Compactness Established Self-Motivated Traffic Control Structure

B.Hakkem[1], M.Vignesh[2]
[1, 2] *Assistant Professor, Department of Electronics and Communication Engineering, Hindusthan Institute of Technology, Coimbatore-641 032*

***Abstract*- Traffic congestion is a major problem in many cities across the world and it has become a nightmare for the commuters in these cities. Conventional traffic light system is based on fixed time concept allotted to each side of the junction which cannot be varied as per varying traffic density. Junction timings allotted are fixed. Sometimes higher traffic density at one side of the junction demands longer green time as compared to standard allotted time. Here Image processing technique is used to sense the traffic density of every road in the junction and release the heavily dense road frequently. The system uses a camera fixed on a gear motor for image acquisition. The image captured in the traffic signal is processed and converted into grayscale image then its threshold is calculated based on which the contour has been drawn in order to calculate the number of vehicles present in the image. After calculating the number of vehicles we will came to know in which side the density is high based on which signals will be allotted for a particular side. Raspberry pi model B is used as a microcontroller which provides the signal timing based on the traffic density. The system will append a growth to Computer vision area with competent maneuver restoring the ancient timer based traffic control system.**

***Index Terms*- Traffic congestion, Traffic density, Image processing, Raspberry Pi, etc.,**

I.INTRODUCTION

Traffic congestion is one of the foremost problems faced by the urban and suburban dwellers of today. Traffic congestion is a condition on transport networks that occur when use of vehicles increases, and is characterized by slower speeds, longer trips, and increased vehicular queuing. When traffic demand is great, enough that the interaction between vehicles slows the speed of the traffic stream, this results in some congestion.

The multiple reasons for traffic congestion are

- Too many cars for the roadway due to inadequate mass transit options or other reasons.
- Obstacles in the road causing a blockage and merger.

These can be any of the following:

  - Double parking
  - Road work
  - Lane closure due to utility work
  - Road narrowing down
  - An accident

- Inadequate green time.
- Too many pedestrians crossing not permitting cars to turn.
- Too many trucks on the road due to inadequate rail freight opportunities.
- Overdevelopment in areas where the mass transit system is already overcrowded and the road system is inadequate.

Traffic congestion has a number of negative effects:

- Wasting time of motorists and passengers ("opportunity cost"). As a non-productive activity for most people, congestion reduces regional economic health.
- Delays: Delays, which may result in late arrival for employment, meetings, and education, resulting in lost business, disciplinary action or other personal losses.
- Inability to forecast travel time accurately, leading to drivers allocating more time to travel "just in case", and less time on productive activities.
- Wasted fuel increasing air pollution and carbon dioxide emissions owing to increased idling, acceleration and braking.

- Wear and tear on vehicles as a result of idling in traffic and frequent acceleration and braking, leading to more frequent repairs and replacements.
- Stressed and frustrated motorists, encouraging road rage and reduced health of motorists
- Emergencies: Blocked traffic may interfere with the passage of emergency vehicles traveling to their destinations where they are urgently needed.
- Spillover effect: Spillover effect from congested main arteries to secondary roads and side streets as alternative routes are attempted ('rat running'), which may affect neighbourhood amenity and real estate prices.
- Higher chance of collisions due to tight spacing and constant stopping-and-going.

II. LITERATURE SURVEY AND RELATED WORKS

A lot of techniques are generally used to measure the traffic density to control traffic signal. Image processing is one of the most popular techniques. Many methodologies are used in image processing for traffic regulation. The most widely used one is based on Edge detection [1]. This paper uses 'Canny edge detection algorithm' for processing. In Canny method, Gaussian smoothing is used hence this makes harder to detect the junctions and corners due to blurring. The efficiency of this system is not up to the expected level [1]. Some methods use IR transmitter and receiver for revealing the vehicles density. This method also faced some consequences because of the Line-of- Sight transmission. By eliminating the need for installation of additional devices like RFids, traffic is regulated [2]. Some methods use image processing along with the mathematical procedure for traffic regulation [3]. This method uses the camera mounted at the cross-road and the videos are obtained for processing. In this method, the traffic flow is not controlled by considering the vehicle density. One important duty of traffic police is to allow the emergency vehicles on the road. It is also possible to allocate way for ambulance using image processing and microcontroller [4]. Various image processing techniques are currently in use. Image enhancement techniques are used to control the traffic in efficient way [5]. This method uses the image processing for controlling the traffic light by using microcontroller. This lowers the time wasting in traffic signal.

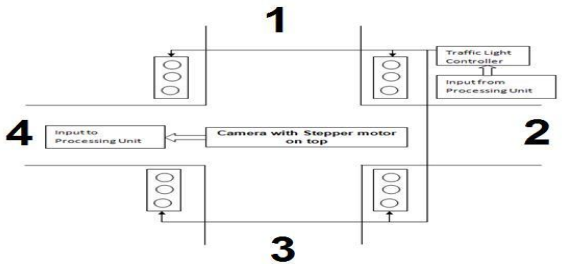
We have proposed an efficient traffic density monitoring system based on the density of the vehicles on each lane by using image processing and Raspberry Pi. This method will be more effective for controlling traffic lights than conventional methods.

III. SYSTEM ARCHITECTURE

Figure 1 shows a typical Traffic light controller along with a camera at the center (Intentionally fixed at center to cover all the lanes). Conventionally the controller is preset to time intervals for opening and closing a signal. In the proposed system the control signal modifies the time intervals based on the density of the traffic monitored. Since most of the signals are monitored by the control rooms, transferring data from camera to processing unit and then processing unit to controller. In this prototype a Wi- Fi enabled CCTV camera is used for monitoring the density of the traffic. Only one live frame is sent to the PU (Processing Unit). The frame received by the PU (Processing Unit) is preprocessed for edge detection. The edge image is then compared with the reference image which is taken during the leisure time (at day, evening and night time). The comparison results are used to make decision on control signal. When signal at side 1 is opened the camera is focused on side 4 for getting the speed side 1 is opened for X seconds. Let the time taken to process the video frame and comparison is n seconds. The sum of delay in transmission of control signal and controller delay is m seconds. Now the following equation gives the time at which the frame has to be taken.

Frame _time = X – (n + m) sec (1)

Figure 1: Overview of Density Based Dynamic Traffic Control System



The PU (Processing Unit) will be programmed to take the frame occurring at Frame time for processing and the control signal is generated as explained in the algorithm.

The controller used here is Raspberry pi 3 with Wi-Fi since the control signal from PU is transferred to the controller via Wi- Fi.

IV. DESIGN OF DENSITY BASED DYNAMIC TRAFFIC CONTROL SYSTEM

The main objective of the density based dynamic traffic control system is to control traffic signals with the help of the surveillance camera present at the junction points. The frames are obtained from the camera through continuous video processing. An image from the camera is used to calculate the number of vehicles in each lane. According to the number of vehicles in each lane, the time for green signal is given to respective lane which varies from time to time. If there are same numbers of vehicles in all lanes, the control signal is turned to the default case.

A Density Based Dynamic Traffic Control System proposes a new technique for controlling the traffic signal by image processing and Raspberry Pi module. A camera will be fixed at the center of the road along with the Gear motor. The camera is rotated through 360o in clockwise and anti-clockwise direction by using a Gear motor and the images are captured in sequences. In this method, the system uses a camera fixed on a Gear motor for image acquisition. The acquired image is converted into frames and processed further for edge detection by using Prewitt algorithm. The edge detected image from one lane is compared with other lane for measuring the density and the decision is done based on the outcomes. The control signal is given for traffic regulation. This method uses image processing for obtaining the edge detected images. Here, the Prewitt edge detection algorithm is used to overcome the difficulties caused by the canny operator. Figure 2 shows the block diagram for proposed technique.

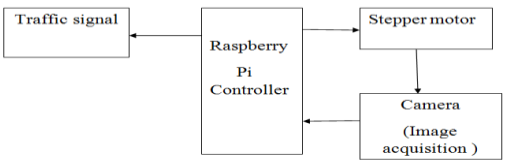


Figure 2: Block diagram of Density based dynamic traffic control system

A Density Based Dynamic Traffic Control System uses Raspberry Pi controller for processing. Raspberry Pi is a small computer, a SoC device, which runs on the Linux operating system specially designed for it, named Noobs. It is capable of doing multitasking. Here, motor is used to rotate the camera. Hence, the need for more cameras to capture images is reduced.

Edge ← Edge (F, Prewitt) / Edge Detection
Ref_im ← Select (Rj) /Based on current time select the Reference frame
result = abs (Ref_im – F_Edge) / Comparison and taking absolute values
if 0 <result<Th1
op← 0
send op to controller elseif Th1 <result < Th2
op← 1 t ← 0
send op, t to controller else
op←1
t ←1
send op, t to controller
At controller side:
Input: Open (op), timing (t)
output: signal open / close, skip if op == 0
skip the signal if op ==1
if t == 0
open signal
No change in time else
open signal
Increase time by 25%

V. SYSTEM DESIGN

In Dynamic traffic control system, the hardware used are:

- Image acquisition unit- It contains USB web camera and gear motor to capture images,
- PC with Linux OS - For all the image processing work,
- Raspberry Pi unit – for signal prioritizing,
- Toy cars for the prototype of a road junction and traffic model.

1. IMAGE ACQUISITION UNIT

Digital image acquisition is the creation of digital images, typically from a physical scene. The term is often assumed to imply or include the processing,

compression, storage, printing, and display of such images. The most usual method is by digital photography with a digital camera but other methods are also employed. Digital image can also be computed from a geometric model or mathematical formula. In this case the name image synthesis is more appropriate, and it is more often known as rendering. Digital image authentication is an issue for the providers and producers of digital images such as health care organizations, law enforcement agencies and insurance companies. There will be more time to shoot and will be able to cover assignments more effectively. Digital imaging means that photographers no longer have to rush their film to the office, so they can stay on location longer while still meeting deadlines.

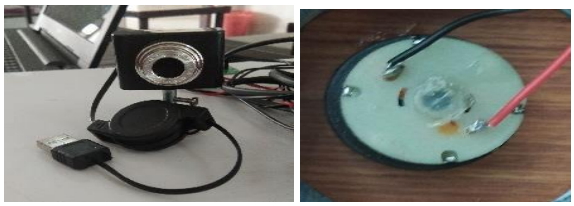


Figure 3: Image acquisition unit

2. RASPBERRY PI UNIT

Raspberry Pi unit is used for controlling the entire efficient traffic control system. It has 64bit ARMv8 Quad Core Processor powered Single Board computer running at 1.2GHz. It has inbuilt 1GB RAM. Raspberry Pi 3 model B contains BCM43143 WiFi and Bluetooth Low Energy (BLE). It has 40pin extended GPIO and 4 x USB 2 ports. Its main advantage is, it contains 4 pole Stereo output and Composite video port. In addition, it is possible to use Micro SD port for loading your operating system and storing data. Complete compatibility with Raspberry Pi 1 and 2. Publicly documented 3D graphics core VideoCore IV 3D subsystem at 400MHz and the 3D core at 300MHz. Energy efficient - Do more but consume lesser power. Figure 4 represents the diagram of Raspberry Pi 3 model B.

Figure 4: Raspberry Pi model

3. IMAGE PROCESSING UNIT

Image acquisition is done with the help of the USB web camera and motor. Here Raspberry Pi with Raspbian operating system is used for entire processing. Image preprocessing functions are imported from OPEN CV libraries and is included in final python program. This will automatically process the image when the program is invoked.

The videos obtained using Web camera along with the motor is converted into frames and these frames are taken for processing. Among the key features of an image i.e., edges, lines, points, we have used edge in our present work which can be detected from the abrupt change in gray level. An edge essentially demarcates between two distinctly different regions, which means that an edge is the border between two different regions. Here we are using edge detection method for image matching. Edge detection methods locate the pixels in the image that correspond to the edges of the objects with the detected edge pixels. The algorithm for edge detection used here is Prewitt operator. Figure 5 gives the working of PU.

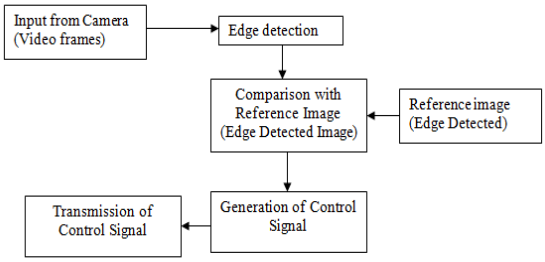


Figure 5: Working of PU

VI. PROTOTYPE OF THE PROPOSED SYSTEM

Figure 6 represents the prototype of the proposed system.

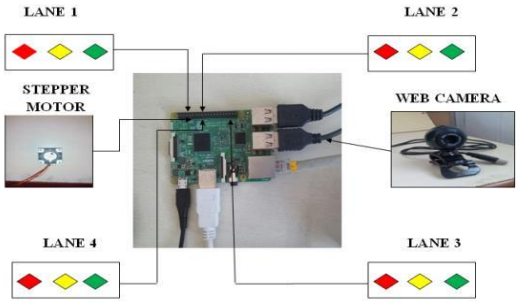


Figure 6: Prototype of the proposed system

VII. MODEL EXECUTION

The existing system uses the microcontroller for traffic regulation. In this scenario, the green time is provided to all lanes in cyclic order. If the lane 2 contains no vehicles and the lane 4 has more number of vehicles, but the microcontroller based monitoring system opens the lane 2 alone. Because, the control is given in cyclic manner. This makes the people in lane 4 to wait for a long time. This leads to Just-in- Case time. In this same scenario, if lane 4 contains more traffic density it will wait for its green time. This increases the traffic congestion hence there by increases the waiting time. But, our proposed system solves this problem very easily by the use of image processing and raspberry pi. Here the density is calculated from each lanes and compared to get the control signal. Figure 7 represents the traffic monitoring system of existing system. Figure 8 represents the dynamic traffic control system of the proposed system.

Phase	Density	Preferences	Green Light Timing(s)	Red Light Timing (s)
1 to 2	130	I	60	30
2 to 1	98	II	60	30
3 to 4	112	III	60	30
4 to 3	84	IV	60	30

Table 1: Traffic light timing as per controller preset

Phase	Density	Preferences	Green Light Timing(s)	Red Light Timing (s)
1 to 2	130	I	55	35
2 to 1	98	III	40	25
3 to 4	112	II	50	30
4 to 3	84	IV	30	15

Table 2: Traffic light timing as per density estimation

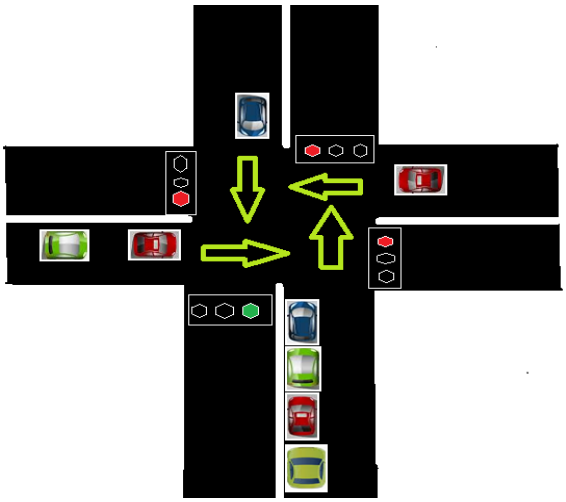


Figure 7: Traffic regulation in Existing System

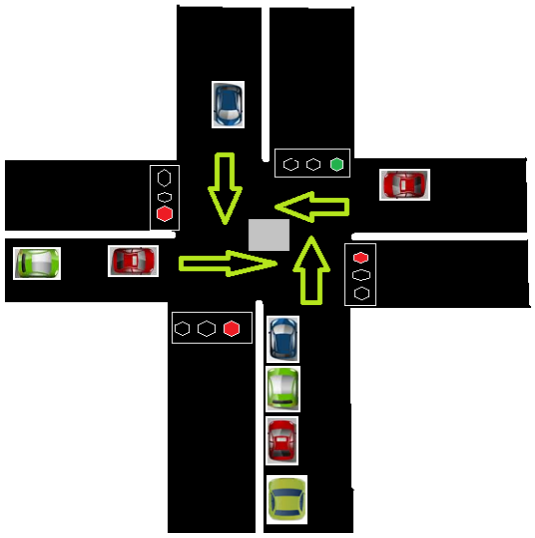


Figure 8: Density based dynamic traffic control system

Table 1 gives green light and red light timings according to microcontroller based traffic control system. Table 2 represents the green light and red light timings according to density based estimation system.

VIII. CONCLUSION

This project tries to introduce a new method for traffic control based on vehicle density, which is found to be very efficient method. As opposed to digital image processing using bulky computers, this method uses Raspberry Pi module, a microcomputer for image processing which greatly reduces the space and processing time. An external USB Camera is used to capture images since the Raspberry Pi camera module may be prone to changing environments as the software used in this is open source. To determine traffic density different edge detection techniques are used. As far as this project is concerned with Prewitt edge detection is much more efficient as compared to other operators. This is due to its simplicity, less computational time, easy detection of edges and less sensitivity to noise. In addition, to implement the traffic controller in smart cities this method plays an imperative role. This method is not possible in existing roadways. This method is effectively implemented if the roadways are constructed in accordance with our idea.

VIII. FUTURE ENHANCEMENT

This method can be further enhanced by proposing a system that provides green time for pedestrian crossing also. The accuracy can be improved by using thermal image processing. By using higher versions of Raspberry Pi, the processing time can be reduced.

IX. ACKNOWLEDGEMENT

The authors would like to thank Ms. G. Pavithra Devi, M.E., whose guidance and support from the initial to the final level enabled us to develop an understanding of the subject. On a more personal note we would like to thank Mr. Dr. D. Ganesh kumar, Head of the Department for supporting us, encouraging us and making us realize that there is a whole world outside our research.

Lastly, we offer our regards and blessings to all of those who supported us in any respect during the completion of the project.

REFERENCES

[1] Mohammad Shahab Uddin, Ayon Kumar Das and Md. Abu Taleb, (2015) 'Real-time Area Based Traffic Density Estimation by Image Processing for Traffic Signal Control System: Bangladesh Perspective', Conference on Electrical Engineering and Information & Communication Technology.

[2] Naeem Abbas, Muhammad Tayyab, M.Tahir Qadri, (2013)'Real Time Traffic Density Count using Image Processing', International Journal of Computer Applications (0975 – 8887) Volume 83 – No 9

[3] Nguyen Viet Hung, Le Chung Tran, Nguyen Hoang Dung, Thang Manh Hoang and Nguyen Tien Dzung, (2016) 'A Traffic Monitoring System for a Mixed Traffic Flow Via Road Estimation and Analysis', IEEE.

[4] Parthasarathi. V, Surya. M, Akshay. B.Murali Siva.K and Shriram K.Vasudevan, (2015) 'Smart Control of Traffic Signal System using Image Processing', Indian Journal of Science and Technology

[5] Vismay Pandit, Jinesh Doshi, Dhruv Mehta, Ashay Mhatre and Abhilash Janardhan, (2014) 'Smart Traffic Control System Using Image Processing', International Journal of Emerging Trends & Technology in Computer Science (IJETTCS) Volume 3, Issue 1, January – February