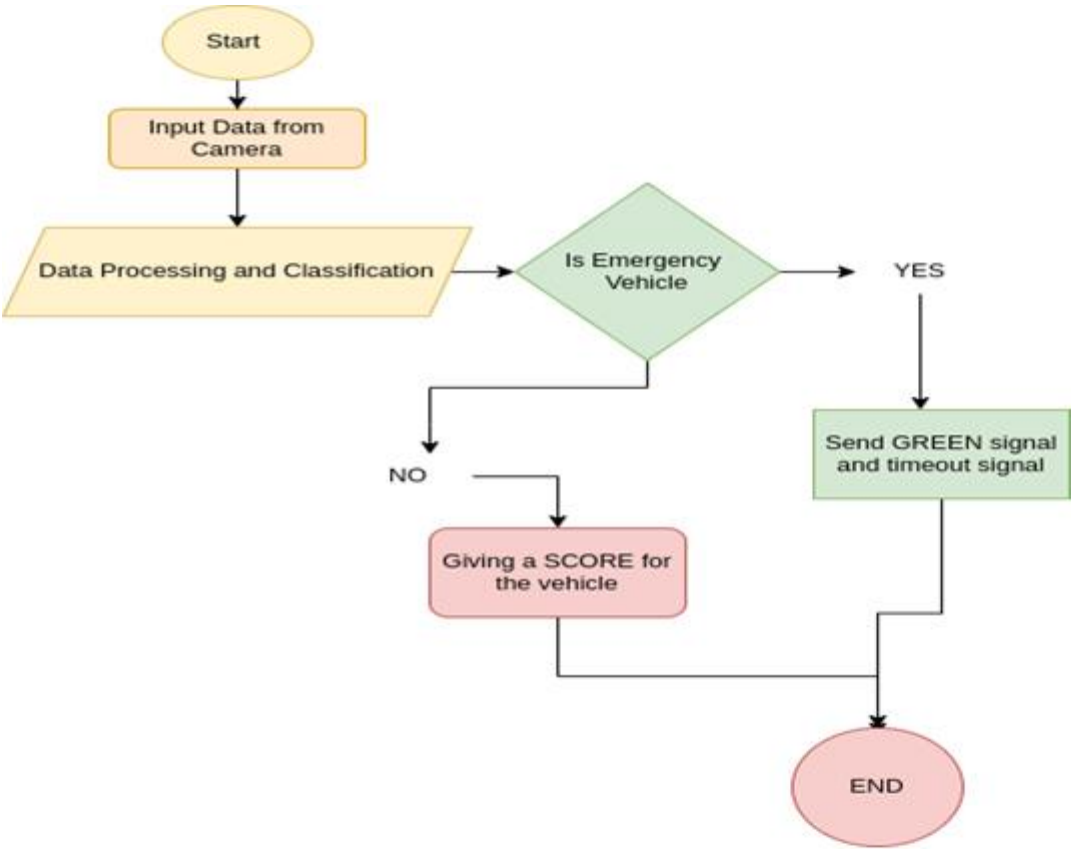


Figure 3: Signal Calculation Process

C) Application and Actuation Layer

In this layer, there are two types of information delivered: i) duration of a green signal from node to traffic signal and ii) daily, weekly, monthly and yearly reports to the administration of smart traffic management system through the web application from a centralized server. First of all, the system calculates rush interval by using Regression Tree algorithm on the data saved at the local server and updates this report to the centralized server on the daily basis (after 24 hours). The rush interval is the time span of thirty minutes. This report is then displayed on the web application which is linked to a centralized server which is for the administration of smart traffic management system, that shows daily, weekly, monthly and yearly graphs of rush intervals for roads. This graphical information is fruitful for the future road planning and resource management.

Secondly in the actuation module, whenever the rush interval is identified, the local server intimates to the respective micro-controller along with the road id. After receiving the rush interval intimation, the decision-making module updates the duration of the green signal to the respective traffic signal. In this modern era, where time is money and wastage of time are not affordable, there is a need to know the traffic condition on the particular road prior to travel on that road by using mobile application. Moreover, this system is also capable of managing emergency situations like if the smoke and fire are detected on the road. In case of fire on the road, which is detected by flame sensors and extensive smoke through smoke sensors, the system intimates to the nearby relevant department through a mobile application for further actions.

Anticipated Results

A prototype was developed to demonstrate the applicability of the proposed system. Several experiments on real traffic data were carried out to evaluate the efficiency of the proposed algorithm. The vehicle types were monitored and classified in Figure 4. As soon as the traffic score crosses the specified threshold on a road, the system stopped the normal operation and kept the green light on till the situation on the road became normal. The real-time data was also being sent to the local and central server as well datai.e. number of vehicles passed in a particular time span at a particular road.

The bar graph is representing real-time traffic data. Different bar graphs based on historical and real-time data are being drawn in this application which is helpful for traffic department and other related authorities for i) managing traffic prioritization on roads ii) and future planning.

1.5. Research justification/Significance of the study

The main rationale behind this project proposal is to come up with a traffic images processing algorithm that can be used to control traffic lights based on the congestion levels and well as the priority of the motor vehicles that are approaching the intersection. This will ensure that emergency vehicles and other VIP vehicles are not seriously delayed at traffic intersections

1.7 Scope of the Project

1) The scope of the project involves the following activities:
2) Designing an algorithm that determines the type of a picture and allocates an image classification/rating.
3) Designing an algorithm that allocates a weight to every image it receives based on the type/class it belongs to.
4) Generating a weight for the road based on the sum of the images collected from a particular road. If a weight reaches a certain threshold, a correct traffic signal is generated and submitted to the road intersection.
5) The project does not cover the following areas:

Image collection. The assumption will be that there are already existing high-resolution cameras which can be installed on road intersections to collect traffic images. These images will be submitted to a remote H. P. C computer for storage and real-time processing.

1.8 Definition of terms and abbreviations

The following terms and abbreviations were used in the thesis:
1) HPC - High Processing Computer
2) RGB - Red Green Blue
3) ML - Machine Learning
4) AI-Artificial Intelligence

2. Literature Review

After thorough reading and researching various approaches to solve problems are:

An Image Processing Based Approach for Real-Time Road Traffic Applications Sensors at their current state are not enough to provide necessary traffic information to engineers and their setup can create a disturbance in the ecosystem. Hence morphological edge detection is one of the best possible approaches to analyze traffic. This system then uses a background neural network to judge traffic parameters. Neural networks are faster and more accurate as compared to other systems when it comes to analyzing traffic.

An Automated Vehicle Counting System for Traffic Surveillance. This system uses a software comprising of two components: An embedded digital signal processing (DSP) software and a host personal computer (PC) software. The embedded DSP software is used to acquire a video image from a camera, counts the number of cars and transmits the results and real-time images to the PC software. The PC software works as a graphical user interface (GUI) through which the end user can access the image processing results. The algorithm used is designed to maintain efficient counting of vehicles over inconsistent illumination levels.

A smart traffic management system that is partially deployed in Cambridge city where queue detectors are buried in the roads that detect the traffic queue and inform the central control unit which takes decision accordingly. Since the system is centralized that can slow down due to networking issues [9]. The researcher used surveillance cameras to detect traffic and OCR to identify the vehicles through number plate recognition which is a simple detection method but the system will fail in Zimbabwe as there are different kinds of traffic including cycles which have no number plate [10].

Osman et al. proposed a system in which they have used surveillance cameras to detect traffic density using MATLAB, a traffic controller and a wireless transmitter used to send images to the server after that server calculated traffic density by using those images of every section. This system used fixed (predefined) thresholds that depend on a number of vehicles on road. An algorithm was used to set a time span of red light for a particular lane of the intersection, which is determined by traffic density on road and forwarded to the microcontroller and then server [11].

Jadhav et al. used surveillance cameras, MATLAB and KEIL (Microcontroller coding) to control traffic congestion. This paper also discusses the priority-based traffic clearance and red signal broker (Number plate detection). Due to using heavy hardware, it is difficult to manage and become costly [8]. Bui et al. Analyzed a real-time process synchronization based system to manage the traffic flow dynamically. Sensors were used to detect the traffic, where vehicle to vehicle and vehicle to infrastructure communication was done by using wireless communication devices. Controller placed at the center of the intersection received vehicles' and pedestrians' information and requests and process using first come first serve method [12].

Swathi et al. proposed smart traffic routing system that chooses the shortest route having the least congestion. Sensors are used to collect data about traffic density, these sensors use solar energy and battery. Sensors kept transmitting infrared light and when an object came near, they detect traffic density by monitoring the reflected light from the vehicle. However, readings may change with the change in temperature and humidity [13].

Al-Sakran et al. proposed a system in which major goals were detecting vehicles and get their location by using sensors and RFIDs after getting data it sent to centralized controlling center by using a wireless connection for further processing. Researchers used cloud computing, RFIDs, GPS, wireless sensor network (WSN), agent and other modern tools and technologies to collect, store, manage and supervise traffic information [14].

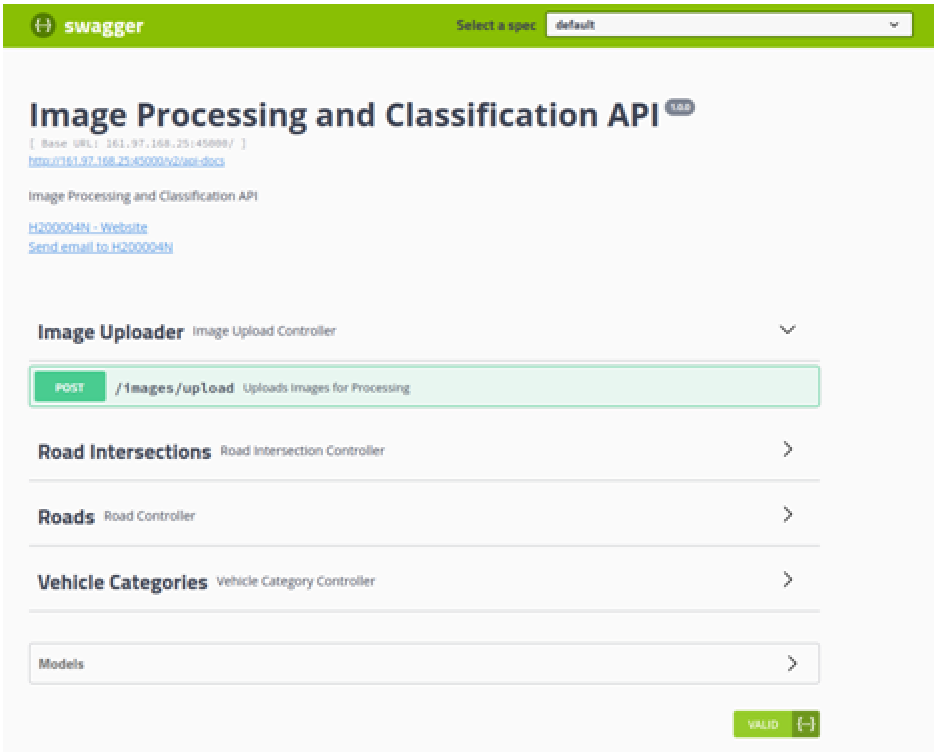
3. Research Methodology

The following methodologies will be followed in achieving the objectives above:

1) Data Collection using high resolution cameras.
2) Data identification using the DCNN algorithm such as OpenCV. and the Java Deep Learning algorithm.
3) Data analysis.
4) Produce the results.

4. Discussion of Results

The project prototype was developed and deployed in a cloud based server. It can be accessed via an API call on this address: http://161.97.168.25: 45000/swagger-ui. html#/



5. Summary, Conclusions and Recommendations

Cloud computing concepts can be used together with Machine Learning to improve on decision making and process automation. Human error can be eliminated from repetitive tasks.

References

[1] Carvalho Barbosa R, Shoaib Ayub M, Lopes Rosa R, Zegarra Rodríguez D, Wuttisittikulkij L. Lightweight PVIDNet: A Priority Vehicles Detection Network Model Based on Deep Learning for Intelligent Traffic Lights. *Sensors (Basel)*.2020; 20 (21): 6218. Published 2020 Oct 31. doi: 10.3390/s20216218

[2] Mansour and M. R. M. Rizk, "Towards Traffic Congestion-Free through Intelligent Traffic Control System, " 2019 7th International Japan-Africa Conference on Electronics, Communications, and Computations, (JAC-ECC), 2019, pp.5-8, doi: 10.1109/JAC-ECC48896.2019.9051064.

[3] Shukla, Suraj & Kandeth, Aadithya & Santhiya, D & Jayavel, Kayalvizhi. (2018). Efficient Traffic Management System. International Journal of Engineering & Technology.7.926.10.14419/ijet. v7i3.12.16563.

[4] Elkhatib, Mohammed & Adwan, Alarqam & Alsamna, Ahmed & Abu-Hudrouss, Ammar. (2019). Smart Traffic Lights using Image Processing Algorithms.1-6.10.1109/PICECE.2019.8747225.

[5] Klilou, Abdessamad & Bourzeix, François & Bourja, Omar & Zennayi, Yahya & Mabrouk, Lhoussein & Belkouch, Said. (2015). Real-time parallel implementation of road traffic radar video processing algorithms on a parallel architecture based on DSP and ARM processors.10.1109/ISDA.2015.7489222.

[6] Arjun Raja and R. Swarnalatha, “Smart traffic signal by digital image processing using PCA algorithm and edge detection techniques” Journal of Engineering and Applied Sciences 12 (22): 6076-6082, 2017

[7] Taqi Tahmid and Eklas Hossain, “Density Based Smart Traffic Control System Using Canny Edge Detection Algorithm for Congregating Traffic Information” 2017 3rd International Conference on Electrical Information and Communication Technology (EICT), 7-9 December 2017, Khulna, Bangladesh