International Journal of Information & Computation Technology.
ISSN 0974-2239 Volume 6, Number 1 (2016), pp. 11-18

http://www. irphouse.com

Design of Intelligent Traffic Management Systems using Computer Vision and Internet of Things

Angad Kundra and Aman Maheshwari

Department of Computer Science, BITS Pilani Dubai Campus, Dubai, U.A.E

Abstract

This paper aims at giving a design for an intelligent traffic management system using computer vision and IOT. Computer vision can be used to implement a traffic counting mechanism at an intersection and prioritizing and prioritizing lanes with heavy traffic. It will also improve enforcement of traffic rules by capturing the number plates and image of drivers, the registration number will be captured by a letter & digit recognition software and storing it in a database. Emergency vehicles will be given priority using communication between sensors in emergency vehicles and traffic light system. This can be implemented using IOT, a type of V2X communication.

Keywords: Computer Vision, Intelligent Traffic Management System, Lane Prioritization, IOT.

INTRODUCTION

Traffic management is a major and ever-growing issue which even the most developed countries face. The conventional traffic signals which operate on time dependent systems is and out dated system in this era because it doesn’t manage traffic according traffic density in each lane. The proposed framework is an intelligent traffic management system which combines use of vision based traffic monitoring and use of IoT sensors to manage traffic in the most efficient manner. Using vision based traffic monitoring and IoT based sensors we try to implement an artificially intelligent system which is as good as a traffic policemen who conventionally manages traffic in a very efficient way. Using high end video camera and computer vision we can count the number of vehicles in each side of the signal and prioritize the lane with highest traffic density using a simple algorithm. The video captured by the cameras will be sent to a cloud storage where it can be analyzed by a computer vision software and retrieve the traffic density on each side of a traffic signal. After the analysis is complete the cloud will send the traffic timing information to all the traffic signals. Once the traffic signal in red the cameras will send any video frame involving traffic rules violation, the number plates of the traffic violator can be stored after detection using computer vision. In case of emergency the traffic signals will be embedded with IoT sensors in which each emergency vehicle will communicate with the nearest traffic signal through V2X communication by sending a request. This request will be sent to the cloud where it can be processed and after processing the cloud will send traffic signal timing information to the traffic signal again. The cloud will prioritize the traffic signal thereby allowing the emergency vehicle to pass first.

This way the central server can process information for each traffic signal and thereby reduce traffic at each signal. This will help in reducing traffic overall by reducing traffic at each point.

This paper is organized in following portions: a brief introduction followed by related work in this field and then the proposed design. The design may be categorized into two tasks i.e. implementation of vision based algorithm and implementation of IoT sensors for emergency.

RELATED WORK

A great amount of research has already been carried out in the field of image detection. Vehicles may be detected using background subtraction and frame difference technique. In [1] the tracking has been done using the frame difference method and it is widely used to detect moving objects through a fixed camera. This is the algorithm we will be employing to detect moving cars in the video. The algorithm converts the image captured by static camera to gray scale and then performs subtraction and the subtracted image is converted to binary image to get shape of the object which can then be classified into a car.

As in the case of classification of objects based on a data set, the computation time is usually greater as a result of which this mechanism is not used for our approach as it will not work efficiently in a real time scenario.

Static subtraction is a widely used method for detection of moving objects. A video which is a stream of frames at a given fps is subtracted from a given reference frame. This reference frame is the image having the background. However in external environments the background may keep varying as a result of which multiple invalid objects may be detected. In order to avoid this a series of images containing the background are averaged.

$$B(x,y,t) = \frac{1}{N}\sum_{i=1}^{N} V(x,y,t-i)$$

Where N is the number of images, x and y denote the pixel of the image and t denotes the time of the image at time t, i denotes the initial image time. After calculating B(x, y, t) we can calculate

$$|V(x,y,t) - B(x,y,t)| > \mathrm{Th}$$

Where Th is the threshold. Faster moving objects require higher threshold.

[2] Discusses that the error rate of background subtraction method is 17.596% - 17.742% with any gray scale and color video. The output binary image can be used for classification of cars. [3] uses Haar based methods to perform classification of the image. We can use this concept to classify the binary image of the car in an efficient time. Haar classifiers can be used to detect features of absolute intensity. For each feature a probability is calculated from the training data set which can then be used to classify the binary image.

V2X and V2I communication is the future of traffic management allowing communication between vehicles and infrastructure also between different vehicles. In this paper we will be making use of V2X communication to send information from an emergency vehicle to the approaching traffic light to prioritize that lane thereby allowing the emergency vehicle to pass first. IEEE 802.11p allows communication between vehicles and infrastructure using 5.96GHz bandwidth. [4] Describes the use of DSRC units for V2X communication. It also includes onboard units (OBU) and road side units (RSU). The OBU can send UDP packets and DSRC unit can convert it into non IP packets thereby sending it to every RSU in the coverage area.
The RSU can then send the received packet to the cloud server application where the information can be processed and the traffic signal timing algorithm can be overridden.

Once the traffic signal is red the cameras capture the number plates of any traffic violator. This can be done using similar background subtraction technique in which only the area of the intersection will be the reference image. Any vehicle in that region will get captured and the image will be processed on the cloud to retrieve the number plate and will be stored in a database. [5] uses DSP platform and consists of an alphabet and number recognition module. Character classification may be done using nearest neighbor classifier but in [5] support vector machine approach is used.

VISION BASED APPROACH

As stated earlier we will be using background subtraction and frame differencing to detect moving cars and the binary image output from these will be used to classify them as cars making the classification faster. This can be used to calculate the number of vehicles in a particular lane.

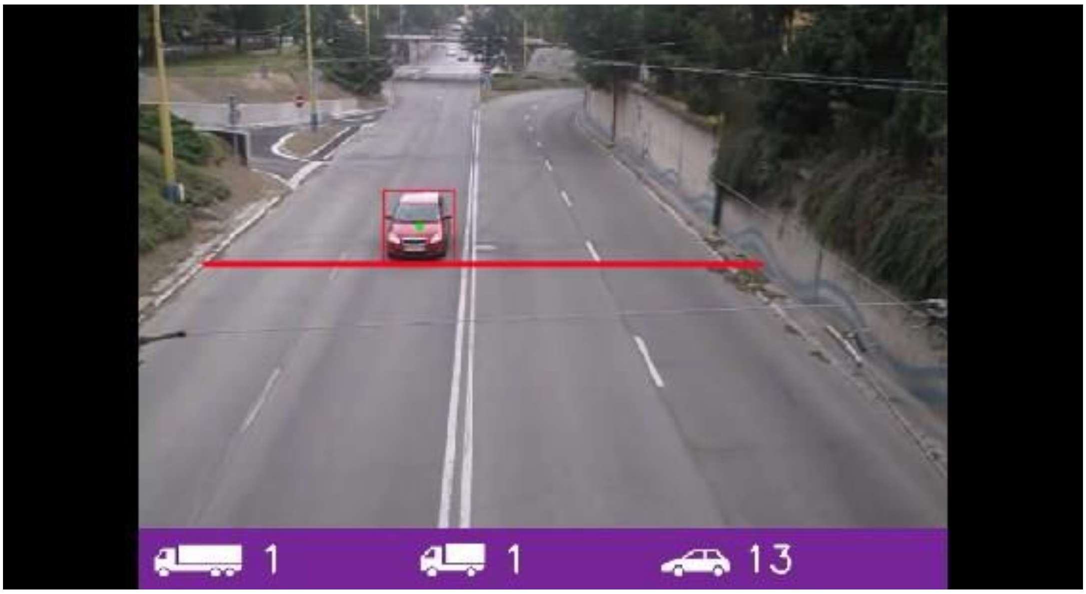

Figure 1

The figure 1 shows the number of cars calculated as 13 and number of LUV's as 1 and HUV's as 1. The frame differencing mechanism allows to put a box of the differentiated object and a green dot is placed on the center of the red box which when intersects with the red line increases the count.

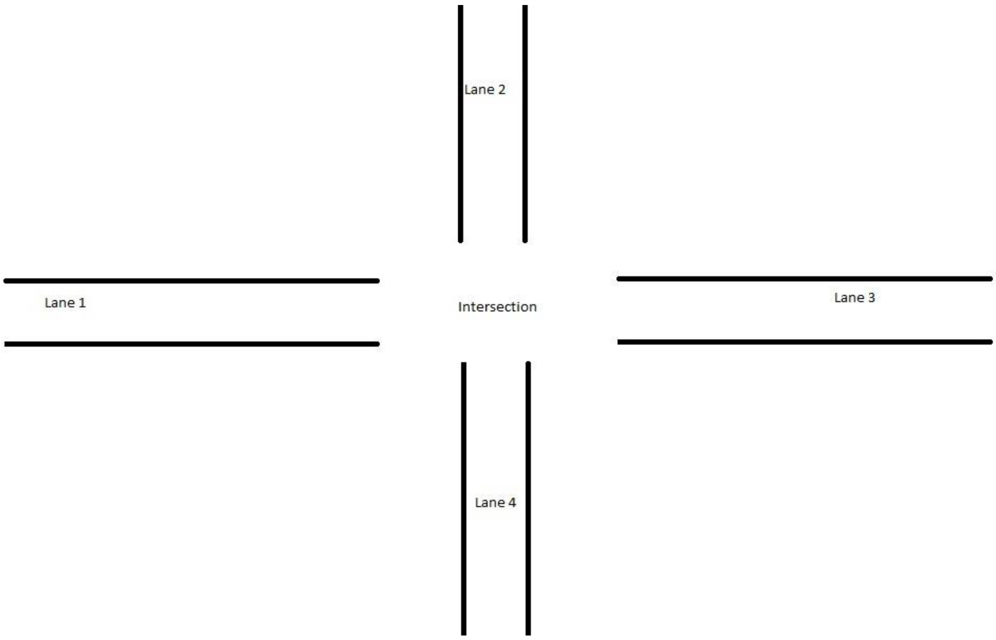

Figure 2

In an intersection let there be 4 lanes and traffic density of each lane be ln1, ln2, ln3, ln4.

Let the max green time be denoted by MAX_ALLOW. So that traffic density doesn't make only flow possible in one direction and thereby allowing different lanes to get a green signal as well.

Now suppose that lane 1 has green signal which means that lane 2, lane 3 and lane 4 has a red signal.

After the algorithm has determined the traffic density ln2 this data will be sent to the cloud and will assign a green time according to the density. Say lane 2 has a density of 40 cars. So the backend computation will tell the cloud to send a value of say 45 seconds (green_time) to lane 2. We can choose the value of MAX_ALLOW such that the red time of any lane is lesser than max red value. As a result:

3*(green_time + wait_time)<= MAX_STOP (red time)

MAX_ALLOW<= (MAX_STOP – wait_time)/3

Let the waiting time

Red time of lane 3 = green_time + wait_time

Red time of lane 4 = 2*(green_time + wait_time)

Red time of lane 1 = 3*(green_time + wait_time)

Similar computation will happen for lane 3 and lane 4.

Suppose lane 2 is green and lane 1, lane 3 and lane 4 are red. Now each of these lanes will also check for rule violator by comparing the image of intersection (background subtraction). If any extra object is encountered the camera captures the image and the image is sent to the cloud for processing. On the backend the image is used to detect the number plate using alphabet and numeric classification using support vector machine. The fetched image is store along with the number that has been classified in a database for further processing by transport department.

IOT BASED APPROACH

The main use of IoT comes into picture through V2X communication. In the vision based approach there is case of emergency vehicles which will always be overlooked to avoid that V2X communication is needed. Any emergency vehicle should be able to communicate with the nearest and approaching traffic light. The traffic light will therefore send information to the cloud which will override the algorithm and prioritize the lane through which the emergency vehicle is approaching. The UDP packet sent via V2X communication may contain the approach vector information and the location information of the emergency vehicle to help classify which traffic signal intersection is being approached in case of multiple traffic signals in the vicinity. Once the backend software has targeted the traffic signal the algorithm will execute an emergency function.

Here there may be 2 cases.

Case 1: The emergency vehicle is approaching from the direction of the lane which is already green.

In this case suppose in figure 1, the emergency vehicle is approaching from lane 1 and lane 1 is green already. The algorithm will execute emergency function on receiving request from the emergency vehicle. As a result of this lane 2, lane 3 and lane 4 timers will stop decrementing and lane 1’s green timer will also stop thereby allowing emergency vehicle to pass first.

Once the vehicle has passed the system will turn lane 1’s light to yellow and will check for lane 2’s traffic density and start from assigning green time lane 2 and red time to lane 1, lane 3 and lane 4.

Case 2: The emergency vehicle approaches from a lane which is red.

In figure 1 suppose the emergency vehicle is approaching from lane 1 and the lane 1 is red and lane 2 is green. In this case as soon as the request is received for emergency the emergency function of the algorithm is executed which will turn the green light of lane 2 to yellow for wait_time and then turn it to red and lane 1 will be turned green. Whereas lane 3 and lane 4 will have their timers stopped and lane 2 will also have its red timer stopped. Lane 1 will stay green till the time the vehicle has passed and after that the lane 1 will turn yellow for wait_time and the algorithm starts again by allocating time to lane 2.

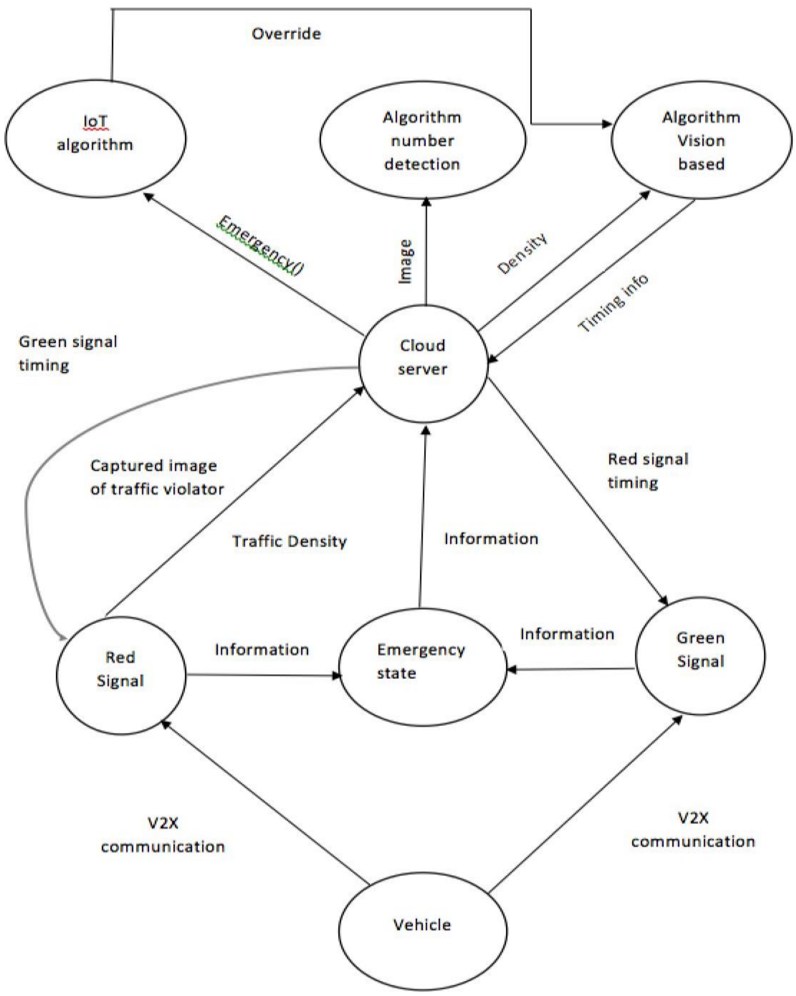

Figure 3

NORMALIZATION

There should be a normalized scheme for allocation of green light to signals. The max wait time should be such that the traffic density doesn't increase drastically also it should be such that there is a minimum amount of traffic for flow. Max wait time should be no more than approx. 1.5 mins - 2mins. Assuming wait_time of 5 seconds.

So, MAX_STOP = 2mins, so MAX_ALLOW becomes 38.3 seconds. also say a row of traffic (3 lane road) takes about 5 seconds to cross the intersection. Now max traffic count can be 21 (7 rows of traffic).

So the counter should stop after every count reaching 21. The calculations may vary for different scenarios. This calculation is valid for a typical city scenario.
Normalization table values for different density of traffic

Row number	Traffic range	Time Allocated (seconds)
1	1-3	5.7
2	4-6	11.4
3	7-9	17.1
4	10-12	22.8
5	13-15	28.5
6	16-18	33.2
7	19-..	38.3

CONCLUSION

The combination of IoT sensor based approach and vision based approach helps us in developing an artificially intelligent system that can handle traffic better than a traffic conductor. As the error rate of both the approaches is very less most of the traffic can be fairly and efficiently be handled and will therefore help in decreasing the traffic congestion on intersections at peek points. Future work on the system can be to automate handling of fines on traffic violators. Also as the traffic density information is stored on a central server the data can be handled such that the overall traffic also reduces by taking into account the traffic density of traffic lights after and before the specific traffic light.

REFERENCES

[1] N. Singla,"Motion Detection Based on Frame Difference Method", International Journal of Information & Computation Technology,Volume 4, Number 15, 2014

[2] T. Intachak, W. Kaewapichai, "Real Time Illumination Feedback System for Adaptive Background Subtraction Workin in Traffic Video Monitoring", International Symposium on Inteligent Signal Processing and Communication System (ISPACS), 2011

[3] Youpan Hu, Qing He, Xiaobin Zhuang, Haibin Wang, Baopu Li, Zhenfu Wen, Bin Leng, Guan Guan, Dongjie Chen, "Algorithm for Vison-Based Vehicle Detection and Classification", International Conference on Robotics and Biomimetrics (ROIO), 2013

[4] Xutin Duan, Yue Yang, Daxin Tian, Yunpeng Wang, Tao Li, "A V2X Communication System and Its Performance Evaluation Test Bed", Innovation Center of Modern Urban Traffic Technologies, 2012

[5] Clemens Arth, Florian Limberger, Horst Bischof, "Real-Time License Plate Recognition on an Embedded DSP-Platform", IEEE, 2007