International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 03 | Mar 2019 www.irjet.net p-ISSN: 2395-0072

Secure Ingenious Automated Traffic Controller by Real Time Traffic Density Detection using IoT

Roshan R. Poduval[1], Dhiraj M. Jadhav[2], Ankur S. More[3], Saahil V. Nar[4]

[1,2,3,4]*Students, Dept. of Electronics & Telecommunication Engineering, W.I.E.E.C.T, Thane*

---***---

Abstract - *The traffic signal which can be programmed to be manipulated with the continuously varying traffic density, will reduced the problem of traffic congestion to a significantly lower level. This Paper describes a smart & lucrative idea to manage traffic density effectively. Using Image Processing as basis of this implementation, can help in optimizing traffic density in crowded intersections. In this project, camera sensors can help in identifying vehicles and gives real-time counts of vehicles in intersections, the corresponding data is made available to Arduino which will control traffic system according to density and employs an automated Traffic system. User can access this Traffic density using WI-FI along the intersection as well as online with smart deployment of (Internet of Things) IoT. Using IoT can help in creating secure and automated Traffic systems along the intersections. This will help government and citizens work in tandem.*

***Key Words*: Image-Processing, Internet of Things (IoT), Arduino Uno, Camera sensor, WI-FI, Python OpenCV, Traffic Density management.**

1.INTRODUCTION

The current traffic control system in the metro cities of India are inefficient due to randomness in the traffic density pattern throughout the day [1]. The traffic signal timers have a fixed time period to switch traffic between different directions. Due to this, the vehicles have to wait for a long time span even if the traffic density is very less. If the traffic signal timer (TST) can be programmed to be manipulated with the continuously varying traffic density, the problem of traffic congestion can be reduced to a significantly lower level. Image-Processing & smart cameras provide an enormous amount of real-time traffic data.

India's first challenge is to put these different traffic data sources together and analyze them, to make use of these information. Very first step towards a better traffic flow is to optimize traffic signals within a single network. Indian traffic density consists of various types of vehicles, 55 % Indian roads comprises of 2-wheeler vehicle [1]. So, the next step is to use different lane density data and then to combine these traffic data to create a detailed picture of current traffic scenario. The last step is to get the right tool that can even automate traffic for up to an hour. With an availability of vast Data analytic tools, data can be obtained from the cluster of Traffic Management Systems, by aligning these in real time with IoT, can provide information to the authorities and public, thus reducing traffic pile up under extreme cases.

Real-Time Transportation systems (R-T-S) suggest real-time traffic systems as an effective solution to apply traffic control strategies which provides drivers with traffic information. Obviously, a reliable real-time traffic information is obtained through three steps: data collection, data processing and dissemination in which a great variety of technologies has been developed, [3]. Another term optical flow is also used by roboticists, encompassing related techniques from image processing and control of navigation including motion detection, object segmentation, time-to-contact information and motion compensated encoding [4], [5]. These data help in understanding the traffic versatility and its behavior. Since, many parameters is observed while understanding this versatility, it isn't a batch wise process rather it's more of a real time behavior and can only be understood with a real-time feed-back system.

As an example of such real-time traffic condition information, we can refer to real-time of traffic scenario in Shanghai. In 2003, the Shanghai Urban Construction & Communications Center, to serve the drivers with real-time traffic information of Shanghai's highways, 196 electronic traffic information boards were installed to show different levels of congestion with three colors; red, yellow and green." Red means blocked (speed less than 20km/h), yellow indicates jammed (speed between 20km/h and 45km/h), while green shows smooth traffic (speed of more than 45km/h). Data collection in this system is done through 3,278 underground sensors installed at 300-400m intervals on the main roads. As mentioned above, the received data from many detectors was integrated and translated to provide traffic information [6].

2. Process Overview

This project Emphasizes on creating Secure Ingenious Automated Traffic Controller by Real Time Traffic Density Detection. For creating an R-T-S system following mechanism has be employed in this project, Image capture sensor, Image segregation block, Image-Processing block, Comparator to compare images of different lanes and generating vehicle count, Signal Driver, by count feed-back mechanism, WI-FI module to send this information to the control station and share this information with commuters at Traffic junctions.

3. Functional Description

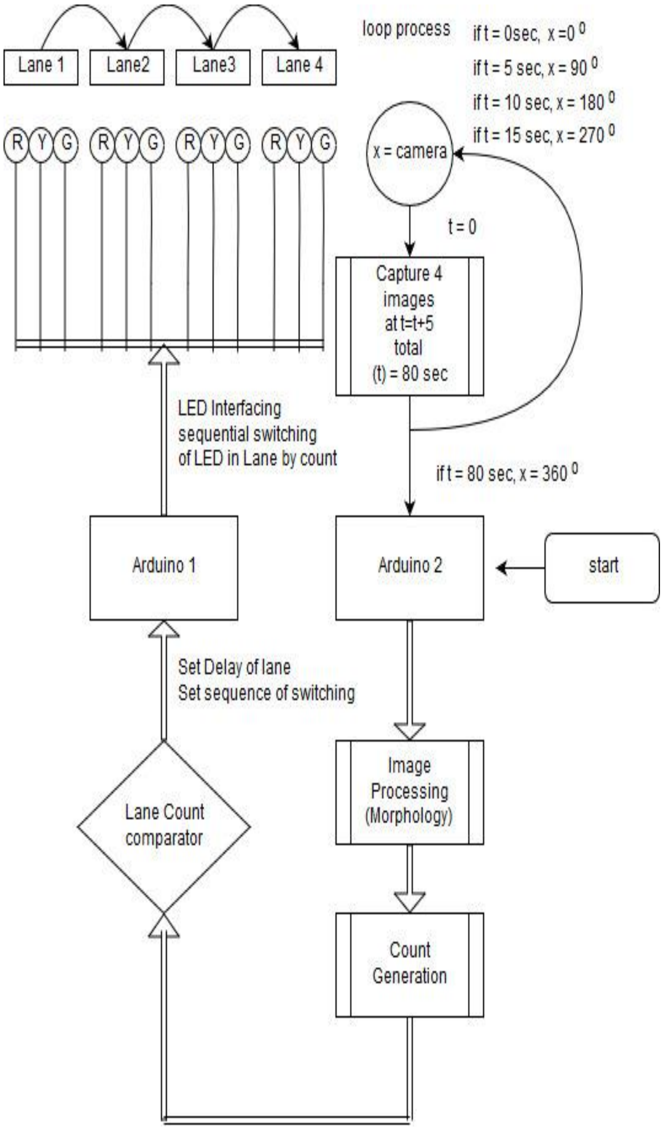

Figure 1: Functional Flow Chart

3.1 Arduino UNO function

In this project 2 Arduino Uno is used, where 1st Arduino is used for 12 Led Interfacing & connecting Wi-Fi module. 2 nd Arduino is used for stepper movement where camera sensor is connected to the stepper module creating a 360^0 rotating camera for taking images of 4 Lane.

3.2 Stepper Motor function (BYJ48 5v)

Stepper motor is connected to 2nd Arduino at digital pin 8,9,10,11 & given a power supply of 5v with a GND. This stepper motor is operated by Python – Arduino serial communication by Arduino serial monitor, here, serial library is used for creating a bridge connection between Arduino & Python. The motor is operated at every 80 sec, & it makes a total rotation of 360^0.

3.3 Camera function

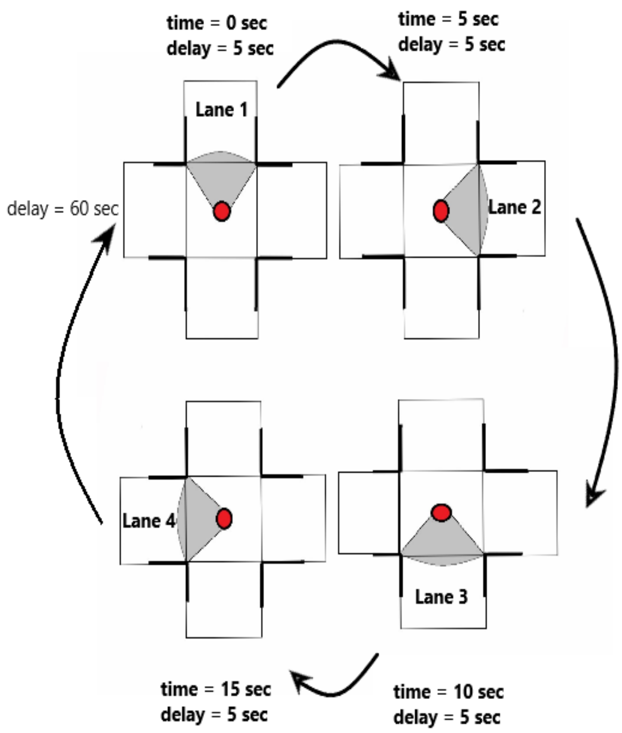

Figure 2: Rotation of camera

In this project 360^0 rotating camera module is used to rotate with the help of stepper motor & assign images at every 80 sec to the Image processing block. The stepper motor is interfaced to the 2nd Arduino.

At time = 0 sec the stepper motor stays at 0^0 and camera takes image of lane 1. At time = 5 sec stepper motor gets high, rotates total 90^0 and camera takes image of lane 2.

At time = 10 sec the stepper motor turns high and rotates a total of 180^0 and camera takes image of lane 3.

At time = 15 sec stepper motor rotates total of 270^0 & takes image of lane 4 after time=20 sec the motor remains low for 60 sec.

After, total of 80 sec in operation it is again initialized to time = 0 at lane 1. This process is executed in a loop when the system is in on state. These image files are further processed in image processing block.

3.3 LED (3v) function

In this project, LED is the Real-time output indicator. There are total of 12 LEDs used in this project, where each lane contains 3 LED for every four lanes. These LED operates at 3v power supply, and occupy 1st Arduino pins from 13 to 2. 5v power supply of Arduino is converted to 3v by interfacing a 220-ohm resistor in short configuration with 12 LEDs.

Three LEDs in a Lane are of color Red, Yellow, Green, it glows in a sequential manner from Red to Yellow for 5 sec and 5 sec from Yellow to Green. Green glows for 30 x (delay) sec. This delay is set by number of counts of the vehicle by 2nd Arduino, LED Green remains in High state only in any one Lane and Rest of 3 Lane will have Red as High state. After the total time the Green LED switches back to Yellow within 5 sec and then from Yellow to Red. High state of Red depends on Green Light glow of another single Lane.

1. If count = 1 to 3, Green light = (30x1) = 30 sec
2. If count = 4 to 8, Green light = (30x2) = 60 sec
3. If 5 < count $\leq$ 10, Green light = (30x3) = 90 sec
4. If count $\geq$ 11, Green light = (30x4) = 120 sec

3.5 WI-FI Module function (ESP8266)

ESP8266 is capable of functioning consistently in High Traffic, due to its wide spread operating temperature range. In this project, ESP8266 is interfaced with 1st Arduino, after image processing to send information to the data-base which can be accessed by both the in authority as well as the commuters. It gives a message after image is processed and gives the count of vehicle in each lane in the form of messages. User can connect to the data-base at Traffic junction with the help of this WI-FI module creating a perfect IoT environment.

4. Image Processing

The four different images by camera is processed with the help of image processing. In this project, process of morphology is employed with the help of python programming language using the library OpenCV [7].

First and the foremost step is to detect the type of background, it plays a big role in filtering of an image and detects optical flow of an image. For this, image is first resized to 340 x 220 pixels and a mask is created of this image by converting it to a proper HSV image within the specified resize limit (340 x 220) and setting an upper bound and lower bound value of array of pixels.

Lower bound value = [0, 99, 100]
Upper bound value = [50, 255, 255]

These lower bound and upper bound values help in deciding the image intensity at each [Red, Green, blue] levels in a pixel, for further converting it into an HSV image. Less the value more that color will be detected & will be converted in white and rest of the image into black.[8][9].

In OpenCV,[7] morphology gives best solution for differentiating the white spaces in these images. Here, mask of an HSV image is used again for morphology. With the help of white spaces in the image it will detect counts (of vehicle) in an image. This process is associated with opening and closing.

It gives following [7][10] transformation: -

1. Extracting Structural information.
2. Create image model such as
   Image model = structure + texture(details)
3. Manipulating White space and detecting count of the vehicle.

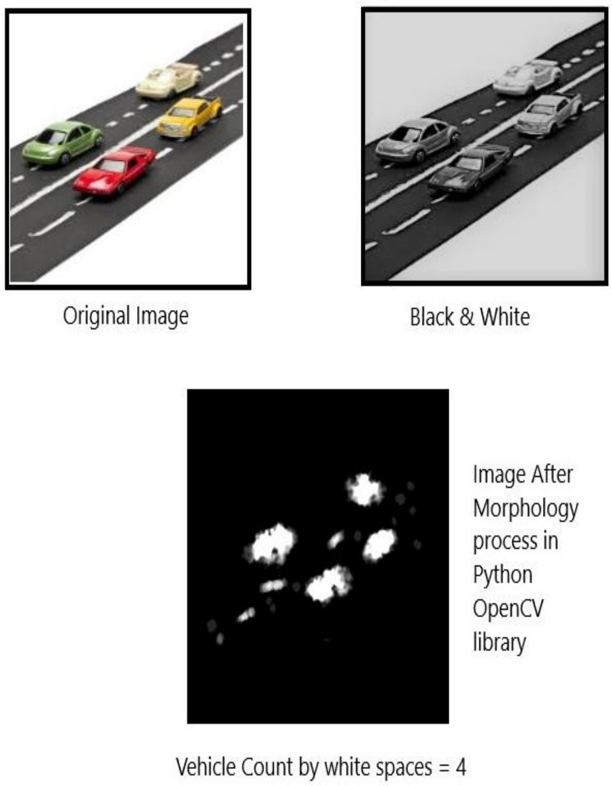

Figure 3: Morphology and Count detection

5. Requirements and configuration

5.1. Components and Pin configuration

Configuration	Pins used	Total Power
1st Arduino Uno	GND, Digtal pins 13 - 2	5v
2nd Arduino Uno	GND 1, GND 2, Analog A0, 3.3v, 5v	5v
RED LED x 4	pin 13, 10, 7, 4 of 1st Arduino	3v
YELLOW LED x 4	Pin 12, 9, 6, 3 of 1st Arduino	3v
GREEN LED x 4	Pin 11, 8, 5, 2 of 1st Arduino	3v
Stepper motor	Pin 8, 9, 10, 11, 5v ,GND of 2nd Arduino	5v
ESP8266 WI-FI module	Pin 3.3v, Analog - A0, Digital 3 and 4	3v

Table 1: Components Pin Configuration [7]

5.2. Software Specifications

Programming Language	Library	Library function
Python	OpenCV	Image Processing mainly for Image morphology and count detection
	Serial	For Giving Input to the Serial monitor of 1st Arduino
	numpy	Getting Image pixel, colour value in the form of matrix to form an array of upper bound and lower bound for HSV conversion
	Matplotlib	For Plotting Experimental values
Arduino C	Scheduler	For setting Delay of LEDs in 4 lane by count
Ardu blocks	Logic Blocks	For creating Logic of LED switching

Table 2: Software Packages and their Function [5][6][7]

6. Experimental Analysis

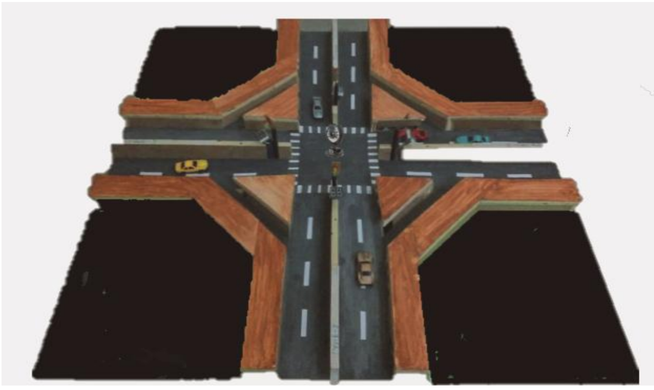

Figure: 4 Experiment Setup

This project Ensures, that preference of traffic signal switching must be only based on vehicle count and density. Hence, analogous to this idea the Experiment was performed by stimulation of artificial toy car for an hour for traffic signal automation (This was artificially created situation and must be only considered as ideal situation) and following results were observed.

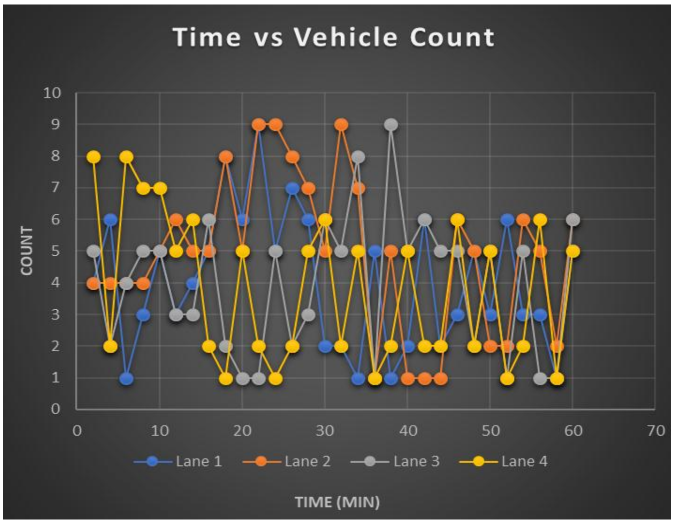

Chart 1: Time vs vehicle count in an hour in each Lane

The Traffic light switching process is fully automated for an hour and following values were obtained, at every 2 min following count were detected in Lane 1, Lane2, Lane 3, Lane 4 for an hour. From above chart it was observed that Lane 2 had higher average count followed by Lane 4, Lane 3 & Lane 1. With just the knowledge of count we could decide delay of each lane during its switching. From the above chart it can be observed that Lane 2 had higher average Time period of Green Signal High, followed by Lane 4, Lane3 & Lane 1.

Hence, more preference is given to Lane 2. Similar to above chart we can get information of many more parameter depending on our need. This information is further sent in the form of message with the help of WI-FI and stored in database, the user can access this data by connecting to the WI-FI system at that junction.

7. Conclusion

In the fixed Timer Traffic System, the lane signals are operated one after the another based on a fixed set timer. This causes traffic congestion on a high traffic lane due to wasted time on a low traffic lane, and this also restricts the use of new technology, but by this paper, we can get following ideas for the future:

1. An efficient and safe traffic ecosystem has been developed which provide commuters with live update of the road's traffic density.

2. The traffic signal timer (TST) can be programmed to be manipulated with the continuously varying traffic density, by this idea of implementation the problem of traffic congestion can be reduced to a significantly lower level. It's always necessary to set this timer by a proper Image-Processing Algorithm.

3. The automated traffic system discussed in this paper can reduce the time spent by commuters on traffic signals and reduce traffic congestion on busy lanes caused by extra time signals on empty lanes.

4. It can also ensure safety of pedestrians using the buzzer alert if vehicles block the zebra crossing.

5. By using rotating camera, the cost of installing 4 cameras along the junction and among a cluster of junctions is highly reduced. The cost to implementation ratio is low compared to conventional Traffic Signal Timer (TST).

6. It can also help commuters re-route their destinations using the live traffic update from the database created at the junctions. This IoT based automated traffic system can be a significant step towards the development of future smart cities.

7. We can make modification so that system can identify the vehicles, giving preferences to emergency vehicles.

8. In future we can use this system for video surveillance, vehicle & its number plate tracking for security purpose.

REFERENCES

[1] Introduction reference of "Indian roads traffic problems & common man problems in this traffic "https://www.youthkiawaaz.com/2017/12/indian-roads-traffic-problems-and-the-common-man/

[2] Stephen Ezell. "Intelligent Transportation Systems". The Formation Technology & Innovation Foundation (ITIF) (January 2010).

[3] " Amir Sodagaran1, Narjes Zarei and Zohreh Azimifar2 "Intelligent Traffic Information System a Real-Time Traffic Information System on the Shiraz Bypass."

[4] Beauchemin, Steven S., and John L. Barron. "The computation of optical flow."ACM computing surveys (CSUR)27.3 (1995): 433-466.

[5] OpenCV documentation for morphology process https://docs.opencv.org/2.4.13.7/doc/tutorials/imgproc/opening_closing_hats/opening_closing_hats.html?highlight=morphology

[6] Tarnoff, P.J., Bullock, D.M., Young, S.E., et al. (2009) Continuing Evolution of Travel Time Data Information Collection and Processing. Transportation Research Board Annual Meeting, Washington, 11-15 January 2009, Paper #09-2030.

[7] WIFI Arduino connection module – ESP8266 WIFI https://arduino-esp8266.readthedocs.io/en/latest/esp8266wifi/readme.html

[8] Morphological Image Analysis of Transmission Systems P. Radhakrishnan, B. S. D. Sagar, and B. Venkatesh, Member, IEEE TRANSACTIONS ON POWER DELIVERY, VOL. 20, NO. 1, JANUARY 2005

[9] J. J. Grainger and W. D. Stevenson, Power System Analysis. New York: McGraw-Hill, 1994.

[10] C. Torres-Huitzil, "Fast hardware architecture for grey level image morphology with flat structuring elements," IET Image Processing, vol. 8, no. 2, pp. 112–121, 2013.