Journal of Applied Science, Engineering, Technology, and Education *ISSN: 2685-0591 (online)*
Vol. 8 No. 1 (2026)
https://doi.org/10.35877/454RI.asci4542

A Real-Time Intelligent Traffic Controller at Signalized Intersections in Samawah City

Noorance Al-Mukaram[a,*], Ameera M. Awad[a*], Salah Alheejawi[b], & Ameer W. Abdul Sattar[c]

[a] *Civil Engineering Department, College of Engineering, Al-Muthanna University, Samawah, Iraq*
[b] *ICT Department, Technical Institute of Samawah, Al-Furat Al-Awsat Technical University, Samawah, Iraq*
[c] *Electronics Engineering Department, College of Engineering, Al-Muthanna University, Samawah, Iraq*

Abstract

Based on computer vision, this study aims to develop algorithms of real-time traffic control systems for analyzing and tracking vehicles approaching a signalized junction. Video recording sequence was installed near an intersection to control traffic lights during traffic congestion situations in Samawah city/ Iraq. The project was designed using SIMULINK, which MathWorks created. Four algorithms were proposed to analyze the video signal inputs and estimate the number of vehicles detected. Gaussian mixture model and edge detection with frame differencing method were used to detect and track arrived vehicles. Optical flow-based approach was used to determine the number of stopping vehicles. Additionally, vehicle classification algorithm was used to detect types of vehicles. In Gaussian mixture model algorithm, implementing trained mask and geometric transform on each frame improved the perception of outputs, which is defined by counting 1100 vehicles on the approach. By using color detection, more control over traffic flow was obtained by prioritizing certain cars. The obtained results showed good representation of vehicle classification for the data detected in the developed system compared with the empirical data. The estimated errors were determined by achieving RMSPE < 15%, GHE < 5 and U_m < 1.

Keywords: Color detection; Gaussian mixture model; Image segmentation; Optical flow; Vehicle detection.

Received: 15 December 2025 Revised: 10 March 2026 Published: 30 April 2026

1. Introduction

Traffic congestion is one of the major concerns in modern cities such as Edmonton. According to studies conducted by the City of Edmonton in 2018, it is evident that congestion usually occurs at the major junctions in the morning and afternoon rush hours, particularly near junctions. The congestion causes an increase in road user delays on roads, and the number of traffic collisions reaches 57% (13,587) of all 24,003 at intersections (City of Edmonton, 2018). To reduce pedestrian-vehicle interaction on the road, it is essential to consider several factors when installing conventional traffic light systems, for instance, pedestrian density, number of vehicles and junction layouts. The traffic lights should be automatically controlled with fixed-time values. However, where no cars arrived at the green period on a particular approach, a queue of stopped vehicles exists on other methods because of the onset of red. In addition, the programmed traffic signal should be adapted to specific pedestrian group's needs, particularly children, older people, and disabled people.

Different traffic light systems are applied to control traffic movement among road users and reduce overall delays on road networks, particularly at junctions. Unlike fixed-time signals, there are green signal countdown displays and flashing green devices. Several researchers claimed that these systems increase drivers' hesitation in their stop/go decisions, particularly within the 3-second yellow period, which may lead later to rear collisions if drivers stop early (Factor et al., 2012; Köll et al., 2004; Lum and Halim, 2006; Shen and Wang, 2015). To improve the performance and capacity of junctions, inductive loop detectors are embedded in the pavement to count several vehicles passing on these detectors and then increase the green duration. This technique significantly reduces the number of conflicts by

* Corresponding author.
E-mail address: noorance@mu.edu.iq; ameer.wisam2015@mu.edu.iq

up to 26% and decreases overall delay by 13% (Dilek and Dener, 2023; Zemmouchi-Ghomari, 2025). However, the extension in the green phase may increase the number of red light violations because of the effect of the dilemma zone near signalized intersections (Al-Mukaram, 2018; Zemmouchi-Ghomari, 2025).

Since the aforementioned traffic light systems are very prone to traffic congestion and conflicts, there is a need to install an intelligent traffic light system for detecting vehicles through images rather than using electronic sensors embedded in the pavement. Santosh et al. (2024) proposed a similar methodology using a video processing system with radio frequency identification (RFID) technology with sensors placed under the pavement for detecting oncoming vehicles. The proposed model considers a more robust design by implementing multiple algorithms for different scenarios, including law enforcement.

However, it does not include image processing of pedestrian movement, and the maintenance cost of this technology is considerably high. Additionally, another simulation study from Spain was presented for tracking pedestrian and vehicle movements at signalized junctions using image analysis and processing to make real-time decisions (Serrano et al., 2005). The real-time limitations in the proposed algorithms concern enforcement rules and color classification features for providing a safe pass of emergency vehicles through intersections. Moreover, image matching and filter techniques were used in the Lebanese study to calculate the percentage of traffic congestion based on the differences shown in the records. The accuracy of the established model was 90%, as mentioned by Anitha et al. (2023). Other studies developed models for controlling traffic lights using artificial intelligence technology with techniques such as photoelectric sensors, Fuzzy logic controllers, Nios II and digital signal processors (Mu et al., 2010; Pandey et al., 2017; Salama et al., 2010; Zhao et al., 2009).

While conventional systems like inductive loop detectors and RFID-based sensors offer partial solutions, they often suffer from high installation costs, limited adaptability, and maintenance challenges. These limitations underscore the need for a flexible, vision-based traffic control system capable of real-time responsiveness and low-cost deployment.

The scope of this study is to develop an intelligent traffic control system based on measuring traffic density using real-time video processing. SIMULINK is an efficient tool for modelling and simulating such systems. The next section will discuss the developed algorithms.

2. Methodology

2.1. Intersection Layout and Cameras Locations

This project considers a cross-signalized intersection of one-way approaches in Samawah city that located in the south of Iraq. It was suggested that cameras be set alongside traffic light signals to provide a clear, unobstructed view of oncoming vehicles. A possible orientation of coverage zones is shown in Figure 1. Both cameras were connected to a central processing unit where data was analyzed, and appropriate decisions were made. To operate, the system needs access to junction traffic light information and permission to change their phases based on real traffic conditions.

Fig. 1. Cameras coverage zones at one-way signalized intersection

2.2. Geometric Transformation

The geometric transformation is a composition of scalar, translation, shear, and rotation transforms are used to eliminate the geometric distortion (Jain, 2001). In the current work, the geometric distortion is defined by the distance between the camera and the end of the road. The geometric distortion can be eliminated by finding a proper Affine matrix "A" and multiplying it with the old coordinates of each frame in the video.

$$A = S_{scalar} \times T_{translation} \times SH_{shear} \times R_{rotation} \tag{1}$$

$$U = A \times F \tag{2}$$

where U is the new coordinate of the frame. To obtain the best mapping points, it can estimate E as follows:

$$E = \frac{H}{\cos\theta} \tag{3}$$

where H is the height of the camera and Θ is the angle as shown in Figure 2.

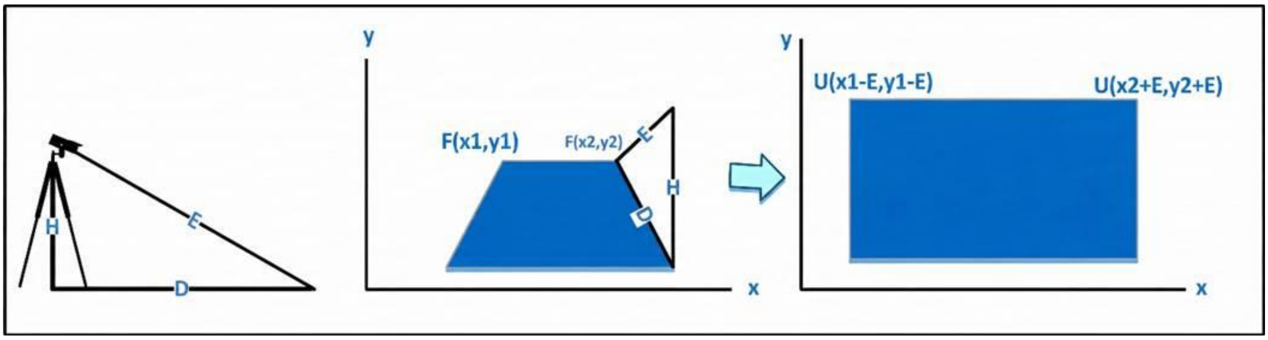

Fig. 2. The geometric mapping points

Using fitgeotrans, a MATLAB Function and providing the fixed points as well as moving points, the Affine matrix "A" can be presented as Table1 .

Table 1. Computed Affine Matrix "A"

Row	Column 1	Column 2	Column 3
1	4.36165955345044	$1.56680846969637\times10^{-15}$	$3.54802642594286\times10^{-18}$
2	2.12864578716075	4.36866301085346	0.00700345740302175
3	-1021.74997783716	-3.36165955345086	1

By multiplying the Affine matrix with each frame, the small regions in the end of the road are going to be enlarged as shown in Figure 3.

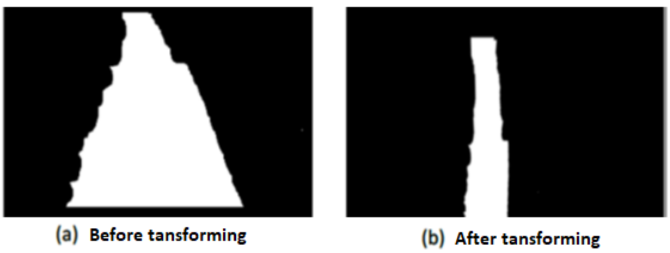

Fig. 3. Captured frame before and after geometric transformation

2.3. Computer Vision System Software

In SIMULINK, systems are built using blocks. Since SIMULINK is integrated with MATLAB, data can be easily transferred between the programs. SIMULINK has a built-in Computer Vision System Toolbox that supports stream processing architecture through blocks. It allows key stream/image processing techniques crucial for overcoming noise and differentiating an object from its background. This project implements four major algorithms to minimize traffic congestion. Since there are many possible scenarios in a stream of moving vehicles, two algorithms can be applied. Gaussian mixture models (GMMs) and edge detection algorithms were used to track and count vehicles driving on the road during the green phase. In the case of the red light, an optical flow-based algorithm was used to determine the number of cars that stopped in the queue. The vehicle classification algorithm was run in parallel with the aforementioned algorithms, which use color information to detect and track a vehicle of a specific color. This is an additional feature of the real-time traffic control system. While vehicle classification is not necessarily required to identify traffic flow, it can be implemented for many practical applications. For example, it could detect the presence of fire trucks and alter traffic flow to accommodate them. This property may prove vital in preventing collisions between firetrucks and civilian vehicles, which are a significant cause of death to firefighters as well as civilians (Donoughe et al., 2012). An overview of the entire system is shown in Figure 4.

As shown in Figure 4, the system requires feedback from the traffic lights to determine which algorithms should run. Using a switch, the system implements either the GMMs, edge detection algorithm or optical flow algorithm. The vehicle classification algorithm runs regardless of the state of the traffic lights. The Camera 2 system is identical to the Camera 1 system. This is useful because it allows for a modular design rather than mainly designed systems for various locations.

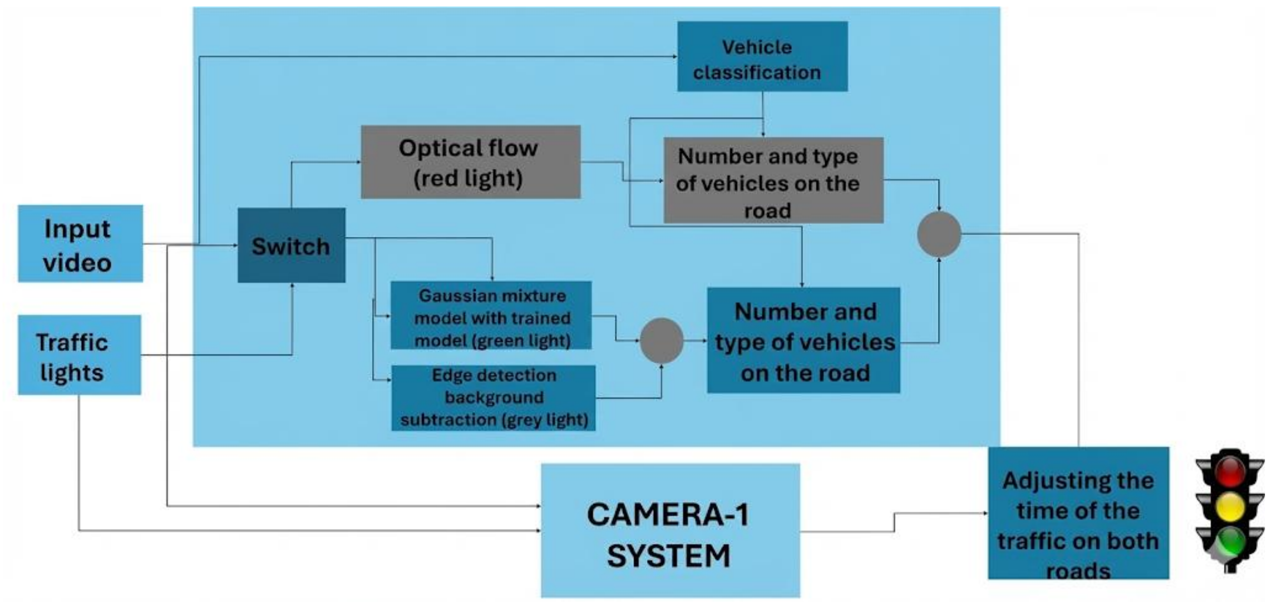

Fig. 4. The whole system diagram. Camera 2 system is the same as Camera 1 system. When one camera runs the Optical Flow algorithm, the other camera runs the GMM and Edge Detection

3. Description of the Developed Algorithms

3.1. Tracking and Counting Vehicles during GREEN Traffic Light

In this subsystem, an edge detection method with background subtraction was designed, modelled, and tested initially, followed by modified GMMs with a trained mask. For tracking a vehicle when the green light is on, it is necessary to extract and distinguish its movement based on static background, which is known as segmentation (Hadi et al., 2014). Background subtraction with frame differencing method was used. Input video was converted first to the intensity or grayscale format, and an automatic thresholding technique was applied to convert it into binary format. Noise, changes in weather conditions and camera movements may influence the precision of image capturing. Therefore, the median filter was used to remove these disturbances. Next, the current frame is subtracted from the previous frame to get the intensities for the pixel locations which have changed in the two frames. If the difference in pixel values for a given pixel is smaller than a threshold, it is then considered as 'background,' otherwise, as 'foreground' or moving vehicle pixels.

After extracting objects of interest from the static background, their edges were detected with an edge detection algorithm (Singh and Singh, 2015). As the edges were identified by edge detection, a few morphological operations were needed to close the boundaries by applying a morphological closing operation (Gonzalez and Woods, 2008). Then, the edges were obtained and moving objects were detected. Next, blob analysis was employed to count the number of vehicles found. The coordinates of the objects were identified by edges that distinguished them, and bounding boxes (i.e. blobs) were drawn around the vehicles. The number of boxes gave the number of vehicles, and the result was displayed. Figure 5 illustrates the system that consists of segmentation, feature extraction, and vehicle counting.

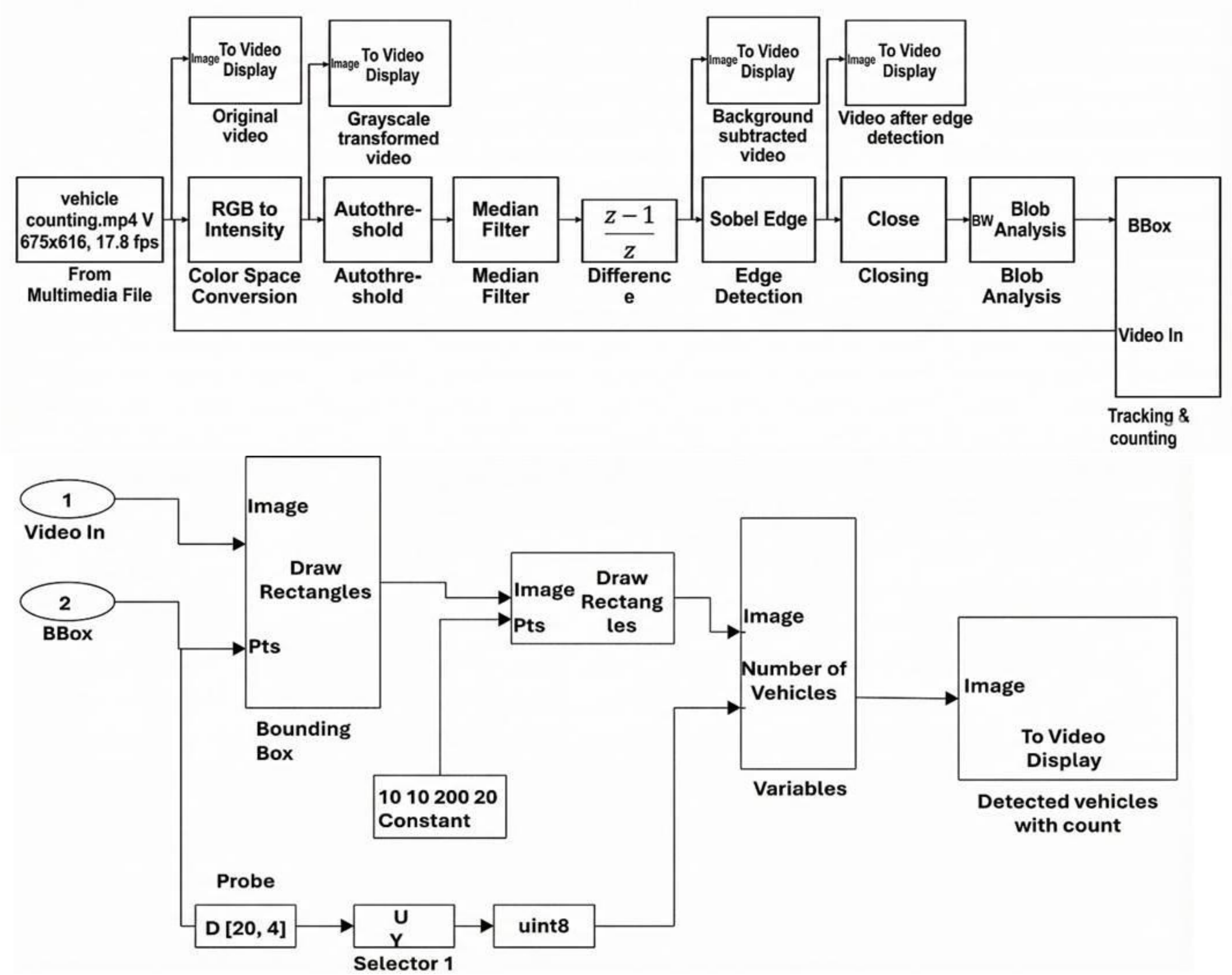

Fig. 5. SIMULINK model for edge detection

In Figure 6, Gaussian mixture models (GMMs) were used to detect the foreground of the moving vehicles in white regions. However, due to winds, the GMMs are sensitive to any fluctuation or slow camera movement. Therefore, generating a trained mask will be more critical to avoid false detection on the roadside. The mask depends on interesting movements, such as vehicles, which have fast movements compared to the slow movements of the camera due to the wind. In this case, the rapid movements will wipe out big regions of the road, while the slow movements will wipe out small roadside areas by adding consecutive frames. A proposed resultant mask was formed by adding and normalizing consecutive frames during a specific time. Then, a proposed morphological method was applied to classify the resultant mask into fast and slow-movement regions by defining each pixel's search range. The final consequent mask and foreground frames were stored in a memory block and combined by the AND logical operation to avoid unwanted regions besides the road. The output of the AND block is passed through a geometric transform. The counter will control the MATLAB block function to indicate the suitable time to generate the trained mask and control the foreground detection using GMM output.

Then, geometric transformation was applied on each frame to maximize the small sizes of moving vehicles at the end of the road. The geometric transformation is performed by the 'warp block'. Finally, a blob analysis block was used to count the white regions on each frame. The warp block has been provided with a constant block (3×3) containing Affine "A", the transform matrix described previously in section 2.2.

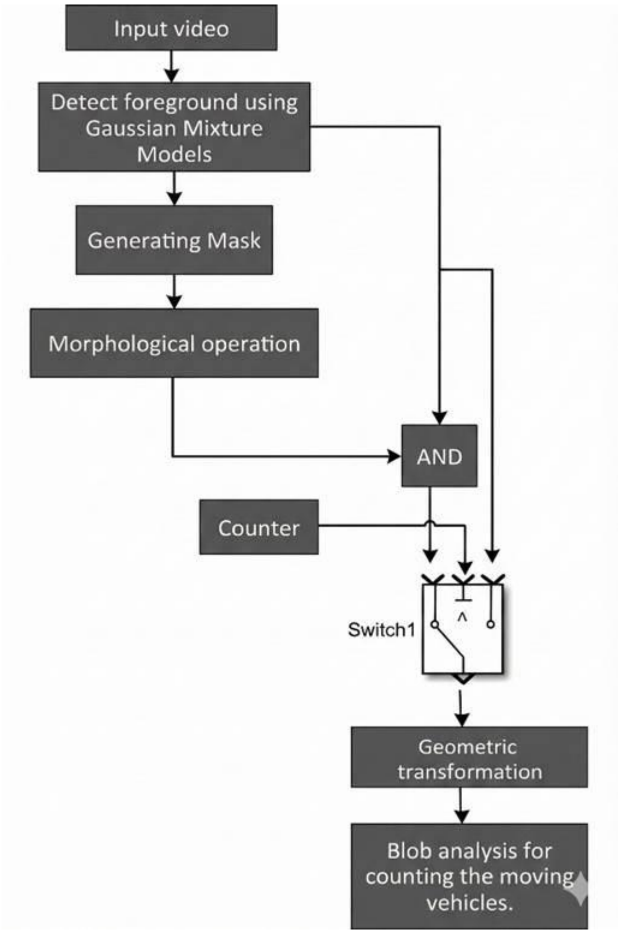

Fig. 6. The block diagram of the GMM with trained mask

The moving vehicles will form white regions, and this subsystem will be used to count the number of moving cars on the road during the green light of the traffic light. The output of this subsystem will not take the final trained mask into account until the counter reaches a specific frame sequence because this counter controls the two switches in the SIMULINK diagram to reduce complexity.

In summary, the improved GMMs introduce good recognition between a background, which is presented by the road and the foreground, which is given by moving vehicles, after applying three Gaussian distributions, which are the first for the road and the others for cars and the darkest components, such as shadows (Pham et al., 2010; Stauffer and Grimson, 1999). The output of this subsystem will be represented by several vehicles on the road, and the new SIMULINK design was tested using a rectangular block on a white region and overlapped with the input video, as shown in Figure 7.

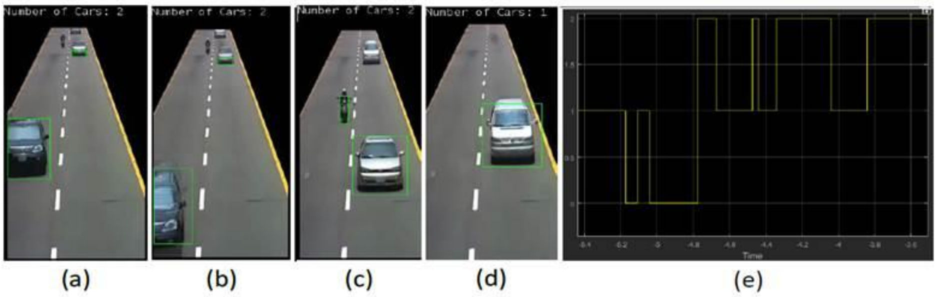

Fig. 7. (a-d) Different frames sequences after detecting the number of vehicles in the road, (e) Number of vehicles over time.

3.2. Tracking and Counting Vehicles during RED traffic Light

Figure 8 illustrates the optical flow-based sub-system isolated from the more extensive system. The input video was immediately converted to an intensity (i.e. grayscale) format because it needs to be prepared for the bottom-hat transformation SIMULINK provides. After the transformation, the output was sent to the optical flow block to find velocity vectors. The numerical values of the velocity vectors were not used so much as the graphical representations of the velocity vectors. This can be done in the block immediately after the optical flow block. The block-labelled optical flow lines generate velocity vectors drawn onto the original image using the draw lines block. Because the system requires these graphical representations of the vectors to be visible, a gain 80 was applied to the original vectors. This does not affect the system's accuracy because the motion was detected relative to other objects in the image sequence. By using the gain to all images, relative motion remains unchanged. While the output looks nothing like vehicles, the blob analysis block can correctly count the large white spaces in the image, which is a close approximation of the number of cars in the image.

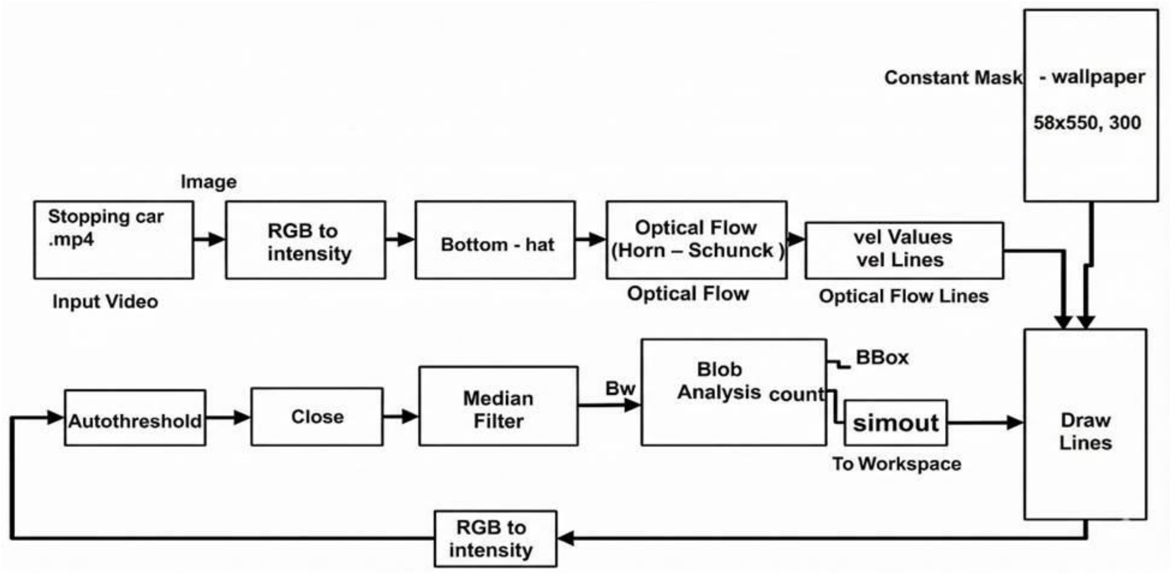

Fig. 8. SIMULINK model of optical flow

However, before being sent to the blob analysis block, a median filter is applied to the image, which removes the small isolated white pixels. By setting this value to 250 pixels, it can be assumed that any blobs detected will not be from noise but from a moving vehicle. The count signal generated by the blob analysis block is sent to the MATLAB workspace, where additional work must be done to produce the desired result. Since this algorithm aims to obtain a number that represents the number of vehicles waiting in the queue during the red-light period, the count signal needs to be averaged to produce a single number. Before averaging the signal, however, all the zero elements of the count array were taken out. This is because they did not represent actual information but a by-product of the optical flow algorithm. Because it relies on frame information, the optical flow generates vector lines in a series of impulses, as shown in Figure 9. Therefore, the zeros need to be removed from the array, and using the mean function from MATLAB gives an average of the number of blocks detected, which can be approximated to be used as the number of vehicles in the image sequence.

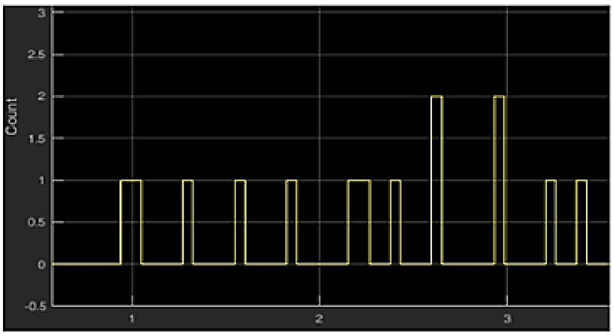

Fig. 9. Time plot of blob count

3.3. Color Classification

There are different methods for detecting a certain type of vehicle in computer vision techniques. For simulation purposes, equations are used to classify vehicles based on shape and area (Al-Mukaram, 2018; Premaratne et al., 2023). In this work, a color-based approach was implemented. The main idea is to make a new development in traffic control by giving priority to specific vehicles such as an ambulance, firetruck, bus, etc.

A proposed technique with MATLAB functions was implemented in SIMULINK that checks every pixel looking for the specified color. Due to lighting conditions, various shades of color are considered, which makes the system more robust against lighting conditions. The function's input was a frame, and the output was a modified frame, with the specified color filled in and the other pixels blacked out. Since the input video has 30 fps, the system will process one frame every 33.3 ms. The function will operate on every image sequence frame, effectively tracking the specified color.

There was some noise in the input images resulting from wind or rain, and the algorithm was performed on this noisy signal. The system performs well even with noise. The color classification algorithm can be given in Figure 10. As shown, only a specific color of the vehicle in the function image will be detected (for example green frame for cars and blue frame for heavy trucks), and any other vehicles will turn black and neglected.

4. Overall System Testing Model

Up to this point, discussion has been limited to the design and results of individual subsystems of the entire project. However, for a complete picture to be presented, the design of the overall system must be considered. Figure 11 shows a block diagram describing the whole system.

Several terms need to be defined to understand the figure. A Primary Road (PR) is a road on which more vehicles are detected, such as R1 and R2, which are the number of cars on Road 1 and 2, respectively. The data can be obtained from both cameras. The subsequent operations were performed on PR, and its dual was performed on the other road. A PR was determined at the beginning of every cycle of the algorithm. A time interval T_{int} is defined as the length of time a traffic light will be in a particular state (i.e. RED or GREEN).

$$T_{int} = \begin{cases} T_{max} & T_{min} + C(R_1 - R_2) \geq T_{int} \\ T_{min} + (R_1 - R_2) & T_{min} \leq T_{int} \leq T_{max} \end{cases} \tag{4}$$

Where, T_{max} and T_{min} are defined as a state's maximum and minimum duration in one iteration of the algorithm, respectively. C is a constant that can be added for specific vehicle types, R_1 and R_2 are the number of cars on Road 1 and 2, respectively. These values can be set using statistical data for particular roads. Also, a constant can be included if specific vehicles are detected. For example, buses make sense to allow for a longer duration for traffic to pass.

A counter calculates the number of times that one road has been chosen as the PR. Once a road has been selected as the PR twice, the system will automatically switch the PR traffic light to red and allow traffic from other roads to pass. This ensures that the maximum waiting time for a vehicle would be $2T_{int}$. The counter is reset if the PR was switched or the PR light defaulted to red due to the same road being selected as the PR for more than two consecutive iterations. A wait block is added to allow traffic to flow for the calculated time.

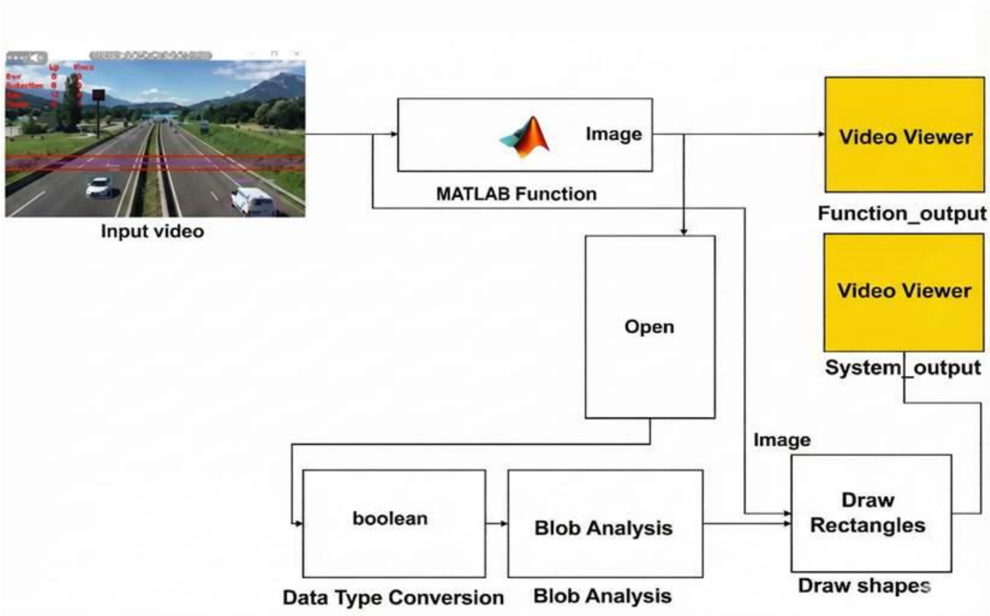

Fig. 10. SIMULINK blocks for color classification.

While the current vehicle classification relies on color detection, it is sensitive to lighting variations and shadows. To improve robustness, future work will incorporate shape descriptors and convolutional neural networks (CNNs) for multi-feature classification. This will enable detection under diverse environmental conditions and improve accuracy for emergency vehicle prioritization.

The system was tested on a standard workstation (Intel Core i7, 16 GB RAM) using MATLAB/SIMULINK. In terms of performance, the system processes video at 30 frames per second, with each frame analyzed in approximately 28 to 35 milliseconds depending on the algorithm in use. The edge detection combined with frame differencing exhibits low computational complexity, making it suitable for real-time operation with minimal delay. The Gaussian Mixture Model (GMM) with a trained mask introduces moderate complexity due to mask generation and geometric transformation, averaging around 35 milliseconds per frame. The optical flow algorithm incurs a higher computational load, requiring approximately 40 to 45 milliseconds per frame due to the generation of velocity vector fields. The color classification module is lightweight, processing frames in about 25 milliseconds, though it remains sensitive to lighting and shadow variations. Overall, the system satisfies real-time constraints for single-intersection deployment. However, for broader scalability across multiple intersections, GPU acceleration or parallel processing is recommended.

5. Data Representation and Discussion

Traffic composition of 1100 vehicles were collected from Al-Nadai signalized intersection in Samawah city during a period from 11:30 AM to 12:30 PM on Monday 5th October 2025 in good weather conditions (sunny) using video recording technique. The junction is operated by fixed-time traffic signal-controlled system as shown in Figure 1. Details of intersection geometry and traffic characteristics can be found in Table 2. Following previous studies (Al-Mukaram, 2018; Alterawi, 2014; Nassrullah, 2016), traffic observation was classified into two main categories based on vehicle length. Passenger cars (PCs) include all vehicles that are no more than 6 meters in length and have a seating capacity of 10 passengers or less.

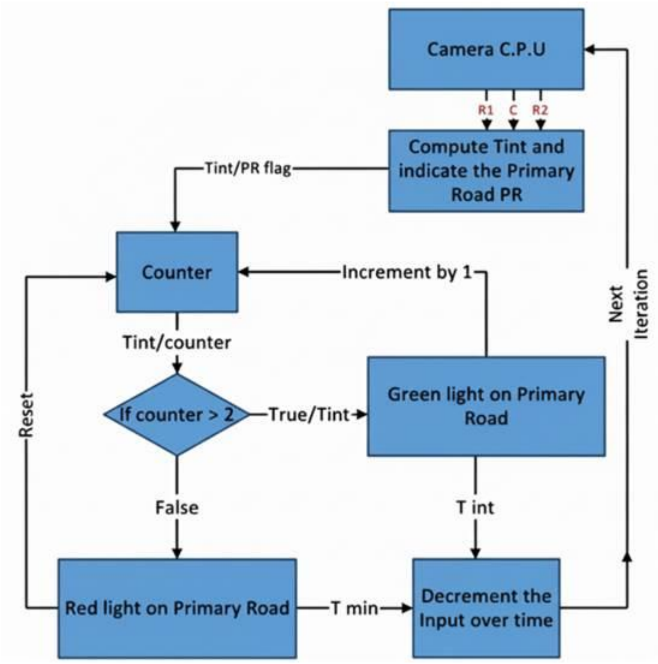

Fig. 11. Overall system testing model.

However, vehicles of more than 6 m length are mainly called heavy goods vehicles (HGVs) involving single and double-deck buses in addition to large vans and trucks that are used for commercial purposes of weight over 3.5 tons. Motorcycles and bicycles were neglected because of rare observation. Then, the video records were examined in the developed system for detecting objects (vehicles), extracting and counting as presented in Table 3.

Table 2. Characteristics of Al-Nadai signalized intersection.

Details	West bound	South bound
Traffic volume, veh/hr	462	638
PCs, %	69	78
HGVs, %	31	22
Green, sec	20	33
Yellow, sec	2	2
Red, sec	98	85
No. of Lanes	2	2

Source by authors.

Table 3. Total observed and detected vehicles.

Type of Data	Number	Mean Length, m	Standard Deviation, m
Obs. PCs	816	4.28	0.78
Det. PCs	788	4.31	0.80
Obs. HGVs	284	7.98	2.01
Det. HGVs	312	8.43	2.14

Det.= Detected data
Obs.= Observed data

Next, the differences between the observed and detected data were tested by several statistical measures such as root mean square percentage error (RMSPE), Theil's mean difference coefficient (U_m) and Geoffrey E. Havers (GEH) as shown in Table 4. The findings showed good representation of the detected objects in the developed system to the observed vehicles.

Table 4. Total observed and detected vehicles.

Statistical test	Vehicle Classification		*Satisfactory Value*
	PCs	HGVs	
RMSPE%	3.95	4.37	< 15%
GEH	0.05	0.16	≤ 5
U_m	0.12	0.28	< 1

6. System Limitations

a) Camera Positioning: The accuracy of vehicle detection is sensitive to camera height and angle. Improper placement may lead to occlusion or distorted perspective, affecting blob analysis and optical flow results.
b) Network Latency: Real-time responsiveness depends on low-latency data transmission between cameras and the central processing unit. Delays may hinder timely traffic light adjustments.
c) Occlusion: Vehicles may be partially or fully blocked by other vehicles or environmental objects (e.g., poles, trees), reducing detection accuracy. This is especially problematic during peak traffic hours.
d) Environmental Noise: Weather conditions such as rain, dust, or glare can introduce noise into video feeds, impacting segmentation and classification.

7. Conclusions

The project's scope is limited to analyzing real-time traffic videos and designing algorithms to detect and track vehicles intelligently. The developed algorithms include GMMs with a trained mask, edge detection, optical flow and vehicle classification. Edge detection could detect and track moving vehicles with reasonable accuracy. In case of false detections, GMMs are used to correct errors. However, this increases the complexity of the design. In GMMs, vehicle detection was improved significantly by using three Gaussian distributions, a trained mask, and geometric transforms on each frame. Optical flow is an effective method to detect and count vehicles in a queue. Color classification effectively tracked a blue car throughout the frame despite the noise in the video. Overall, the proposed system seems promising in its ability to deliver an intelligent real-time traffic controller. It can monitor and control the traffic at one-way intersections by image processing based on the number of vehicles arriving stop-line. The advantage is that the newly developed algorithms can be incorporated into an existing traffic control system without investing in sophisticated hardware, like additional sensors, making it highly cost-effective. As a result, the traffic data from one intersection collected at any point can be transmitted wirelessly to the succeeding intersections to enhance effective coordination of real-time traffic control. The accuracy of this work can be developed so that the system operates even in low visibility conditions such as fog or darkness. Since this capability has not yet been experimentally validated, it is a subject for future investigation.

Declaration: The authors declare no conflict of interest. Also, the manuscript was done depending on the personal efforts.

Funding Statement: The authors confirm that no funds were received from any organization.

References

Al-Mukaram, N., 2018. Modelling drivers' behaviour within the dilemma zone at traffic signal junctions. University of Salford, UK.

Alterawi, M., 2014. Development of a micro-simulation model to evaluate shuttle-lane roadwork operations. University of Salford, UK.

Anitha, K., Manjula, H.M., Leelavathi, H.P., Swetha, P., 2023. Automated Traffic Management Using Image Processing, in: 2023 International Conference on New Frontiers in Communication, Automation, Management and Security (ICCAMS). Presented at the 2023 International Conference on New Frontiers in Communication, Automation, Management and Security (ICCAMS), IEEE, Bangalore, India, pp. 1–4. https://doi.org/10.1109/ICCAMS60113.2023.10525834

City of Edmonton, 2018. Motor Vehicle Collisions.

Dilek, E., Dener, M., 2023. Computer Vision Applications in Intelligent Transportation Systems: A Survey. Sensors 23, 2938. https://doi.org/10.3390/s23062938

Donoughe, K., Whitestone, J., Gabler, H.C., 2012. Analysis of firetruck crashes and associated firefighter injuries in the United States. Annals of Advances in Automotive Medicine/ Annual Scientific Conference, PMID: 23169118; PMCID: PMC350342 56, 69–76.

Factor, R., Prashker, J.N., Mahalel, D., 2012. The flashing green light paradox. Transportation Research Part F: Traffic Psychology and Behaviour 15, 279–288. https://doi.org/10.1016/j.trf.2012.01.003

Gonzalez, R.C., Woods, R.E., 2008. Digital Image Processing. Pearson Prentice Hall.

Hadi, R., Sulong, G., George, L., 2014. An Innovative Vehicle Detection Approach Based on Background Subtraction and Morphological Binary operations Methods. Life Science Journal 1111, 230–238.

Jain, A.K.., 2001. Fundamentals of Digital Image Processing. Prentice-Hall, India.

Köll, H., Bader, M., Axhausen, K.W., 2004. Driver behaviour during flashing green before amber: a comparative study. Accident Analysis & Prevention 36, 273–280. https://doi.org/10.1016/S0001-4575(03)00005-8

Lum, K.M., Halim, H., 2006. A before-and-after study on green signal countdown device installation. Transportation Research Part F: Traffic Psychology and Behaviour 9, 29–41. https://doi.org/10.1016/j.trf.2005.08.007

Mu, H., JianNing Yu, LinZhong Liu, 2010. Traffic signals control of urban traffic intersections group based on fuzzy control, in: 2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery. Presented at the 2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), IEEE, Yantai, China, pp. 763–767. https://doi.org/10.1109/FSKD.2010.5569355

Nassrullah, Z., 2016. Development of a micro-simulation model for motorway roadworks with the use of narrow lanes and lane closure schemes. University of Salford, UK.

Pandey, S., Mathur, P., Patil, T., 2017. Real time traffic signal control using fuzzy logic controller: Review, in: 2017 1st International Conference on Intelligent Systems and Information Management (ICISIM). Presented at the 2017 1st International Conference on Intelligent Systems and Information Management (ICISIM), IEEE, Aurangabad, pp. 307–313. https://doi.org/10.1109/ICISIM.2017.8122190

Pham, V., Vo, P., Hung, V.T., Bac, L.H., 2010. GPU Implementation of Extended Gaussian Mixture Model for Background Subtraction, in: 2010 IEEE RIVF International Conference on Computing & Communication Technologies, Research, Innovation, and Vision for the Future (RIVF). Presented at the Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), IEEE, Hanoi, Vietnam, pp. 1–4. https://doi.org/10.1109/RIVF.2010.5634007

Premaratne, P., Jawad Kadhim, I., Blacklidge, R., Lee, M., 2023. Comprehensive review on vehicle Detection, classification and counting on highways. Neurocomputing 556, 126627. https://doi.org/10.1016/j.neucom.2023.126627

Salama, A.S., Saleh, B.K., Eassa, M.M., 2010. Intelligent cross road traffic management system (ICRTMS), in: 2010 2nd International Conference on Computer Technology and Development. Presented at the 2010 2nd International Conference on Computer Technology and Development (ICCTD), IEEE, Cairo, Egypt, pp. 27–31. https://doi.org/10.1109/ICCTD.2010.5646059

Santosh, K., Makkar, A., Conway, M., Singh, A.K., Vacavant, A., Abou El Kalam, A., Bouguelia, M.-R., Hegadi, R. (Eds.), 2024. Recent Trends in Image Processing and Pattern Recognition: 6th International Conference, RTIP2R 2023, Derby, UK, December 7–8, 2023, Revised Selected Papers, Part I, Communications in Computer and Information Science. Springer Nature Switzerland, Cham. https://doi.org/10.1007/978-3-031-53082-1

Serrano, Á., Conde, C., Rodríguez-Aragón, L.J., Montes, R., Cabello, E., 2005. Computer Vision Application: Real Time Smart Traffic Light, in: Moreno Díaz, R., Pichler, F., Quesada Arencibia, A. (Eds.), Computer Aided Systems Theory – EUROCAST 2005, Lecture Notes in Computer Science. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 525–530. https://doi.org/10.1007/11556985_68

Shen, J., Wang, W., 2015. Effects of flashing green on driver's stop/go decision at signalized intersection. J. Cent. South Univ. 22, 771–778. https://doi.org/10.1007/s11771-015-2581-z

Singh, S., Singh, R., 2015. Comparison of various edge detection techniques. Presented at the 2nd International Conference on Computing for Sustainable Global Development (INDIACom), IEEE.

Stauffer, C., Grimson, W.E.L., 1999. Adaptive background mixture models for real-time tracking, in: Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149). Presented at the Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE Comput. Soc, Fort Collins, CO, USA, pp. 246–252. https://doi.org/10.1109/CVPR.1999.784637

Zemmouchi-Ghomari, L., 2025. Artificial intelligence in intelligent transportation systems. Journal of Intelligent Manufacturing and Special Equipment 6, 26–42. https://doi.org/10.1108/JIMSE-11-2024-0035

Zhao, H., Zheng, X., Liu, W., 2009. Intelligent Traffic Control System Based on DSP and Nios II. Presented at the 2009 International Asia Conference on Informatics in Control, Automation and Robotics (CAR), IEEE, Bangkok, Thailand, pp. 90–94. https://doi.org/10.1109/CAR.2009.59