Evolution of Information, Communication and Computing Systems (EICCS)
https://publisher.uthm.edu.my/bookseries/index.php/eiccs

Design Prototype of Traffic Light Control System using HSV Color Segmentation

Fausto Carlos Filipe da Silva[1*], Diky Siswanto[1], and I. Istiadi[1]

[1]Fakultas Teknik,
Universitas Widyagama, Kota Malang, Jawa Timur, INDONESIA

*Corresponding Author

Abstract:
Most traffic lights are installed at intersections to perform traffic control, which signaling (Red, Green, Yellow) can orderly the traffic flow. However, many of these traffic lights work with fixed time settings. With the technological advancement nowadays, especially in image processing and computer vision, it can create a dynamic time based on traffic density for a traffic light. The OpenCV library can be used to perform various tasks for image processing and computer vision for this project. Therefore, the proposed prototype system is designed to control a traffic light based on traffic density using image processing and HSV color segmentation. The HSV color algorithm for multiple color detection with the provided number of upper and lower for each car color (blue, green, pink and yellow) was applied to estimate traffic density and classified traffic conditions based on a number of vehicles on the road. It used a USB camera as a peripheral to capture road conditions, and Raspberry Pi 3 embedded the HSV algorithm for processing the video in real-time. The experiment result that ran in the room with a lighting intensity range of 230 – 246 lux showed that the HSV algorithm could estimate traffic density and classify traffic conditions based on the number of vehicles.

Keywords:
Color segmentation · Traffic light · HSV · Raspberry Pi · OpenCV

1. Introduction

Travelling to metropolitan cities in Indonesia consumes much time due to the increasing number of vehicles. So, traffic jam at intersections has become one of the many problems. A traffic light is installed to reduce traffic congestion to solve this issue. However, this traffic light works based on a time set in each direction; it causes some directions with a low traffic density to wait for a specific time for the green light. Therefore, an efficient traffic light control based on traffic density becomes significantly important to solve this problem other than this system is easy to install.

Much research has been proposed to address this issue using several image processing techniques; however, it still undergoes some drawbacks, leading to the system not working as expected. For instance, the object's position toward the camera can influence the accuracy of the Histogram of Oriented Gradient (HOG). The detection speed of this technique in the system depends on illuminations and camera position towards an object [1]. The Binary Large Object Detection (BLOB) technique cannot perform appropriately for heavy traffic conditions in which vehicles are sticking together. The BLOB reads all these vehicles into a single unit object vehicle with large dimensions [2].

*Corresponding author: *faustodasilva16@gmail.com*

Image processing techniques with statistical estimation methods have also been applied to the various traffic control system to solve this similar problem. However, it still has some drawbacks, such as the estimation method or the subtraction method working by comparing the object image with the reference image to determine the number of white pixels. It can be discovered that the percentage of queue length and vehicle density in which the distance between the vehicles tends to reduce the number of white pixels that represent the density of the vehicles is not accurate [3].

Open Computer Vision (OpenCV) is an image processing library that is beneficial for developing software and has many essential functions for digital image processing techniques, such as the Hue Saturation Value (HSV) color segmentation model. This technique takes a digital image with Red, Green, and Blue (RGB) format and converts it into the HSV color model. The HSV color format can handle the changes caused by illuminations and provide more accessibility to use color values for image processing. It has also been utilized for detecting multiple objects with outstanding results. This method has also can be used in recognizing emergency vehicles (ambulances) with low time processing. Image processing with HSV segmentation model could detect and track an object at any distance if the object is seen by the camera [4][5][6].

2. Traffic Light System

One of the primary benefits of signal control is that it reduces conflict points inside an intersection by dividing available time between conflicting traffic streams. However, the usage of traffic signals can cause serious road safety issues.

Signalized intersections are projected to account for 8% of all accidents in the United Kingdom when broken down by speed limit and accident severity. One of the primary issues with signalized intersections is that the automated control and hence simplification of the road environment comes at the cost of a driver deciding whether to stop when presented with an amber signal. This decision can be difficult in some situations, such as when the amber signal first appears because of the vehicle's speed and position. The driver is deemed to be in a dilemma when the amber time is insufficient for the driver to stop comfortably or clear the stop-line before the red signal appears. In this case, the driver must choose between passing through the intersection after the red signal has appeared, accelerating 'to beat the red' or braking hard. All three activities enhance the risk of an accident, with rear-end crashes being especially likely when the motorist accelerates or brakes forcefully before the intersection.

At signalized intersections, the most significant risk of an accident arises when drivers, for whatever reason, reject the red signal and go through the intersection. More than 20% of drivers in the United Kingdom who fail to stop at a red light are outside the dilemma zone and should stop comfortably. A traffic light violation occurs when a driver, for whatever reason, fails to stop their vehicle when a red signal instructs them to do so and continues driving over the stop line, usually into and through the intersection.

Violation of traffic lights is a serious problem since it generates a high-risk conflict situation and raises the possibility of an accident. A traffic light violation accident occurs when a driver disregards the red signal and collides with another vehicle or pedestrian due to his or her actions. Traffic light rules define a traffic light violation as a red signal prohibiting vehicular traffic from proceeding beyond the stop line on the roadway supplied with the signals until the green signal is shown. Therefore, violations of traffic rules, such as speeding and running red lights, are one of the possible causes of road crashes and the injuries that occur from them.

Road crashes and traffic light violations are leading causes of crashes, deaths, and injuries at signalized junctions in most reports. As a result, there is a need to develop a prototype of a traffic light control system using HSV color segmentation because they are significant following sought information to overcome the above issues.

2.1 Color Segmentation

Color segmentation is breaking a digital image into many parts (segments) of areas that do not collide with each other (non-overlapping). The resultant area of the segmentation is a group of neighboring or related pixels. The color segmentation process is carried out with a regional approach that analyses each pixel's color value and then divides the image according to the desired features. This study uses color segmentation by converting the image color space from RGB to HSV by using a boundary approach. The tolerance value for each HSV color dimension is determined to form segments according to the desired color. The next stage calculates the threshold value based on a predetermined tolerance value [8].

2.2 Red, Green and Blue (RGB)

RGB is a color space that defines colors based on three color channels: red, green, and blue. RGB image model is shown as Fig. 1. These three colors are called primary colors because a mixture of these three colors can produce other colors. Young (1802) confirm this by stating that mixing C1, C2, and C3 with a certain composition can produce any color. When the color image is digitized, three filters are used to extract the intensity of the red, green, and blue colors, and these three combined, we obtain color perception. This color space is generally used to display images or images in electronic devices such as televisions, computers, digital cameras, and scanners. RGB color model is represented by the value of each component, which is between 0 to 255, so that each component consists of 256 levels. When combined, 256 x 256 x 256 or 16,777,216 RGB color combinations can be formed [8].

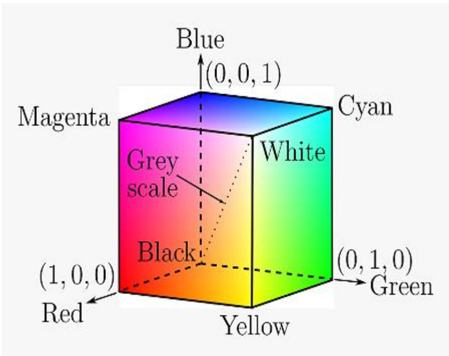


Figure 1: RGB color model cube

2.3 Hue Saturation Value (HSV)

The HSV color model is one of the image color processing models that defines color in terms of Hue, Saturation, and Value, as shown in Fig. 2. Hue states the level of color purity with a value range of 0 – 360 degrees. Hue is used to distinguishing various colors and determine the light red, green, etc. Saturation (sometimes called Chroma) expresses the color saturation and indicates how much white is added to a color. Value is a color brightness level with a 0-100%. Table 1 provides the composition of the color values of each component of Hue, Saturation, and Value component [8].

A color with a value of 100% looks very bright. Conversely, a color with a value of 0 appears very dark. The HSV color space is based on cylindrical coordinates. The HSV color space has a color range of 0 to 1, which can be obtained from the RGB transformation using the transformation formula as shown in equations (1) to (5) [8].

$$r = \frac{R}{(R+G+B)}, g = \frac{G}{(R+G+B)}, b = \frac{B}{(R+G+B)} \quad \textit{Eq. 1}$$

$$V = \max(r, g, b) \quad \textit{Eq. 2}$$

$$S = \begin{cases} 0, \text{ if } V=0 \\ \frac{\max(r,g,b)-\min(r,g,b)}{\max(r,g,b)}, V > 0 \end{cases} \quad \textit{Eq. 3}$$

$$H = \begin{cases} 60 * \left(\begin{matrix} 0, \text{ if } S=0 \\ \frac{(g-b)}{\max(r,g,b)-\min(r,g,b)} \end{matrix} \right), \text{ if } V = r \\ 120 + 60 * \left(\frac{(b-r)}{\max(r,b,g)-\min(r,g,b)} \right), \text{ if } V = g \\ 240 + 60 * \left(\frac{(r-g)}{\max(r,g,b)-\min(r,g,b)} \right), \text{ if } V = b \end{cases} \quad \textit{Eq. 4}$$

$H = H + 360$, if $H < 0$ [9] *Eq. 5*

Where:

R : Pixel value red

G : Pixel value green

B : Pixel value blue

H : Pixel value hue

S : Pixel value saturation

V : Pixel value

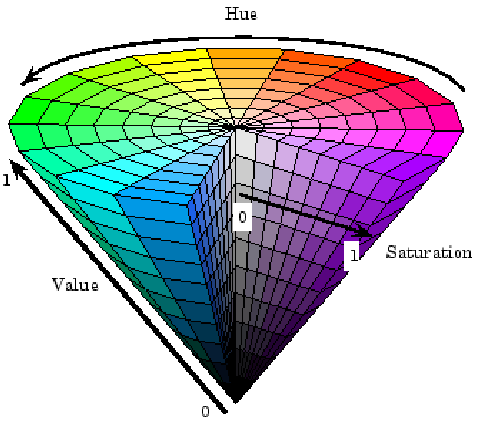


Figure 2: HSV color model

Table 1: HSV color values

No.	Color Name	Hue (°)	Saturation (%)	Value (%)
1.	White	0	0	0
2.	Black	0	0	100
3.	Red	0	100	100
4.	Yellow	60	100	100
5.	Green	120	100	100
6.	Blue	240	100	100
7.	Purple	300	100	100

3. Traffic Light System

Fig. 3 displayed detail of the system's flow in this project. Besides that, a block diagram in Fig. 4 shows the installation system. Based on the block diagram in Fig. 4, the system works are as follows:

(i) The USB camera connects to Raspberry Pi through USB serial port. Then it's mounted on top of the servo motor to rotate to the intersection N.
(ii) The USB camera captures the condition of the highway on the intersection N as input data.
(iii) Raspberry Pi processes the data.
(iv) Raspberry Pi process digital images of the condition of each intersection and compare them to get the results.
(v) Furthermore, the Raspberry Pi divides time for the green light according to traffic conditions.
(vi) LED as a traffic light display on the road N.

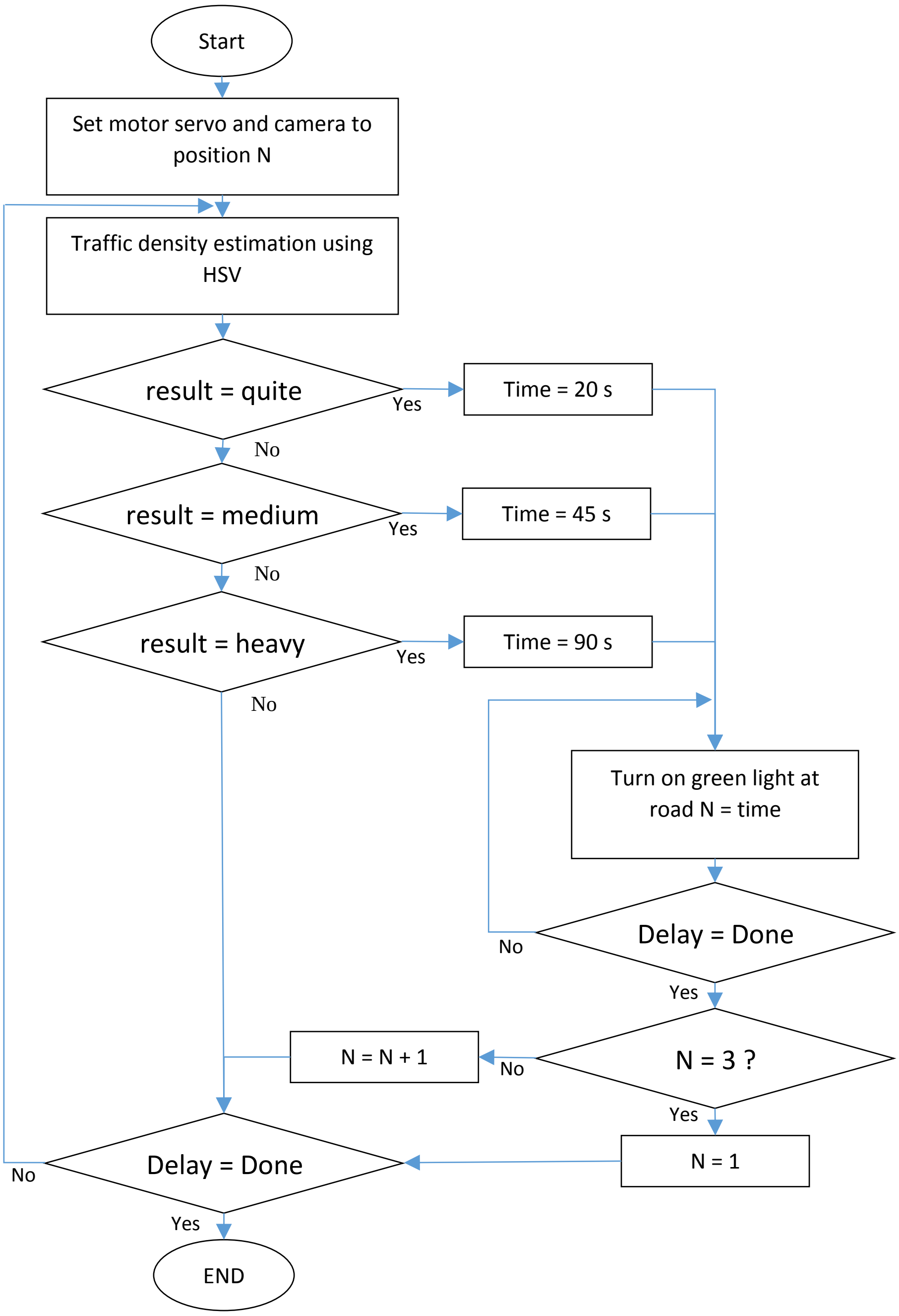


Figure 3: The flowchart of the system

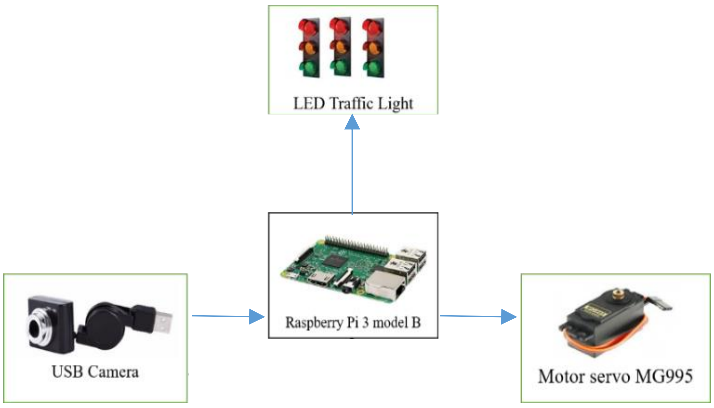


Figure 4: Block diagram of device used for this project

The prototype for the three-way junction has been made from plywood, and the ratio used is: 100 cm squared. The width of the road is 20 cm. The road length is 20 cm, and the size of each car is 3.5 cm x 3.5 cm, as in Fig. 5.

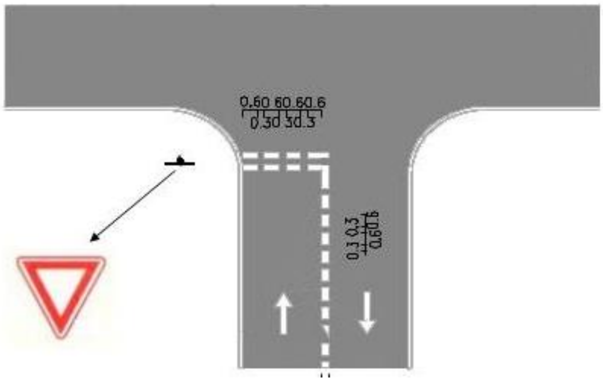


Figure 5: Prototype of highway

4. Result and Discussions

The Raspberry Pi 3 Model B+ has been connected with many peripherals, including a USB web camera, LCD monitor, motor servo, and LED traffic light for each road. Fig. 6 shows the developed prototype of the proposed system. This system uses two sources of power supply.

i. The main power supply source is 220 V, powered by an adapter device connected via GPIO pins 2 and 4 for 5 VDC power.
ii. The other connects through a micro-USB power input with 5 VDC power.

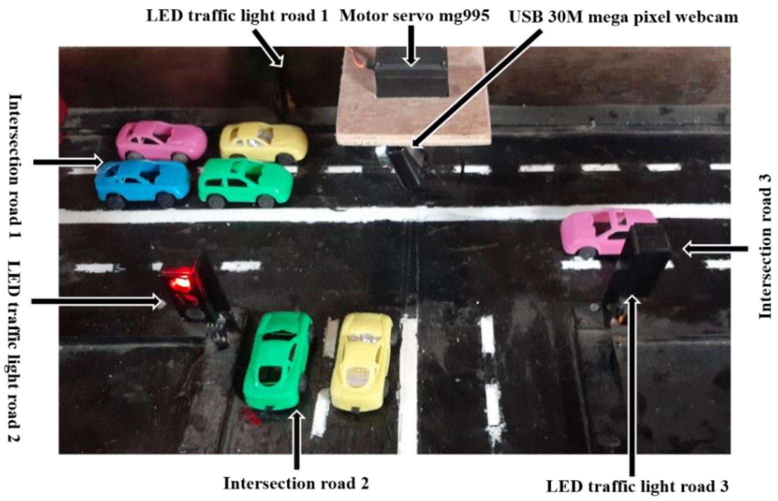


Figure 6: (a) Develop prototype from surface view

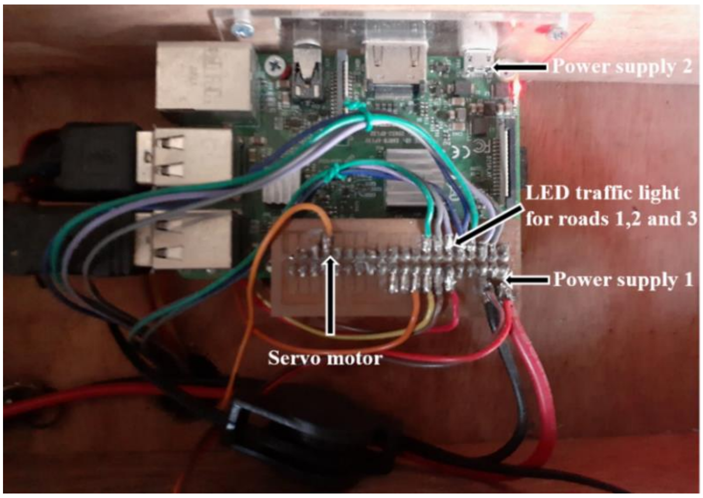


Figure 6: (b) Develop prototype from connection view

4.1 Upper and Lower HSV Range

The HSV color detection algorithm works by a predefining upper and lower range of the HSV value. To filter out any color for detection, it needs to set proper color HSV range. Fig. 7 shows an adjustable HSV trackbar to the upper and lower HSV range predefine.

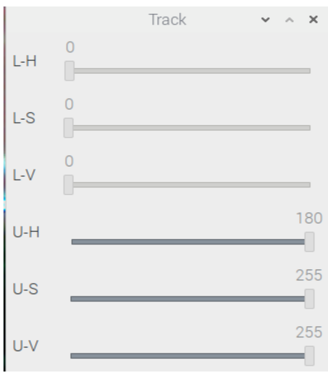


Figure 7: HSV trackbar

In this project, 4 miniature cars in color (blue, green, pink and yellow) have been used to pull out the upper and lower HSV color range. The details are shown in Table 2.

Table 2: HSV color value

No	Car color	Lower Hue	Lower Saturation	Lower Value	Upper Hue	Upper Saturation	Upper Value
1	Blue	81	63	61	180	255	255
2	Green	16	97	87	101	253	255
3	Pink	133	121	89	180	255	255
4	Yellow	11	53	168	90	228	255

4.2 Car Color Target Detection using HSV

The examination hardware for car color detection using HSV is located in the room with a light intensity range from 230 – 246 lux measured using Lux Meter. Besides that, the analysis uses four different colors (Blue, Green, Pink and Yellow) to target or filter out a specific color. Fig. 8 shows the target car color detection in Pink. Meanwhile, Fig. 9 and Fig. 10 show the color mask for car color target detection and at Terminal.

Figure 8: Detection of pink color car

Figure 9: Corresponding color mask

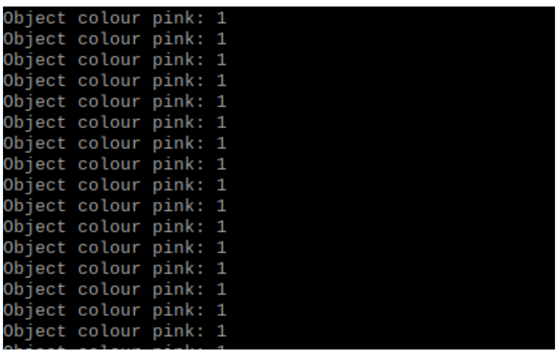


Figure 10: Result of color detection

4.3 Traffic Light Control Based on Traffic Density

The traffic density measured in the developed prototype consists of three conditions heavy traffic, medium traffic and slow traffic. Heavy traffic uses four different colors, medium traffic with two colors, and low traffic can be one color or none. Table 3 provide the number of cars that state traffic condition.

Table 3: Traffic density

No	Total cars	Traffic condition	Time for the green light
1	4	Heavy	90 seconds
2	2	Medium	45 seconds
3	1	Low	20 seconds
4	0	Low	20 seconds

4.4 Traffic Density Estimations using Multiple Color Detection

Detecting multiple colors at once in real-time using an HSV contour object appear to give an unstable result because one car color emerges to give more than one contour. Table 4 shows result number of contours for multiple objects.

Table 4: Result number of contour object

No	Number of car	Number of contours	Traffic condition
1	Empty	0	Low
2	1	1 – 4	Low
3	2	5 – 7	Medium
4	3	8 – 9	Heavy
5	4	9 – 12	Heavy

Therefore, this method has been proposed to estimate the traffic density in the developed prototype of *Traffic condition=Total number of contour*.

4.5 Traffic Density Estimations using Multiple Color Detection

Detecting multiple colors at once in real-time using an HSV contour object appears to be unstable because one particular car color emerges to give more than one contour.

The result for contour objects in multiple colors detection is shown in Fig. 11, while the result at Terminal is shown in Fig. 12.

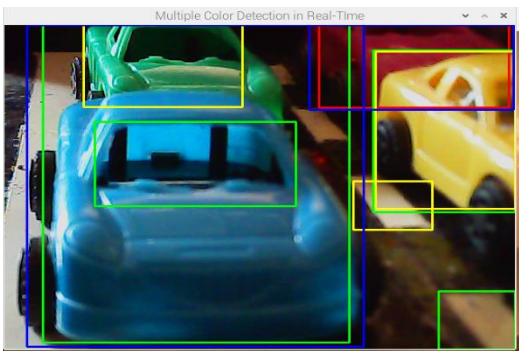
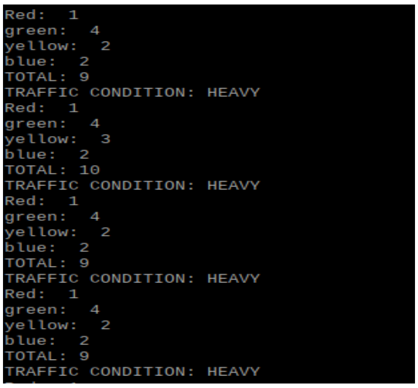


Figure 11: The result in real-time video

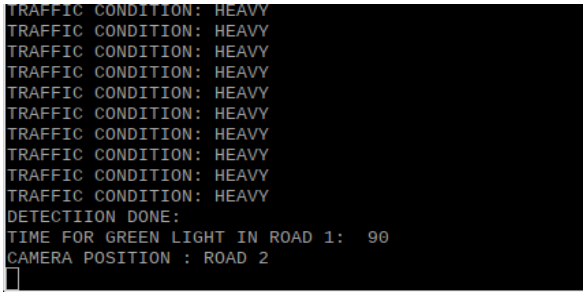


```
Red:  1
green:  4
yellow:  2
blue:  2
TOTAL: 9
TRAFFIC CONDITION: HEAVY
Red:  1
green:  4
yellow:  3
blue:  2
TOTAL: 10
TRAFFIC CONDITION: HEAVY
Red:  1
green:  4
yellow:  2
blue:  2
TOTAL: 9
TRAFFIC CONDITION: HEAVY
Red:  1
green:  4
yellow:  2
blue:  2
TOTAL: 9
TRAFFIC CONDITION: HEAVY
```

Figure 12: The result in terminal for contour object detection

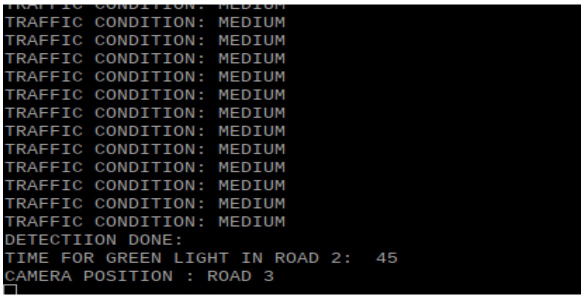
4.6 Result Detection of Traffic Condition in Terminal

The result of the traffic condition detection and the time for green light based on traffic conditions has been displayed at the Raspbian Terminal as shown in Fig. 13 until 15. However, this developed prototype is a real-time based detection; it keeps updating the detection until the final condition, which is considered a traffic condition that becomes input data.



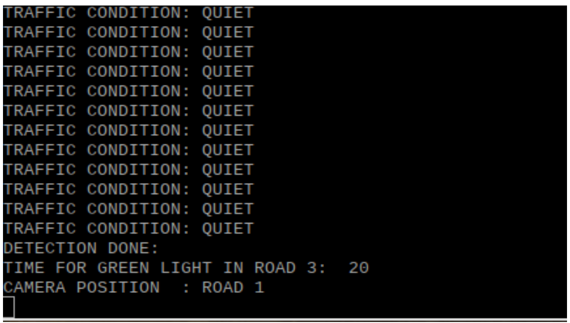
```
TRAFFIC CONDITION: HEAVY
TRAFFIC CONDITION: HEAVY
TRAFFIC CONDITION: HEAVY
TRAFFIC CONDITION: HEAVY
TRAFFIC CONDITION: HEAVY
TRAFFIC CONDITION: HEAVY
TRAFFIC CONDITION: HEAVY
TRAFFIC CONDITION: HEAVY
TRAFFIC CONDITION: HEAVY
TRAFFIC CONDITION: HEAVY
DETECTIION DONE:
TIME FOR GREEN LIGHT IN ROAD 1:  90
CAMERA POSITION : ROAD 2
```

Figure 13: Traffic condition detection road 1



```
TRAFFIC CONDITION: MEDIUM
TRAFFIC CONDITION: MEDIUM
TRAFFIC CONDITION: MEDIUM
TRAFFIC CONDITION: MEDIUM
TRAFFIC CONDITION: MEDIUM
TRAFFIC CONDITION: MEDIUM
TRAFFIC CONDITION: MEDIUM
TRAFFIC CONDITION: MEDIUM
TRAFFIC CONDITION: MEDIUM
TRAFFIC CONDITION: MEDIUM
TRAFFIC CONDITION: MEDIUM
TRAFFIC CONDITION: MEDIUM
DETECTIION DONE:
TIME FOR GREEN LIGHT IN ROAD 2:  45
CAMERA POSITION : ROAD 3
```

Figure 14: Traffic condition detection road 2



```
TRAFFIC CONDITION: QUIET
TRAFFIC CONDITION: QUIET
TRAFFIC CONDITION: QUIET
TRAFFIC CONDITION: QUIET
TRAFFIC CONDITION: QUIET
TRAFFIC CONDITION: QUIET
TRAFFIC CONDITION: QUIET
TRAFFIC CONDITION: QUIET
TRAFFIC CONDITION: QUIET
TRAFFIC CONDITION: QUIET
TRAFFIC CONDITION: QUIET
TRAFFIC CONDITION: QUIET
DETECTION DONE:
TIME FOR GREEN LIGHT IN ROAD 3:  20
CAMERA POSITION  : ROAD 1
```

Figure 15: Traffic condition detection road 3

5. Conclusion

Color is a unique characteristic of an object other than shape, line, form etc. Color is very eye-catching. Many algorithms are used for traffic density estimation from deep learning, machine learning and image processing. An image processing method based on color detection is proposed for this project. OpenCV is used to filter out target color from a video frame. HSV color segmentation is one of the color-based image processing; it works by giving a predefined upper and lower HSV range, which is crucial for the detection's precision. This study showed that using HSV color segmentation for multiple color detection with the provided number of upper and lower for every four colors (blue, green, pink and yellow) can filter out a targeting color from a video frame. It is also able to estimate traffic density and classify traffic conditions based on the number of vehicles on the road.

References

[1] Fibriliyanti, Y., Faradila, L.R., Taqwa, A.: Implementasi pengolahan citra dengan metode Histogram of oriented gradient (HOG) untuk pengatur waktu pada traffic light berdasarkan deteksi kepadatan kendaraan. Seminar Nasional Teknologi dan Informatika, pp. 403–412, 2017.

[2] Hidayati, Q.: Kendali Lampu Lalu Lintas dengan Deteksi Kendaraan Menggunakan Metode Blob Detection. Jurnal Nasional Teknik Elektro dan Tekologi Informasi, 6(2), pp. 215 - 221, 2017.

[3] Djavendra, G.L., Aisyah, S., Jamzuri, E.R.: Desain sistem pengatur lampu lalu lintas dengan identifikasi kepadatan kendaraan menggunakan metode subtraction. Jurnal Nasional Teknik Elektro, 7(2), 2018. https://doi.org/10.25077/jnte.v7n2.541.2018

[4] Cameron, K., Islam, S.: Multiple Objects Detection using HSV. NAECON 2018 - IEEE Natl. Aerosp. Electron. Conf., pp. 270–273, 2018.

[5] Razalli, H., Ramli, R., Alkawaz, M.H.: Emergency Vehicle Recognition and Classification Method Using HSV Color Segmentation. IEEE Int. Colloq. Signal Process. its Appl., pp. 28–29, 2020.

[6] Rahmadani, T., Dyan Nugroho, W., Triwiyatno, A.: Pendeteksian objek menggunakan algoritma HSV berbasis raspberry pi 3B. TRANSIENT, 7(2), pp. 341-344, 2018. https://doi.org/10.14710/transient.7.2.341-344

[7] Hendry: Perancangan Prototype Penetas Telur Ayam Otomatis Berbasis Teknologi IoT. J. Ilmiah Core IT, 8(5), pp. 25–27, 2020.

[8] Nayif, A.N.M., Magdalena, R., Susatio, E.: Aplikasi Pengenal Warna untuk Penderita Buta Warna Menggunakan Pengolahan Model Warna HSV. e-Proceeding Eng., vol. 7, no. 1, pp. 586–593, 2020.

[9] Ayuningsih, K., Sari, Y.A., Adikara, P.P.: Klasifikasi Citra Makanan Menggunakan HSV Color Moment dan Local Binary Pattern dengan Naïve Bayes Classifier. Pengemb. Teknol. Inf. dan Ilmu Komput., vol. 3, no. 4, pp. 3166–3173, 2019.

[10] Rahmadina, F.: Sistem informasi kepadatan lalu lintas berbasis raspberry pi pc board. J. Nas. Tek. Elektro, vol. 5, no. 1, pp. 1–5, 2016.