Density Based Dynamic Traffic Scheduling System

Krishna
Department Of Computer Science and Engineering
BMS College Of Engineering, Bangalore, India
krishnaravishankar555@gmail.com

Madhav Poddar
Department of Computer Science and Engineering
BMS College Of Engineering, Bangalore, India
madhav.poddar19@gmail.com

Giridhar M K
Department of Computer Science and Engineering
BMS College Of Engineering, Bangalore, India
giridharraju95@gmail.com

Amit Suresh Prabhu
Department of Computer Science and Engineering
BMS College Of Engineering, Bangalore, India
aspamit96@gmail.com

Umadevi V
Department of Computer Science and Engineering
BMS College Of Engineering, Bangalore, India
umadevi.cse@bmsce.ac.in

***Abstract*— It has been man's effort to minimise human intervention and automate procedures. This has been applied in controlling traffic too. At times we would've noticed sparsely populated lanes and highly populated lanes having the same amount of signal duration for movement of vehicles. We cannot have a one-size that fits all policy. The simple rule of thumb to overcome this problem was to accord a higher priority and more time to lanes having higher vehicle density. Keeping this in mind we have designed a traffic scheduling algorithm which allocates signal durations based on the traffic density of individual lanes. We determine the vehicular density up till a threshold distance in every lane and then accord priorities and proportionate signal durations. Our algorithm accounts for the real-time change in vehicular density. This is done in a round-robin fashion to prevent starvation problems in the algorithm**

Keywords-vehicular density, scheduling algorithm, round-robin, starvation problem, signal duration.

I. INTRODUCTION

Traffic management is one of the important areas in which automation can be introduced. Countless number of man hours can be saved and it also makes the procedure really efficient. Designing a system to that effect has been our goal. Firstly, the system should be able to identify vehicle and collect traffic data. Secondly, we need to account for real time changes in traffic. Thirdly, the scheduling algorithm has to be implemented. Identification of vehicles is done by using high resolution cameras placed such that the top view of the lanes are received in the camera feed. The scheduling algorithm has been designed to take into account the changing traffic density at real time. It is done by updating the individual lane vehicular counts constantly. This is implemented by spawning parallel system threads equal to the number of traffic lanes along with the scheduling thread. The allotment of green light times to individual lanes is done in a cyclic fashion to prevent any starvation.

The scheduling algorithm also takes into account the the noise generated while identifying traffic vehicles. We use the concept of disconnected components to identify false positives and improve the accuracy of the system. This system has proved to be a feasible, low-cost and profitable.

II. RELATED WORK

Karlsruhe [1] proposed a model-based tracking approach with a vehicle model for a intraframe matching strategy. The potential moving vehicles are identified in this approach by identifying the moving features in the image through the image segmentation component. Region-based tracking [2] is a technique that identifies the connected regions in the image and associates it with each vehicle and tracks them. The strategy uses a background subtraction technique. However it fails in congested traffic conditions where there are vehicles in the foreground image that partially occlude each other and thus the occluding vehicles are detected as a single large blob which results in reduced accuracy. Using a Kalman filter based approach helps the background estimate evolve through various changes in weather, time, lighting during the day. Active-Contour based tracking tracks active contour models. This technique has a relatively low computational complexity. Feature-based tracking identifies sub-features in the image which are distinguishable and important [3]. Even when there is partial occlusion some of these sub-features chosen are still distinguishable and hence helps to identify vehicles more accurately even during partial occlusion of vehicles. [4] and [5] are techniques that used a Haar-like feature detector for detection of vehicles.Other approaches involve use of a fuzzy logic system started by Zadeh [6]. Other approaches involve the use of Artificial neural networks where decisions are made based upon the previous knowledge. Finally there are also hybrid systems as developed by Patel [7] for urban traffic control applications.

III. PROPOSED SYSTEM

In the following section we explain in detail the technique used for simulating traffic light control by identifying the count of vehicles across each lane and carrying out a suitable scheduling algorithm. This scheduling algorithm has been implemented as a successor to vehicle counting and detection [8]. A brief description of the components of the system is given below.

A. *Configuration*

The configuration module allows the user to load a traffic surveillance video to the system. The initial frame which should have empty lane, can be set by selecting the frame number. The user can select a point on the frame displayed and hence construct a reference line by specifying the desired length in pixels. Two reference lines can be set by specifying the desired distance between them in pixels. The same procedure has to be applied to all the four lanes. This initialize the four threads, each assigned with a lane, with appropriate configurations. The surveillance videos are used here instead of live streaming, to stimulate traffic scenario, in order to show the behaviour of scheduling algorithm.

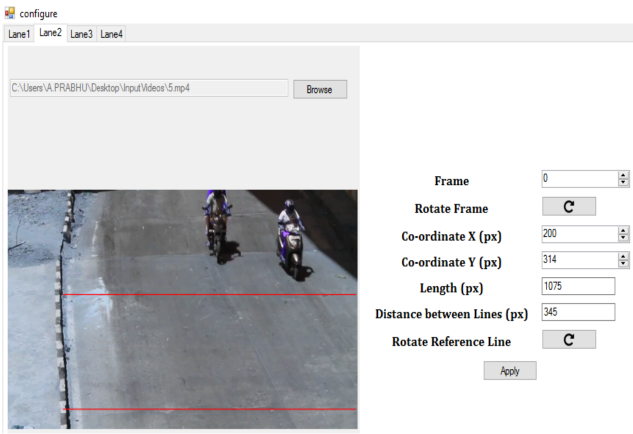

Figure-1 Configuration module (UI)

B. *Vehicle Count for Each Lane*

The proposed model assigns each lane to an independent thread. The thread is responsible for the task of determining the count of vehicles dynamically by taking into account the change in RGB value of the pixels along the single reference line used for counting vehicles. Also to ensure good accuracy for occluding vehicles a count of disconnected components based on assumed threshold values is applied. Here a disconnected component is said to be a set of continuous pixels which have changed from initial values.

Due to the presence of noise and distortion, the number of vehicles currently on the reference line is not equal to the number of disconnected components. So some thresholds are used to overcome this issue.

Four different thresholds have been used in the model. Firstly, RGB threshold(20), this is the sensitivity of the algorithm to the changing RGB values of pixels. Higher the threshold, vehicles having nearest colour as lane will be ignored. Lower the threshold, the system becomes too sensitive and considers minute changes as valid. Secondly, noise threshold(8), a very small group of pixels which form a disconnected component, caused due to shadow or dust on the reference line. Higher the threshold, valid changes get ignored and Lower the threshold, invalid changes get considered. Thirdly, gap threshold(20), this ignores the gaps formed between disconnected components. It is observed that sometimes, a single vehicle contains multiple disconnected components with minor gaps in between. If such gaps are fixed, it results in a single disconnected component. Higher the threshold, occluding vehicles are considered as one and Lower the threshold, single vehicle is considered as occluding vehicles. Fourthly, valid vehicle threshold(25), this is about considerable amount of pixels to be changed to consider it as a vehicle. Higher the threshold, it ignores some vehicles. Lower the threshold, invalid vehicles are considered to be valid. So choosing the best thresholds, gives better accuracy. Applying grid search on a part of the video, helps to determine the best thresholds.

The process of determining vehicle count starts from the frame successor to initial frame. The RGB values of the pixels forming the reference line in the current frame are compared with corresponding initial values. A binary array of length equals to length of reference line is formed whose entry corresponds to 1 if there is a

change greater than RGB threshold and 0 if not. Noise

and gaps are removed by using suitable thresholds. A set of continuous ones form a disconnected component. The number of disconnected components in the current frame, when deducted from that in the previous frame, gives the count of vehicles which exited the reference line in the current frame. This is added to the total vehicle count, which keeps track of total number of vehicles which crossed the reference line so far.

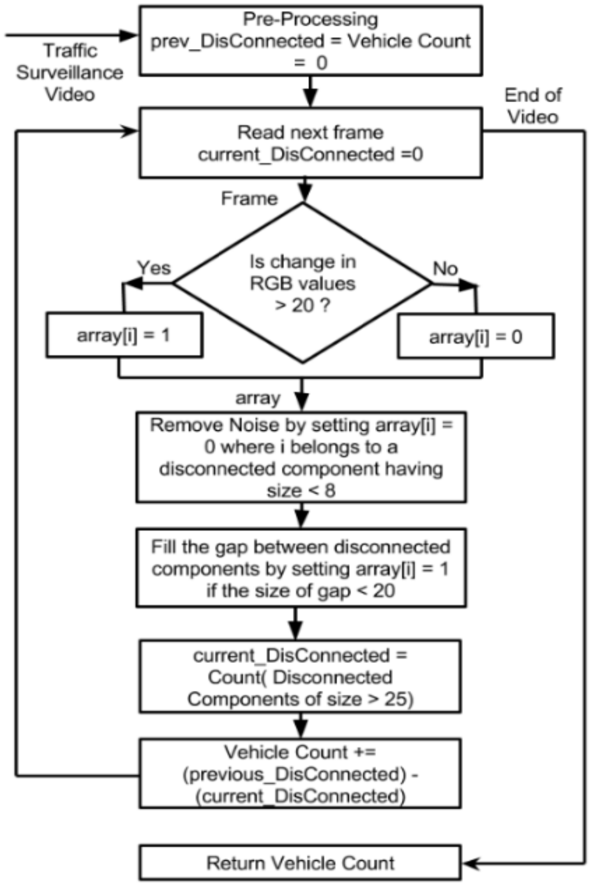

Figure-2 Vehicle Count Algorithm Flowchart

In Figure-2, the flowchart explains the improvised algorithm implemented to determine the vehicle count in the traffic surveillance video.

C. Scheduling Traffic

The dynamically changing vehicle count along each lane is made available to a thread, which is assigned the task of scheduling traffic among the four lanes. Initially, the signal at each lane is turned Red and the scheduling status of each lane is set to be unscheduled. A time of 15 sec is given to gather enough data for scheduling algorithm to make decision. Once the time lapses, the lane with maximum vehicle count is determined and is set to be the current scheduling lane.

Based on the vehicle count in this lane, reasonable amount of green signal time is calculated. The signal at current scheduling lane is turned Green and Scheduling thread is set to sleep for green signal time. When the thread gets back to running state, the scheduling status of current scheduling lane is set to be scheduled and the signal at this lane switches to yellow and then to red. After that, the decision phase for choosing next lane begins.

The algorithm which is proposed here is a priority based cyclic scheduling algorithm. As a result, when the decision phase for choosing the lane starts, only unscheduled lanes in that cycle are taken into account and vehicle count in those lanes at new instant of time is considered. Here, a cycle is said to be complete when all the lanes are scheduled. When a cycle completes, the signal at each lane is turned Red and the scheduling status of each lane is set to be unscheduled. The vehicle count at each lane is captured and decision phase for new cycle starts.

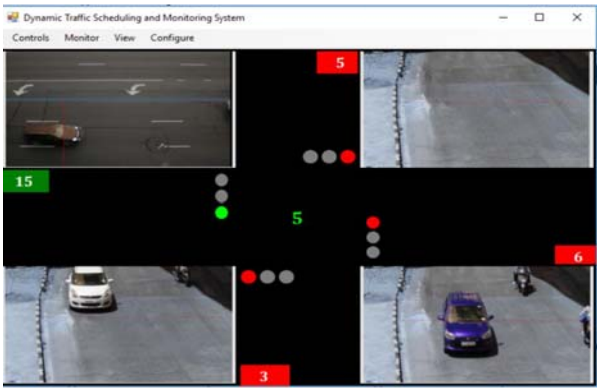

Figure-3 Sample Window Of Scheduling

In Figure-3, the scheduling part of the proposed system can be seen. Four videos will be playing simultaneously. The four threads, each assigned with a video, update vehicle count in a textbox next to the imagebox. The scheduling thread determines current scheduling lane and switches traffic lights accordingly. The timer shown at the center shows the remaining green signal time for the current scheduling lane.

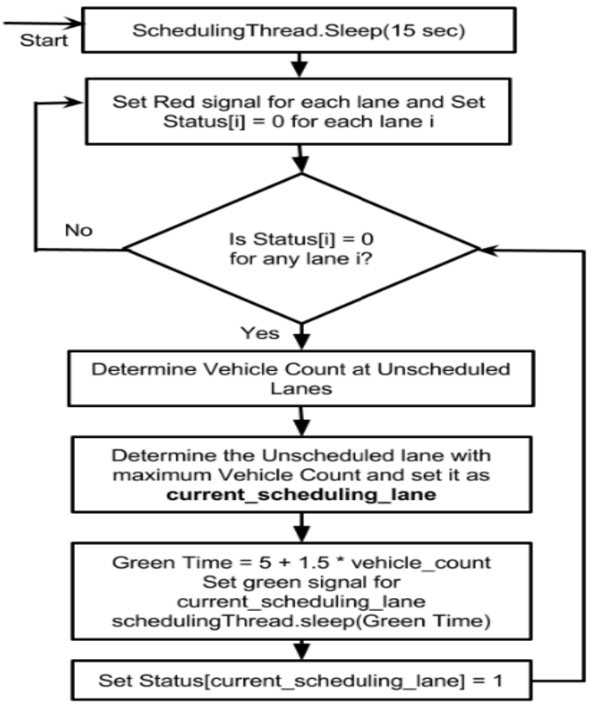

Figure-4 Scheduling Algorithm Flowchart

In Figure-4, the flowchart explains the Priority based Cyclic Scheduling Algorithm, implemented to schedule the traffic from each lane.

IV. EXPERIMENTAL RESULTS

In order to test effectiveness of the proposed method, the system was implemented in Visual Studio and tested on freeway surveillance video files, with various frames per second.

Table-1 gives the actual and detected number of vehicles in the video files we tested according to the method proposed in [8] and the current method.

TABLE-1 PERFORMANCE EVALUATION OF VEHICLE DETECTION

Input Video	Frames per second	Actual Number of Vehicles	Detected Number of Vehicles		Accuracy %	
			According to [8]	According to Current Method	According to [8]	According to Current Method
Video 1 [9]	30	17	13	17	76.47	100
Video 2 [10]	30	29	24	30	82.75	96.55
Video 3 [11]	30	27	22	29	81.48	92.59
Video 4 [12]	30	23	18	27	78.26	82.60
Video 5 [13]	30	15	11	15	73.33	100

The method proposed in [8] performs extremely well when there are less number of occluding vehicles and fails in the Indian scenario. This method is based on the idea of dividing single lane into sub lanes. In the Indian scenario, vehicles do not move in proper sub lanes and keep switching. So the proposed system in [8] scored 78.45% as most of the videos in the test dataset belonged to the Indian scenario. The current method of counting vehicles which does not divide lanes into sub lanes, handled occluding vehicles very well, and could achieve an accuracy of 94.348%.

V. CONCLUSION

In this paper, we have presented an algorithm to dynamically schedule traffic. The improvised vehicle counting algorithm has been found to be more accurate in counting vehicles. The scheduling algorithm accounts for real-time traffic and accordingly allots green light times. The results have shown that the system is accurate and feasible. The proposed system showed an accuracy of 94.348% for vehicle count while the scheduling algorithm was applied.

When the top view orientation over the lane of interest is taken from the video camera, the system generates more accurate results. Also the use of videos with higher frame rate and better quality produces better results. These conditions can be satisfied with ease in real applications, and hence system can yield accurate results.

VI. ACKNOWLEDGMENT

We are grateful to BMS College of Engineering for having supported and provided us with the facilities needed for the successful completion of this paper. The work reported in this paper is supported by the college through the TECHNICAL EDUCATION QUALITY IMPROVEMENT PROGRAMME [TEQIP-II] of the MHRD, Government of India

VII. REFERENCES

[1] Koller, Dieter, Kostas Daniilidis, and Hans-Hellmut Nagel. "Model-based object tracking in monocular image sequences of road traffic scenes." International Journal of Computer Vision,vol. 10, no. 3, pp. 257–281, Jun. 1993.

[2] M. Fathy and M. Y. Siyal, "An image detection technique based on morphological edge detection and background differencing for real-time traffic analysis," Pattern Recognition Letters, vol. 16, pp. 1321-1330, Dec'1995

[3] Coifman, Benjamin, et al. "A real-time computer vision system for vehicle tracking and traffic surveillance." Transportation Research Part C: Emerging Technologies, vol. 6, no. 4, pp. 271–288, Aug. 1998.

[4] Jaesik Choi, "Realtime On-Road Vehicle Detection with Optical Flows and Haar-Like Feature Detectors", Technical Report from Computer Science Research, University of Illinois at Urbana-Champaign, 2012

[5] Han, Sungji, Youngjoon Han, and Hernsoo Hahn. "Vehicle detection method using Haar-like feature on real time system." World Academy of Science, Engineering and Technology 59 (2009): 455-459.

[6] Zadeh, Lotfi A. "Fuzzy sets." Information and control 8, no. 3 (1965): 338-353.
[7] Patel, M., and N. Ranganathan. "IDUTC: An intelligent decision-making system for urban traffic-control applications." IEEE transactions on vehicular technology 50, no. 3 (2001): 816-829.
[8] Krishna, M. Poddar, Giridhar M K, A. S. Prabhu and Umadevi V, "Automated traffic monitoring system using computer vision," 2016 International Conference on ICT in Business Industry & Government (ICTBIG), Indore, India, 2016, pp. 1-5.
[9] https://drive.google.com/open?id=0ByCeV8sERlUfeVdadzY3THliTU0
[10] https://drive.google.com/open?id=0ByCeV8sERlUfWkpCMHIxTm1uNEU
[11] https://drive.google.com/open?id=0ByCeV8sERlUfeTd1N0d3NXN4ZFk
[12] https://drive.google.com/open?id=0ByCeV8sERlUfNzB5aUtaUGQxcVk
[13] https://drive.google.com/open?id=0ByCeV8sERlUfZUN2OVpQT2ZXWUk