

Prototype Self-Adaptive Traffic Light Control System using Cameras

Channareth Srun^{a,*}, Saran Meas^b, Sok Oeun Un^c, Saokun Khim^d, Virbora Ny^e

^aFaculty of Electronics, National Polytechnic Institute of Cambodia. Email: nareth16npic@gmail.com

^bFaculty of Electronics, National Polytechnic Institute of Cambodia.

^cFaculty of Electronics, National Polytechnic Institute of Cambodia.

^dFaculty of Electronics, National Polytechnic Institute of Cambodia.

^eFaculty of Electronics, National Polytechnic Institute of Cambodia.

Abstract

Cambodia is a developing country with rapid development. More vehicle usage on the road is increased, this leads to more traffic problems. The traditional traffic light and traffic police officer help solve the problem at a level and limited conditions. For better improvement in this research, the smart traffic light is a better solution that can adapt to the real-world environment. There are four cameras are used for capturing and analyzing the vehicle traffic to control the modes of the traffic light. This method gives the improvement over the limited time of the traditional traffic light. This smart traffic light system comes with three-term mode conditions including normal mode, camera mode, and night mode.

Keywords: Camera; self-adaptive; smart traffic; traffic light

1. Introduction

Along with the rapid development in Cambodia, traffic has increased rapidly. This led to traffic problems in the country such as traffic jams and traffic accidents. These reasons are due to the traditional timing-based traffic light system that is based on the fixed timed control sequence. Adaptive Traffic Control System (ATCS) is a better technique than the traditional preprogrammed time delay traffic light, this method helps to ensure that the system can adapt to the real traffic situation and reduce the traffic jam by relying on the sensors and cameras [1]. Since the actual traffic volume is not always the same from time to time. According to research, showing that delays during traffic can cause an environmental issue due to the CO_2 emission from a vehicle's engine [2]. To reduce the traffic along with the growth of the population in an urban area, the smart city concept is applied. This concept is to use the smart and automatic control system for monitoring and creating an effective control algorithm for the traffic problem [3]. To solve this problem, the camera and computer vision are used to reduce the traffic jam with adaption to the traffic situation [4, 5]. The camera is the main component for capturing the image for vehicle recognition and gives the command to the traffic light controller. This solution helps improve the traffic light's ability to adapt to the traffic condition along with the adjustable mode to different conditions of use. This is a

system designed as a standalone system for monitoring and controlling the traffic condition from the computer software [6, 7]. In our research, there are three terms mode available for the traffic operation control such as normal mode (embedded with sub-mode 1 to 6), camera mode (embedded with 15 conditions), and night mode. Normal mode operates with the duration set by the traffic controller. The camera operates automatically by analyzing and computing the data for generating the traffic sub-mode command. Night mode is enabled during the night-time with fixed conditions, this mode operates the traffic light with yellow color as a warning for the vehicle operator and pedestrian traveling at night.

2. Image Processing

Image processing is the method of calculating the image and processing the image into another form of data. This method also allows retrieving the data from the image and using it for other purposes [8]. These methods include:

- *Method of Converting RGB Image to Gray Image*

A Pixel was created with four elements including alpha, red, green, and blue shown in Fig. 1.

| ALPHA | | | | RED | | | | GREEN | | | | BLUE | | | | | | | | | | | | | | | | | | | | | | | |
|-------|----|----|----|-----|----|----|----|-------|----|----|----|------|----|----|----|---|----|----|----|---|----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 7 | 31 | 23 | 23 | 6 | 22 | 21 | 20 | 5 | 19 | 18 | 17 | 4 | 16 | 15 | 14 | 3 | 13 | 12 | 11 | 2 | 10 | 9 | 8 | 1 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 0 | 0 | 0 |

Figure 1. Bit of a pixel

*Corresponding author. Tel.: +855-17-592-963
 Phnom Penh, Cambodia, 12000

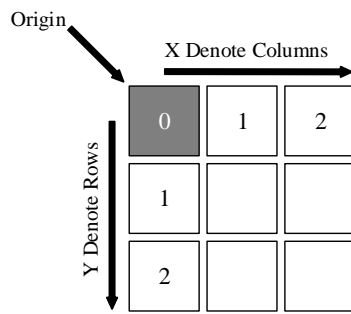


Figure 2. Pixel of an image

The alpha value set the transparency of the three colors red, green, and blue pixel. These values are known as Alpha-A, Red-R, Green-G, and Blue-B [9, 10]. These elements (A, R, G, and B) have the 8 bits value ranging from 0 to 255, in a total of 32 bits for a pixel.

Figure 2 describes the 2D pixel aligned on the column and row of an image. The first pixel is called the origin (starting pixel) of the image with the coordinate of. X and Y is the coordinate of a pixel with A, R, G and B is the value of a pixel.

An example: $P_{0,0}(255,100,150,200)$ indicate the pixel with the coordinate of has the value of $A = 255, R = 100, G = 150, B = 200$.

To convert the (RGB Image) into the (Grayscale Image) the following step is used:

- a) Take the RGB value of a pixel
- b) Calculate the average value of RGB pixel
- c) Change the RGB value of each pixel to A_{vg} value

A pixel of the coordinate at the pixel, the alpha value stays the same with the R, G, and B value is the same [11].

• *Emgu CV*

Emgu CV is a cross-platform image-processing library related to OpenCV since the Emgu CV is a .NET framework connecting to OpenCV. Emgu CV can be used with programming languages such as C#, VB, Iron Python, and C++ [12]. This library is supported on a desktop platform such as Linux, Windows, and macOS X and a mobile platform such as Android, iPhone, iPod Touch, and iPad. Emgu CV was used for replacement on OpenCV since the .NET is not supported on C and C++.

• *Thresholding Theory*

The intensity of a pixel in the background is the value of outer value limited set to the binary image shown in Fig. 3. The Thresholding process is organizing each pixel value ranging between the value of 0 and 1. Hence, the (Binary image) divided the pixel into two parts:

- a) Value 1 is the value of the set boundary
- b) Value 0 is the value of the outer boundary

Most digital cameras are designed to operate with the Thresholding process on the hardware layer. The output data is generated as a binary image. Thresholding is a method of converting from a Grayscale image to the binary image of one or more objects separated from the background. This method is efficient for separating the object from the background image [13, 14].

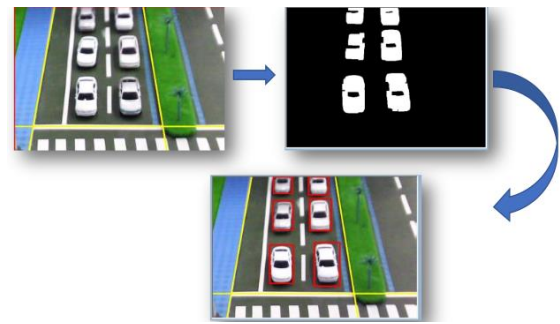


Figure 3. Image thresholding

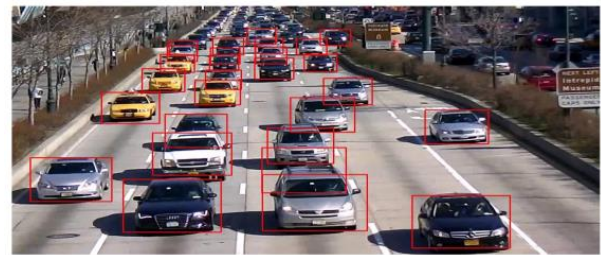


Figure 4. Motion detection principle

• *Motion Detection*

Motion detection is a camera process of detecting an object while the object is moving related to the camera. This technology is divided into two main parts:

- a) Motion detection using software-based motion sensor camera
- b) Motion detection using the sensor in operation (PIR sensor)

In Fig. 4, the software-based motion sensor camera detects the object comparing the changing pixels during changing frame. In case a human or animal moves in a different pixel the object will be detected by the movement [15, 16].

3. Hardware Design and Implementation

Figure 5 is the traffic light model designed as a scale down from the actual traffic light. This model was designed with the dimension of $1 \times 1m$, the height of $370mm$, and the floor thickness of $20mm$. In Fig. 6, the width of each lane is not included in the walking path with the height of the traffic light is $370mm$.

Figure 7 shows the description block diagram of the traffic management system. The input block contains the four cameras as the input to the system. The processor includes the PC as the central processor for image processing and data analysis, Arduino NANO is the embedded controller used for controlling the traffic lights.

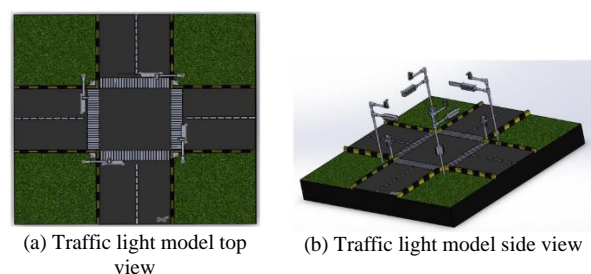


Figure 5. Traffic light model

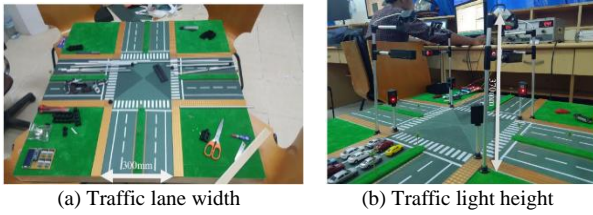


Figure 6. Real-world traffic light model

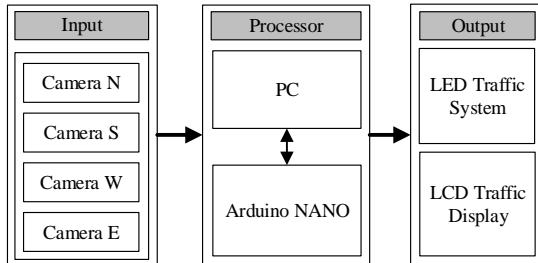


Figure 7. Block diagram of traffic system

The command sent from the computer will activate the operation mode on the traffic light managed by the microcontroller. The serial communication protocol is used for the communication between the computer and the microcontroller. All four cameras are shown in Fig.7 are used as the image capturing sensor communicating with the computer using the USB protocol. These hardware components are assembled as a complete traffic model shown in Fig. 6(b).

4. Software Implementation

The user interface was designed using Microsoft Visual Studio. Figure 8 is interface designed using Khmer text with the security feature before login to the system. Figure 9 is the user interface designed with the control menu on the right side and the four-camera monitoring on the left side.

Figure 10 describes the process of the complete traffic light operating process. The images are captured from the camera are processed through the object detection procedure and determine the percentage of road-free areas. The result of the road free is the input for the system for determining the term and mode of the traffic light. The data will later send to the traffic light hardware controller.



Figure 8. Login user interface

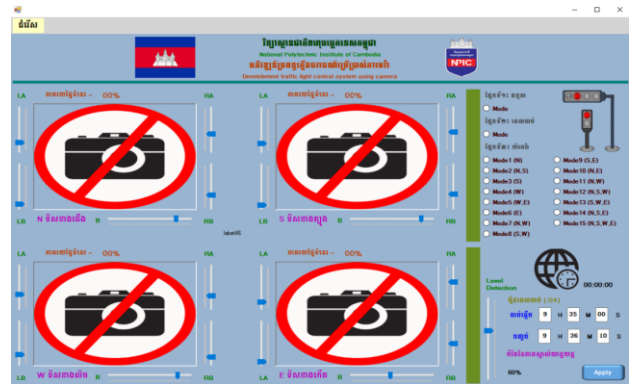


Figure 9. Traffic light management user interface

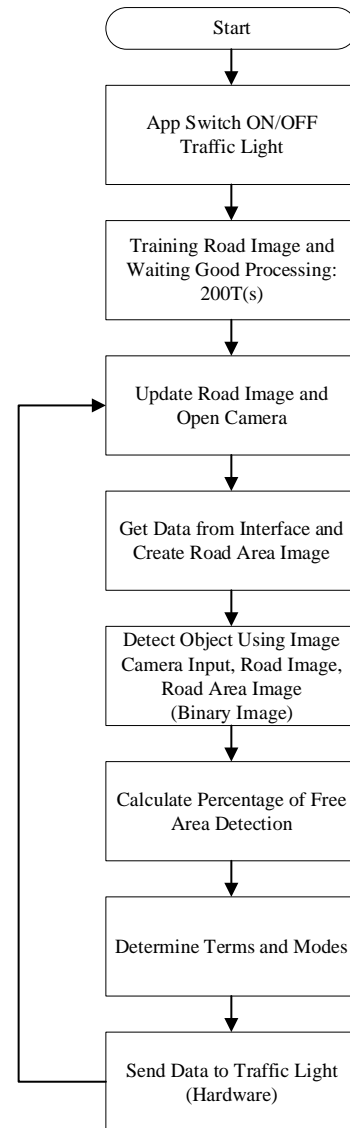


Figure 10. Traffic light system block diagram of a process

The operation modes of the traffic light process are divided into three modes including sub-modes 1 to 6 in normal mode, camera mode, and night mode. Figure 11 shows the detailed decision process made by the traffic light mode through the data from camera image processing.

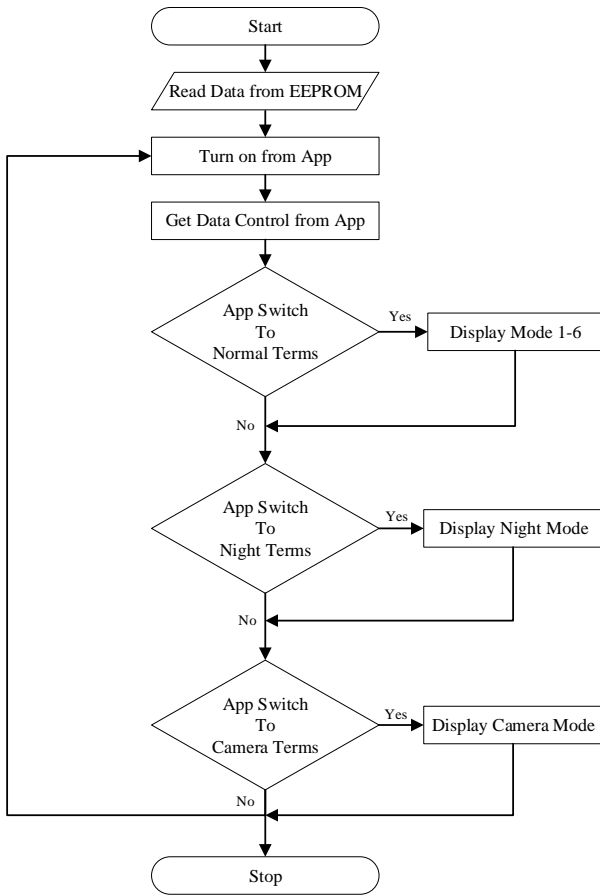


Figure 11. Flowchart of traffic light modes

The system starts the traffic controlling process by the command from the desktop application. The command sent from the desktop application will decide the operation mode of the traffic light. The decision of each operation mode is decided by the data processed from the image processing obtained by the cameras and the traffic control algorithm. The command sent from the traffic management software activates the microcontroller, controlling the colors of the traffic light.

5. Experiment Testing

The experiment process of the traffic light system is divided into two parts:

- Testing the control interface and circuitry of the system
- Testing the responds modes of the system

5.1. Camera Detection Error Calibration

The process of capturing the image via the camera has an error by the distance to the vehicles. This error occurred the further the vehicle to the camera. The result is the smaller size of the vehicle used for analysis. The error detection of the image processing is determined as follows.

Theoretical:

- Full Image (F_A)

$$F_A = 132 \text{ mm} \times 270 \text{ mm}$$

$$= 36828 \text{ mm}^2$$

- Free Detect Area (F_D)

$$F_D = F_A - 2(\text{Length} \times \text{Width})$$

$$= 36828 \text{ mm}^2 - 3250 \text{ mm}^2$$

$$= 33578 \text{ mm}^2$$
- Percentage (P)

$$TP = \frac{33578}{36828} \times 100 = 91.18\%$$

Experimental:

- Full Area (F_A)

$$F_A = 137662$$
- Free Detect Area (F_D)

$$F_D = F_A - \text{Detected}$$

$$= 127622 - 26232$$

$$= 111430$$
- Percentage (P)

$$EP = \frac{111430}{137662} \times 100$$

$$= 80.94\%$$
- % ERROR

$$\% \text{ ERROR} = \frac{E_P - T_P}{T_P} \times 100$$

$$= \frac{80.94 - 91.18}{91.18} \times 100$$

$$= 11.23\%$$
- % ERROR

$$\% \text{ ERROR} = \frac{80.94 - 95}{95} \times 100$$

$$= \mp 14.8\%$$

5.2. Traffic Mode Testing

The traffic mode testing is divided into three modes. The first mode, Figure 12 is the normal traffic mode. In this mode, the road was detected with 100% empty road. The algorithm sends the command to the traffic light as Green.

In night mode, the time is limited to the set value. In this experiment, the time is set to the range of 00:00 (Midnight) to 6:00 AM. During this mode, the algorithm sends the command to the traffic light as Yellow shown in Fig. 13.

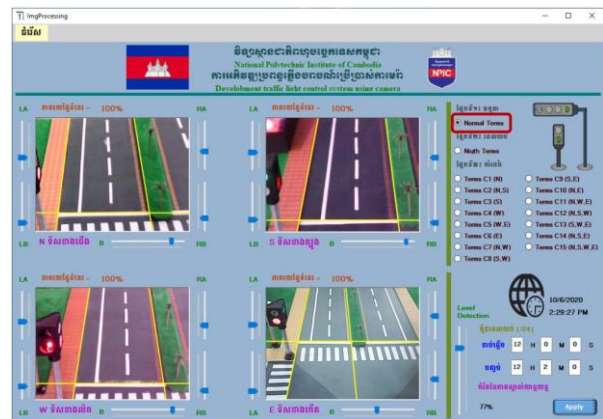


Figure 12. Traffic light "Normal mode" operation



Figure 13. Traffic light "Night mode" operation



Figure 14. Traffic light "Traffic mode" operation

In traffic mode, in testing mode 1. The north side has the detected vehicle on the road. The algorithm is set to at least four cars is defined as a signal to activate the traffic mode algorithm. In Fig 14, the north camera detected 69.40% of free space with five cars detected and the other three lanes available. The algorithm sends the green signal to the other three empty lanes as Green.

5.3. Experimental Result

The testing result is shown in Table 1. This table is divided into terms, camera lane detection, and traffic sub-mode selection. Notably, the terms of the traffic may vary due to the traffic conditions. The algorithm decides the selected mode via the information from the camera's detection. The camera lane detection determines the cardinal traffic conditions. The detection results are divided into two blocks with free space and used space on each lane. The algorithm sends the control signal to the traffic light controller based on the sub-mode of the traffic operation shown from Figs. 15 to 20.

For testing purposes, the time of operation in each mode is 15 seconds with green light. After the main mode timer is completed, the traffic light switches back to the yellow light for warning mode with the blink interval of 0.5s.

Table 1. Traffic mode experimental results

| Term Camera | N | | S | | W | | E | | Sub mode |
|----------------|------------|------------|------------|------------|------------|------------|------------|------------|-------------|
| | F/S (%) | U/S (%) | F/S (%) | U/S (%) | F/S (%) | U/S (%) | F/S (%) | U/S (%) | |
| C1 | 69.40 | 30.60 | 100 | 0 | 100 | 0 | 100 | 0 | 1 |
| C2 | 73.46 | 26.54 | 77.49 | 22.49 | 93.20 | 6.80 | 95.43 | 4.57 | 2 |
| C3 | 73.55 | 26.45 | 69.84 | 30.16 | 100 | 0 | 100 | 0 | 3 |
| C4 | 92.54 | 7.46 | 94.51 | 5.49 | 77.72 | 22.28 | 94.48 | 5.52 | 4 |
| C5 | 92.20 | 7.80 | 94.56 | 5.44 | 73.10 | 26.90 | 78.50 | 21.50 | 5 |
| C6 | 94.25 | 5.75 | 95.51 | 4.49 | 90.99 | 9.01 | 73.86 | 26.14 | 6 |
| C7 | 79.10 | 20.90 | 95.25 | 4.75 | 68.05 | 31.95 | 95.26 | 4.74 | 4 |
| C8 | 94.26 | 5.74 | 82.95 | 17.05 | 71.96 | 28.04 | 94.31 | 5.69 | 4 |
| C9 | 94.25 | 5.75 | 80.53 | 19.47 | 89.30 | 10.70 | 77.73 | 22.27 | 6 |
| C10 | 78.78 | 21.22 | 93.78 | 6.22 | 88.75 | 11.25 | 71.88 | 28.12 | 6 |
| C11 | 68.28 | 31.72 | 100 | 0 | 71.00 | 29.00 | 72.17 | 27.83 | 1 |
| C12 | 69.22 | 30.78 | 63.36 | 36.64 | 72.27 | 27.73 | 96.04 | 3.96 | 3 |
| C13 | 93.29 | 6.71 | 72.93 | 27.07 | 69.71 | 30.29 | 74.62 | 25.38 | 4 |
| C14 | 71.97 | 28.23 | 72.24 | 27.76 | 86.57 | 13.43 | 73.41 | 26.59 | 1 |
| C15 | 70.16 | 29.84 | 72.06 | 27.94 | 70.16 | 29.84 | 72.71 | 27.29 | 1 |

*F/S: Free Space, U/S: Used Space.

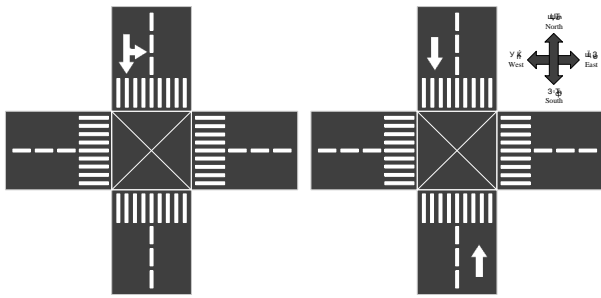


Figure 15. Traffic operation sub-mode 1 and sub-mode 2

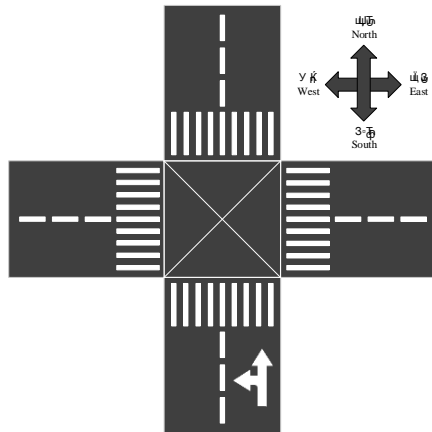


Figure 16. Traffic operation sub-mode 3

Figure 15 describes the operation process of traffic sub-mode 1 and sub-mode 2 of the camera detection sequence.

Sub-mode 1: This mode allows the vehicle to pass from the north lane to the south lane and turn to the left and right.

Sub-mode 2: This mode allows the vehicle to pass from the south lane to the north lane. The operation time is the same as the mode one operation.

Sub-mode 3: This mode allows the vehicle to pass through from the south to the north and able to turn left and right shown in Fig. 16.

Sub-mode 4: This mode allows the vehicle to pass through from the west to the east and able to turn left and right shown in Fig. 17.

Sub-mode 5: this mode allows the vehicle to pass from the east to the west lane, and able to pass from the west to the east lane shown in Fig. 18.

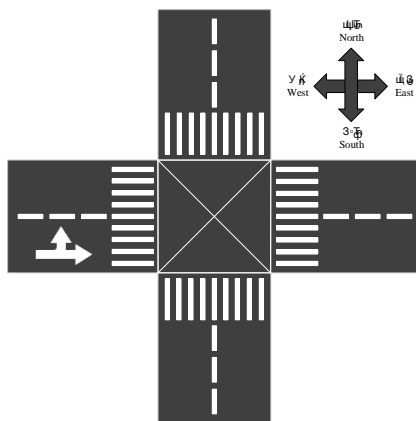


Figure 17. Traffic operation sub-mode 4

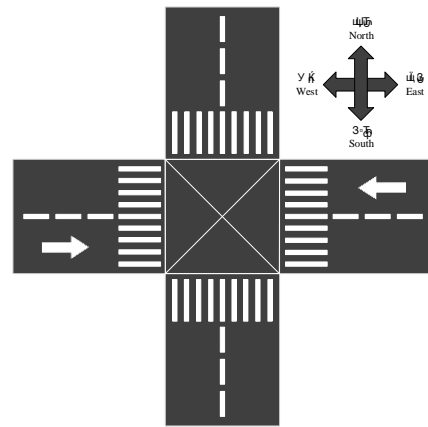


Figure 18. Traffic operation sub-mode 5

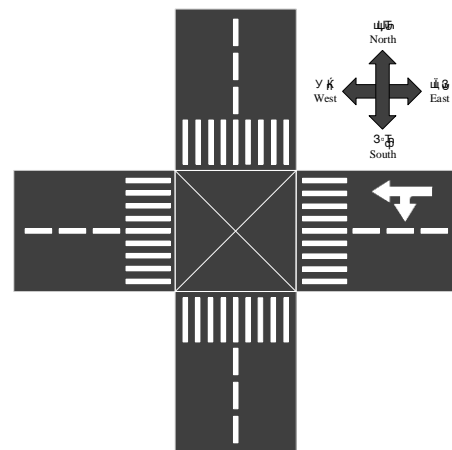


Figure 19. Traffic operation sub-mode 6

Sub-mode 6: this mode allows the vehicle to pass from the east to the west lane, and able to turn to the left and right lane shown in Fig. 19.

Figure 20 shows the possible turns for each lane at night mode. This mode is enabled due to the less traffic at nighttime the night mode activation is set from 11:00 PM to 4:00 AM. In this mode, the camera detections are disabled and only enable for yellow light as a warning for pedestrians.

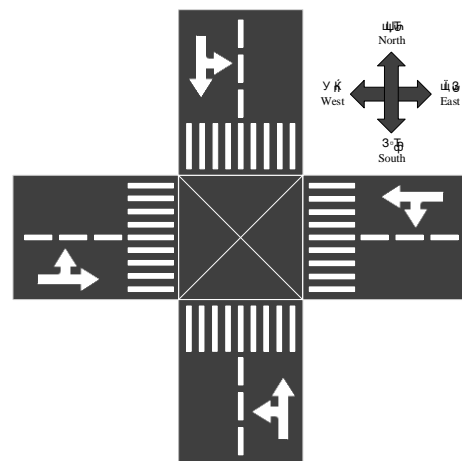


Figure 20. Traffic operation in "Night mode"

6. Conclusion

The process of controlling the traffic light conditions can be achieved by analyzing the data from the input cameras. The modes of the traffic operation can be controlled from the desktop application. The user can switch between the modes manually and the command will be processed by the traffic control algorithm controlling the traffic operation. The percentage of vehicle detection is determined as a percentage value. In the experiment, the camera with the vehicle detection, the system has an $\pm 14.8\%$ error rate. The detection error is caused by the shadows of the vehicle due to the small size of the traffic light model and the position of the camera. For further practice, applying this system to the actual traffic application can help reduce the error rate.

References

- [1] S. Shankaran and L. Rajendran, "Real-Time Adaptive Traffic Control System for Smart Cities," in *International Conference on Computer Communication and Informatics (ICCCI-2021)*, 2021, pp. 27–29.
- [2] K. Cikhardtova, Z. Belinova, T. Tichy, and J. Ruzicka, "Evaluation of Traffic Control Impact on Smart Cities Environment," in *Smart Cities Symposium Prague*, 2016, pp. 1–4.
- [3] L. F. P. de Oliveira, L. T. Manera, and P. D. G. da Luz, "Development of a Smart Traffic Light Control System With Real-Time Monitoring," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3384–3393, 2021.
- [4] M. Razavi, M. Hamidkhani, and R. Sadeghi, "Smart Traffic Light Scheduling in Smart City Using Image and Video Processing," in *2019 3rd International Conference on Internet of Things and Applications (IoT)*, 2019, pp. 1–4.
- [5] K. Iwaoka, M. Maruyama, and S. Nukura, "Public Transportation Priority Systems Combined with Multiple Traffic Control Strategies," in *IEEE Intelligent Transportation Systems.*, 2000, pp. 173–177.
- [6] M. F. Rachmadi *et al.*, "Adaptive Traffic Signal Control System Using Camera Sensor and Embedded System," in *TENCON 2011 - 2011 IEEE Region 10 Conference*, 2011, pp. 1261–1265.
- [7] P. Rizwan, K. Suresh, and M. R. Babu, "Real-time Smart Traffic Management System for Smart Cities by Using Internet of Things and Big Data," in *2016 International Conference on Emerging Technological Trends (ICETT)*, 2016, pp. 1–7.
- [8] M. M. Hasan, G. Saha, A. Hoque, and M. B. Majumder, "Smart Traffic Control System with Application of Image Processing Techniques," in *2014 International Conference on Informatics, Electronics Vision (ICIEV)*, 2014, pp. 1–4.
- [9] M. Sinecen, *Applications from Engineering with MATLAB Concepts*. 2016.
- [10] S. Annadurai, *Fundamentals of Digital Image Processing*. India: Pearson, 2006.
- [11] Z. Li, "Introduction to Computer Vision," 2020.
- [12] S. Shi, *Emgu CV Essential*. Birmingham: Packt Publishing Ltd, 2012.
- [13] M. V. K. S. C. Thavreak, *Machine Vision Inspection System*. Phnom Penh, 2019.
- [14] R. Holzer, "OpenCV Tutorial Documentation," pp. 61–68, 2019.
- [15] R. Jain, R. Kasturi, and B. G. Schunck, *Machine vision*. McGraw-Hill, 1995.
- [16] Z. Yu and Y. Chen, "A Real-Time Motion Detection Algorithm for Traffic Monitoring Systems Based on Consecutive Temporal Difference," in *2009 7th Asian Control Conference*, 2009, pp. 1594–1599.