Real-Time Traffic Flow Controls at Signalised Intersections: A Case Study of the Said-Jawda intersection in Karbala

Raid R. A. Almuhanna¹, Shahed A. Salman²
¹ Faculty member, Civil engineering Department, University of Kerbala, Iraq.
² Postgraduate Student, Civil engineering Department, University of Kerbala, Iraq.
E-mail: raids@uokerbala.edu.iq, shahed20126@gmail.com

Abstract. As the problem of urban traffic congestion increases, the need for advanced technology and equipment to be introduced for the improvement of traffic control also increases. Traffic problems have worsened last years due to significant increases in the number of vehicles on the roads and the limited availability of resources in the current infrastructure. The easiest way to control traffic lights is to use timer for each phase; however, another way is to use electronic sensors to detect vehicles to supply the required traffic signals for a given cycle. To enable effective traffic flow through a signalised intersection, this study proposes a system for controlling traffic lights that detects vehicles using videos and imaging techniques instead of electronic sensors embedded in the pavement. Four cameras were installed at a height of three metres on the four approaches to a set of traffic lights. The captured images were then analysed using digital image processing based on Python to facilitate vehicle detection and traffic condition analysis. The work is thus divided into two segments: data collection and traffic control solution generation. The results indicated that using the proposed automated system could produce more effective traffic control as compared to SIDRA fixed-time controls, with delay reduced within the range 40 to 90% and cycle length reduced by 30 to 65%.

Keywords: Intelligent Transportation System (ITS), Traffic control system, Image Processing, Python, Optimal Cycle Length.

1. Introduction

Signalised intersections play a vital role in the transportation system, particularly in terms of urban roads. Vehicular delays during red lights are controlled by the signal systems used to regulate the intersection during traffic movement, however, and both the speed and cost-effectiveness of the whole network can be decreased as a result of poorly controlled intersections. Vehicular delays also lead to increases in total travel time [1].

The current work focused on reducing this delay by optimising signal cycle length. The problem of sensing the number of vehicles on an approach was solved by video capture using an online camera, with the videos pre-processed by transformation into a collection of frames. Each frame was then converted to a binary image to calculate the contours of vehicles, and the number of vehicles were obtained by counting the number of contours on each approach. The calculation of green light time then depended on the number of vehicles on each approach of the intersection [2]. Object detection was obtained also by using a background subtraction model, which is more accurate and has better results in terms of detecting moving vehicles than other available models. Each vehicle thus becomes represented as a blob in the resulting image, obtained by subtracting the background and the current image, and a blob analysis method can be used to count the number of vehicles in each approach [3].

2. Background

Various researchers have dealt with image processing to address traffic problems. Farooq et. al. [4] proposed a system with a moving motor running a camera to capture traffic volume. This system offered...
the capacity to vary green light time depending on the real-time traffic volume rather than utilising a fixed time signal.

Choudkar et. al. [5] proposed an image processing system to control traffic lights. Vehicles are detected using images instead of embedded electrical sensors, with an edge detection process applied after image matching to control the duration of traffic lights.

Nicole et al. [6] worked on a smart traffic light system that updated the signal time automatically using a Raspberry Pi microcomputer with an Internet of Things (IoT) system. Sensors, including a Passive Infrared Sensor (PIR), were utilised rather than camera-related functions, however, as PIRs have ultra-low energy consumption.

Singh et al. [7] built a strategy for controlling traffic signals using a genetic evolutionary algorithm. The strategy provided a close match to the optimal traffic flow, with the genetic algorithm obtaining the optimal green light time based on the number of vehicles.

Panda et al. [8] proposed a system for traffic size estimation using image processing techniques. This system was implemented using MATLAB to prohibit traffic on a highway. They thus improved existing traffic control systems with additional intelligence in artificial vision terms, and the system nevertheless retained the ability to give priority to emergency vehicles.

Maheshwari et al [9] proposed a dynamic image processing system to prevent the waste of resources such as fuel and time at intersections. The system processes images from the installed camera to monitor tasks and prevents congestion by setting timers according to its scope of view. The system saves resources and offers a quick turnaround time for each junction. It also has the capacity to apply additional machine learning techniques to handle emergency circumstances in the future.

Jadhav et al. [10] implemented a system in MATLAB to prevent heavy traffic congestion. Images were captured using a web camera placed at congestion points, which were then pre-processed to allow computation of traffic density. Based on this density, commands are then sent by the controller to the traffic signals to manage the traffic.

Eamthanakul et al. [11] proposed a system utilising CCTV cameras and image processing techniques to determine traffic conditions under three situations: flow, heavy, and jammed. The images are pre-processed to analyse the traffic conditions based on detecting the number of vehicles. The system then uses order transportation planning according to the detected condition based on a dataset prepared for the purpose. The system can be used with lights to control intersections.

3. Data collection method

The traffic volume used in this research was obtained from a video record for the period from 8:15 to 11.15 AM, a total length of three hours, at Said-Jawda intersection in Karbala city. Figure 1 shows the intersection layout in Google Maps.

![Figure 1. Said-Jawda intersection.](image-url)
The traffic volume for the intersection is shown in Table 1.

**Table 1. Traffic volume for minor and major streets for Said Jawda Intersection.**

| Time             | Approach          | Traffic volume |
|------------------|-------------------|----------------|
| 8:15 AM - 9:15 AM| East (minor)      | 253            |
|                  | West(minor)       | 317            |
|                  | North (major)     | 298            |
|                  | South (major)     | 254            |
| 9:15 AM - 10:15 AM| East (minor)      | 369            |
|                  | West(minor)       | 305            |
|                  | North (major)     | 332            |
|                  | South (major)     | 336            |
| 10:15 AM -11:15 AM| East (minor)      | 290            |
|                  | West(minor)       | 251            |
|                  | North (major)     | 255            |
|                  | South (major)     | 267            |

4. **Proposed system**

Image measurement can deliver quantitative information on various aspects of traffic status, including vehicle classification by size (vehicles, trunks, motorcycle), average speed (from SIDRA software), and vehicle counts. Calculation of the optimum cycle length requires both hardware and software components, however.

a. **Hardware**

i. **Raspberry Pi 3**

The Raspberry Pi is a micro-computer that can run the LINUX operating system to operate connected cameras to obtain data from the selected intersection and assign the required green light-time by using a Python program to apply variance and slab algorithms. Figure 2 shows the Raspberry Pi used, available in the local market.
Figure 2. Raspberry Pi 3

ii. 3.8-port switch

This switch is used to establish connections between the other components.

iii. RJ45 Cable

iv. Video recording cameras

The camera used has the following specifications:
- 2 Megapixel high-performance CMOS
- Analog HD output with up to 1080p resolution
- Day/Night switch
- Smart IR
- Up to 40 m IR distance

v. Server

The server was used to receive, send, and save information

vi. Fibreoptic Cable

vii. Computer

The computer was used to display the data received from the intersection and to display the data in a php page [12].

b. Software

i. Data centre

The data centre was a web page designed in PHP to display incoming data from the MYSQL database stored in the Raspberry Pi 3.

ii. Application
The application was programmed in Python; Figure 3 shows the steps of the proposed system.

![Figure 3. The structure of the system.](image)

1. **Video acquisition**
   A special camera was used to monitor the intersection in real-time. The camera output was a stream of video frames with a frame rate of 30 frames per second [7]. The length of the road within the camera view was 15 metres.

2. **Video analysis**
   Video frames were binarized by using a background model with a threshold to detect and track all objects (vehicles) [13]. Region growing segmentation was used to segment the images and thus to count the vehicles and their bounding boxes [14]. Figure 4 shows this frame pre-processing.

![Figure 4. Frame before and after pre-processing](image)
3. Data processing algorithms

Variance and slab algorithms were used to process the data in order to calculate the time of the green signal.

a. Variance algorithm

Variance as a statistical representation is implemented to identify the appearance of a combination of numbers. It is defined as the value of the squared displacement from the mean. The variance of the discrete random variable was calculated according to the distribution of the number of vehicles in the four approaches to the intersection. The maximum corresponding time was obtained based on the maximum variance, with each vehicle taking two to three seconds to pass the intersection, as shown in Table 2 [15]:

\[ \sigma^2 = \sum_x (x - \mu)^2 P(x) \]  

(1)

Table 2. Values of variance with corresponding time

| Seq. | Variance  | Time (seconds) |
|------|-----------|---------------|
| 1    | <100      | 25            |
| 2    | 100-200   | 40            |
| 3    | 201-400   | 50            |
| 4    | 401-650   | 60            |

b. Slab algorithm

The slab algorithm is usually a memory locating algorithm used in operating systems. In this paper, the algorithm was used to allocate time instead of allocating memory. It thus consisted of a set of lower and upper bounds that formed slabs that split the numbers of cars to make congested traffic movement more continuous. These were obtained to determine suitable green light times based on real-time traffic flow [16].

\[ Gei = 2 \times \text{number of vehicles in approach} \]  

(2)

where 2 represents the required time in minutes for vehicles to pass the intersection.

Table 3 represents the number of cars with the corresponding time used in the slab algorithm.

Table 3. Number of cars with corresponding time using the slab algorithm.

| Congestion | Number of cars | Time |
|------------|----------------|------|
| 1          | 0-10           | 20   |
| 2          | 11-40          | 40   |
| 3          | 41-60          | 60   |
4. Optimal green light time calculation

Computing the length of a cycle requires understanding the intervals of that cycle. Figure 5 explains the cycle length in a two-phase signal system [17].

The various times, such as green light time, lost time, effective green light time, and amber light time are described in figure 6, which shows the various portions of the green phase.

![Figure 5. Two phase signal system [18]](image-url)
Effective green light time is obtained using equation (3) [18]:

$$G_{ei} = G_{ai} - L_i + T_i$$  \hspace{2cm} (3)$$

where

- $G_{ei}$: effective green light time,
- $G_{ai}$: actual green light time,
- $T_i$: amber light time,
- $L_i$: lost time.

and lost time is obtained as

$$L_i = G_{ai} + T_i - G_{ei}$$  \hspace{2cm} (4)$$

Actual green light time in this case was obtained from the variance or slab algorithm while the lost time was a constant $\approx 3$ seconds, as calculated from the SIDRA intersection application. The time of the amber signal for a specific phase was calculated using equation (5) [17]:

$$T_i = \delta + \frac{W + L}{u_0} + \frac{u_0}{2(a + Gg)}$$  \hspace{2cm} (5)$$

where

- $W$: Width of intersection (ft).
- $L$: Length of vehicle (ft).
- $\delta$: perception-reaction time (sec).
- $a$: constant rate of braking deceleration (ft/sec$^2$).
- $G$: grade.
- $g$: acceleration due to gravity (32.2 ft/sec$^2$) [18].
The average delay in a given lane of a vehicle at the intersection was obtained using equation (6) [17]:

\[
d = \frac{0.3BC(1-g)X^2}{1-(\frac{g}{C})(X)} + 173X^2 \left( (X-1) + \left( \frac{16X}{gC} \right)^{\frac{1}{2}} \right)
\]

(6)

where

- \( D \): average stopped delay per vehicle in a group of vehicles in a lane.
- \( C \): cycle length per second.
- \( g/C \): green ratio for the lane group or lane.
- \( g \): effective green light time for a lane or lane group per second.
- \( X \): lane group V/c ratio.
- \( V \): actual flow rate design for the lane group or the lane.
- \( c \): lane group capacity (pcu/hour).

Cycle length was computed using equation (7) [18]

\[
C_0 = \frac{1.5L + 5}{1 - \sum_{i=1}^{\phi} Y_i}
\]

(7)

where

- \( C_0 \): optimal cycle length.
- \( L \): total lost time during the cycle.
- \( Y_i \): maximum value of ratios of approach flows to saturation flows for all road groups utilising the phase.
- \( \phi \): number of phases.

5. Results

SIDRA is a software package used by traffic design, operations, and planning professionals to assess intersection (junction), network capacity, level of service performance, signalised intersections, and network timing calculations [19]. Ten experiments were used to evaluate the system in this work, with the same number of vehicles and conditions applied to SIDRA in a fixed time period. The proposed system was then run in real-time to obtain the results listed below:

The results of tests 1, 2, 3, 4, 5, 6, 7, 8, 9, and 10 are summarized in Tables 4 to 13, respectively.

| Approach | The proposed system | SIDRA |
|----------|---------------------|-------|
|          | Delay | LOS | Cycle time | Delay | LOS | Cycle time |
| South    | 16.59 | B   |            | 53.8  | D   |            |
| East     | 23.87 | C   |            | 75.4  | E   |            |
| North    | 15.62 | B   | 68.5       | 47.1  | D   |            |
| West     | 14.41 | B   |            | 47.6  | D   | 160         |
| Criteria | Average delay | Average LOS | Average delay | Average LOS |
| Intersection | 10.74 | B | | 45.7 | D |
### Table 5. Test 2 results.

| Approach | The proposed system | SIDRA |
|----------|---------------------|-------|
|          | Delay | LOS | Cycle time | Delay | LOS | Cycle time |
| South    | 13.88 | B   |            | 53.2  | D   |            |
| East     | 14.99 | B   |            | 58.9  | E   |            |
| North    | 18.14 | B   | 74         | 77.3  | E   |            |
| West     | 10.28 | B   |            | 35.7  | D   | 160        |

| Criteria | Average delay | Average LOS |
|----------|---------------|-------------|
| Intersection | 14.32 | B          | 50.4 | D |

### Table 6. Test 3 results.

| Approach | The proposed system | SIDRA |
|----------|---------------------|-------|
|          | Delay | LOS | Cycle time | Delay | LOS | Cycle time |
| South    | 4.71  | A   |            | 42.3  | D   |            |
| East     | No vehicles | No LOS | 77.3 | E   | 160 |
|          | No vehicles | No LOS | 77.3 | E   | 160 |
| West     | 4.54  | A   |            | 34.4  | C   |            |

| Criteria | Average delay | Average LOS |
|----------|---------------|-------------|
| Intersection | 4.63 | A          | 46.7 | D |

### Table 7. Test 4 results.

| Approach | The proposed system | SIDRA |
|----------|---------------------|-------|
|          | Delay | LOS | Cycle time | Delay | LOS | Cycle time |
| South    | 18.34 | B   |            | 48.6  | D   |            |
| East     | 23.37 | C   |            | 79.8  | E   |            |
| North    | 18.94 | B   | 86         | 56.8  | E   |            |
| West     | 10.75 | B   |            | 40.7  | D   | 160        |

| Criteria | Average delay | Average LOS |
|----------|---------------|-------------|
| Intersection | 17.85 | B         | 49.8 | D |

10
## Table 8. Test 5 results.

| Approach | The proposed system | SIDRA |
|----------|---------------------|-------|
|          | Delay   | LOS    | Cycle time | Delay   | LOS    | Cycle time |
| South    | 16.59   | B      | 88         | 53.8    | D      |           |
| East     | 23.87   | C      |            | 75.4    | E      |           |
| North    | 15.62   | B      |            | 47.1    | D      |           |
| West     | 14.41   | B      |            | 47.6    | D      | 160       |
| Criteria | Average delay | Average LOS | | Average delay | Average LOS | |
| Intersection | 17.62   | B      |            | 52.1    | D      | |

## Table 9. Test 6 results.

| Approach | The proposed system | SIDRA |
|----------|---------------------|-------|
|          | Delay   | LOS    | Cycle time | Delay   | LOS    | Cycle time |
| South    | 24.91   | C      |            | 79.8    | E      |           |
| East     | 16.32   | B      | 88         | 46.0    | D      |           |
| North    | 10.51   | B      |            | 29.7    | C      |           |
| West     | 26.84   | C      |            | 79.8    | E      | 160       |
| Criteria | Average delay | Average LOS | | Average delay | Average LOS | |
| Intersection | 19.56   | B      |            | 43.9    | D      | |

## Table 10. Test 7 results.

| Approach | The proposed system | SIDRA |
|----------|---------------------|-------|
|          | Delay   | LOS    | Cycle time | Delay   | LOS    | Cycle time |
| South    | 35.34   | C      |            | 77.8    | E      |           |
| East     | 22.56   | C      |            | 53.6    | D      |           |
| North    | 26.61   | C      | 108        | 53.2    | D      | 160       |
| West     | 16.97   | B      |            | 41.7    | D      |           |
| Criteria | Average delay | Average LOS | | Average delay | Average LOS | |
| Intersection | 25.37   | C      |            | 51.2    | D      | |
### Table 11. Test 8 results.

| Approach | The proposed system | SIDRA |
|----------|---------------------|-------|
|          | Delay | LOS | Cycle time | Delay | LOS | Cycle time |
| South    | No vehicles | No LOS | | 81.2 | F | |
| East     | 12.24 | B | 86.5 | 46.5 | D | |
| North    | 9.01  | A | | 28.6 | C | |
| West     | 25.78 | C | | 81.2 | F | |
| Criteria | Average delay | Average LOS | | Average delay | Average LOS | 160 |
| Intersection | 15.68 | B | | 43.5 | D | |

### Table 12. Test 9 results.

| Approach | The proposed system | SIDRA |
|----------|---------------------|-------|
|          | Delay | LOS | Cycle time | Delay | LOS | Cycle time |
| South    | 29.05 | C | | 69.1 | E | |
| East     | 15.24 | B | | 33.6 | C | |
| North    | 45.47 | C | 110 | 62.2 | E | |
| West     | 26.35 | C | | 62.5 | E | 160 |
| Criteria | Average delay | Average LOS | | Average delay | Average LOS | |
| Intersection | 29.03 | C | | 51.7 | D | |

### Table 13. Test 10 results.

| Approach | The proposed system | SIDRA |
|----------|---------------------|-------|
|          | Delay (Sec.) | LOS | Cycle time | Delay | LOS | Cycle time |
| South    | 28.41 | C | 94 | 82.6 | F | |
| East     | 21.83 | C | | 55.3 | E | |
| North    | 16.72 | B | | 39.2 | D | |
| West     | 19.24 | B | | 48.7 | D | 160 |
| Criteria | Average delay | Average LOS | | Average delay | Average LOS | |
| Intersection | 21.55 | C | | 49.3 | D | |

12
Based on the test results, the proposed system offers a reduction in cycle length ranging from 31.25 to 64.38% while the reduction in delay time ranges from 43.85 to 90.09%. Table 14 offers the summary for all ten scenarios.

| Test No. | Delay Proposed system | Percent reduction in delay time | Time cycle Proposed system | Percent reduction in cycle length % |
|----------|-----------------------|---------------------------------|----------------------------|-----------------------------------|
| 1        | 10.74                 | 76.5                            | 68.5                       | 57.19                             |
| 2        | 14.32                 | 71.59                           | 74                          | 53.75                             |
| 3        | 4.63                  | 90.09                           | 57                          | 64.38                             |
| 4        | 17.85                 | 64.16                           | 86                          | 46.25                             |
| 5        | 17.62                 | 66.18                           | 88                          | 45                                |
| 6        | 19.56                 | 55.44                           | 88                          | 45                                |
| 7        | 25.37                 | 50.45                           | 108                         | 32.5                              |
| 8        | 15.68                 | 63.95                           | 86.5                        | 45.94                             |
| 9        | 29.03                 | 43.85                           | 110                         | 31.25                             |
| 10       | 21.55                 | 56.29                           | 94                          | 41.25                             |

Table 14. Summary of the ten scenarios

![Chart showing the difference in delay between the real time of the proposed system and fixed time under SIDRA](image)

**Figure 7.** Difference in delay between the real time of the proposed system and fixed time under SIDRA.

Figure 8 shows the differences between the cycle length in the proposed system and the fixed time set by SIDRA software.
6. Conclusions
The level of service, delay time, and cycle length at signalised intersections are significantly improved by the use of real-time control systems to replace fixed time systems.

From the results in this work, the following points can be concluded:

1. Using an intelligent traffic control system allows the overall performance of an intersection to be improved.
2. The resulting reduction in delay time is about 40 to 90%.
3. Costs and air pollution are both reduced.
4. The reduction in cycle length is noticeably large, between 30 and 65%.
5. The results of applying the variance algorithm are more dynamic than those of applying the slab algorithm.
6. The slab algorithm offers less efficiency than the variance algorithm.

References
[1] K. Kelewatan and P. Berlampu 2017 Calibrating and Modelling of Statistical Delay for Signalized Intersections at Calibrating and Modelling of Statistical Delay for Signalized Intersections at Al-Nasiriyah City in Iraq J. Kejuruter 29 pp. 22–34, 2018.
[2] K. Vidhya and A. B. Banu, “Density Based Traffic Signal System 2014 Int. J. Innov. Res. Sci. Eng. Technol. 3 pp 2218–2223.
[3] P. D. D. Pukale, P. Chauhan, A. S. Satish, P. Nawal, and N. Kumari 2016 Density Based Traffic Control System Using Video Processing (Hardware and Software Implementation) Imp. J. Interdiscip. Res. 6 pp 293–297.
[4] U. Farooq, H. M. Atiq, M. U. Asad, A. Iqbal, and Z. Azmat 2010 Design and Development of an Image Processing Based Adaptive Traffic Control System with GSM Interface IEEE pp 166–171.
[5] P. Choudekar, S. Banerjee, and M.K.Muju 2011 Implementation of Image Processing in Real Time Traffic Light Control IEEE pp 94–9.
[6] D. Nicole, J. Nicola, and J. Guerra 2018 Smart Traffic Light Control System IEEE.
[7] L. Singh, S. Tripathi, and H. Arora 2009 Time Optimization for Traffic Signal Control Using Genetic Algorithm Int. J. Recent Trends Eng. 2 pp 4–6.
[8] A. Panda, A. Naik, and P. Patel 2015 Application of Image Processing In Road Traffic Control Int. J. Adv. Res. Trends Eng. Technol. 2 pp 42–45.
[9] P. Maheshwari, D. Suneja, P. Singh, and Y. Mutneja 2015 Smart Traffic Optimization Using Image Processing IEEE pp 1–4.
[10] P. Jadhav, P. Kelkar, K. Patil, and S. Thorat 2016 Smart Traffic Control System Using Image Processing Int. Res. J. Eng. Technol. pp 1207–1211.
[11] B. Eamthanakul, M. Ketcham, and N. Chumuang 2017 The Traffic Congestion Investigating System by Image Processing from CCTV Camera IEEE.
[12] M. A. Dye, A. W. Rufi, M. A. Dye, R. Mcdonald, and A. W. Rufi 2007 Network Fundamentals CCNA Exploration Companion Guide Cisco Press.
[13] W. Carlos, C. Bobda, and J. Hossain 2020 Internet of Things Internet of smart-cameras for traffic 283 lights optimization in smart cities Elsevier 11.
[14] C. Xie, J. Wang, Z. Zhang, Y. Zhou, L. Xie, and A. Yuille 2017 Adversarial Examples for Semantic Segmentation and Object Detection Proc. IEEE Int. Conf. Comput. Vis. pp 1378–1387.
[15] M. Baron 2013 Probability and Statics for Computer Scientists Second Edition CRC Press Taylor & Francis Group 84.
[16] D. K. R. Swarna, C. C. W. Jayasuriya, and M. T. R. Jayasinghe 2011 Computer Modelling of Suspension Bridges for Light Vehicular Traffic.
[17] E. S. Prassas and R. P. Roess 2020 Highway Capacity Manual: A Conceptual and Research History Volume 2 Signalized and Unsignalized, Springer Nature Switzerland AG 2.
[18] N. J. Garber and L. A. Hoel 2009 Traffic and Highway Fourth Edition. Library of Congress.
[19] https://www.sidrasolutions.com.