Proceedings of the 2nd International Conference on Inventive Communication and Computational Technologies (ICICCT 2018)
IEEE Xplore Compliant - Part Number: CFP18BAC-ART; ISBN:978-1-5386-1974-2

Automated Traffic Monitoring Using Image Vision

R. Krishnamoorthy, Sethu Manickam
Dept. of Information Technology, BIT Campus,
Anna University, Tiruchirappalli, India – 620024
rkrish07@yahoo.com, sethu.sxcs@gmail.com

Abstract—**The process of traffic monitoring is predominantly carried out manually in our country. Traffic monitoring encompasses a set of stringent rules to be followed while ensuring that there are no traffic jams at the juncture of roads. In this paper, a novel method is proposed to automate the traffic signal lights with the assistance of multiple CCTV cameras connected over the Internet to survey various roads at the junction. The process comprises two primary phases: Vehicle Detection System and Traffic Scheduling Algorithm. Vehicle Detection shall be carried out in Digital Image Processing (DIP) by applying a simple kernel-based Edge Detection in Spatial Domain followed by an algorithm to detect the perimeter of closed figures while simultaneously applying the concepts of Machine Learning to classify the vehicle type into the following categories of motorcycle, light motor vehicle, and heavy motor vehicle. Subsequent processes are carried out in a novel Traffic Scheduling Algorithm through the help of a hybrid Round Robin having a dynamic time slice obtained by using Longest Remaining Job First to periodically update the traffic signal lights to relax the traffic. Instead of turning on the green light for a fixed amount of time, the duration will be managed dynamically based on the amount of traffic in each road. Thus, the proposed system is aimed at reducing the jamming of roads by a huge extent.**

Keywords—Digital Image Processing, Chain Linking, Closed Figure Identification, Canny Edge Detection,

I. INTRODUCTION

The most common way of maintaining an automated traffic monitoring system is by means of using CCTV cameras having a frame rate of 15 frames per second placed at strategic locations. Modern cities have been making use of this facility ever since the inception of IoT based cameras. To improve the method, certain specific algorithms need to be applied to reduce the complexity of the method. This is extremely important in order to produce results as quickly as possible. Currently, the traffic information is available in real-time from multiple systems by means of using inductive loop detectors, infra-red detectors, radar detectors and video-based systems [7]. Intelligent traffic monitoring system is the need of the hour. The method applied in our country is to make use of manual traffic control. Although, it is a very commonly applied as well as reliable method, it has a lot of shortcomings including the pressure of man power. Another problem is the inability to rationally count the number of vehicles in a particular road and reduce traffic congestion by prioritizing the roads having most number of vehicles. To count the number of vehicles automatically pressure plates may be used [5] and applying the concept of RFID placed on the vehicle number plates. The other approach is by traffic video surveillance. A very common practice of traffic monitoring is performed semi manually by traffic police sitting in a control room while monitoring the different traffic cameras. This is also not a very efficient method because the process involves man power. Thus, in this paper we are proposing a method to apply the concepts of Digital Image Processing (DIP) to identify the number of vehicles in each road at the traffic cross-section and thereby providing inputs to apply any method to schedule the traffic control. Our method does not emphasize on using high resolution video quality but applies a simplified edge detection followed by closed figure identification to count the number of vehicles in the least complexity possible. This paper proposes a two-step process to count the number of vehicles in each road at the cross-section followed by a Traffic Scheduling Algorithm to ease the traffic congestion at heavily populated road cross-sections. The output obtained from the Vehicle Identification Algorithm is used as input for the Traffic Scheduling Algorithm. The second involves the use of choosing from the most suitable scenario out of 12 possible traffic scenarios in case of traffic junctions of 4 roads have 3 dedicated lanes for Left, Straight, and Right.

II. RELATED WORK

In [2] Pena-Gonzalez et.al invented a scheme to track vehicles in real time and thereby classify them by involving a vision-based system. A HD-RGB camera is placed on the road to acquire data. At the same time clustering and classification algorithms are used to process information. Their method yielded an efficiency score over the 95 percent in test cases. Also, the system achieves 30 fps in image processing with a resolution of 1280x720. Another method was applied by Bhaskar et. al [3] for-Image Processing based vehicle tracking and identification. With a view to do improvements, it is proposed to develop a unique algorithm for vehicle data recognition and tracking using Gaussian mixture model and Blob Detection methods. Although highly efficient method, it is not adequate to control high volumes of traffic congestion. In the [4 - 5], Ali et. al and Kanungo et. al presented an alternative approach to intelligent traffic control switching algorithm. Traffic recognition was achieved using cascade classifier for vehicle recognition utilizing Open CV and Visual Studio. In Kanungo et. al, a comparison is drawn between conventional traffic management and their proposed scheme. The talking point about the system in [5] is that it is capable of being used in low light, thus making it Illumination Invariant Image Processing. In [1], Yu et. al's proposed scheme to normalize images having different illumination conditions wherein they iteratively estimate the relationship of the relative view and illumination of the images, transform the view of one image to the other, and normalize their illumination for accurate matching. Osmal et. al's system [6] of managing traffic at cross-section of roads using computer vision using a set of CCTV cameras along with a dedicated microcontroller for controlling the traffic lights. This method can be improved by using a centralized server and using a single piece of controller for controlling all roads at the cross-section simultaneously. Another aspect about vision-based traffic monitoring involves the concept of using low complexity algorithms. If an algorithm cannot process traffic signals in correct time, it will be deemed as flawed. Engel et. al [8] have worked on a low complexity algorithm to curtail the problems faced in traffic management. The problems are still evident in [8]. It follows a robust adaptive background segmentation strategy based on the Approximated Median Filter technique, which detects pixels corresponding to moving objects. The algorithm is found to handle multiple roads simultaneously. However, the shortcomings are that none of these systems use a 24x7 server to monitor traffic. Moreover, the concept of using server based fully automated traffic monitoring system is a very alien concept in our country. We have the resources available and this method is extremely viable to the general public.

III. PROPOSED WORK

The proposed method involves a two-step process to count the number of vehicles in each road at a cross-section. The steps are simplified edge detection followed by closed figure identification. The simplified edge detection is a process where we apply a kernel based Gaussian blur with the kernel matrix followed by kernel-based edge detection. The closed figure identification identifies the number of closed windshields in the given video frame. The counted number of vehicles in each road is then fed as input to the Traffic Scheduling Algorithm and then executes a specific algorithm to identify and update the prioritized queue periodically while dynamically slicing the round robin scheduling. The above steps are elaborated in the forthcoming subsections.

A. Overview

Videos are recorded by CCTV surveillance cameras having a frame rate of 15 frames per second placed to monitor each road at a cross-section. Frames are extracted at any instant and then our proposed method comes into effect. The first step is to apply a Gaussian Blur by kernel matrix followed by Kernel based Edge Detection. Gaussian Blur is applied to eliminate all the medium to minor edges. We only concentrate on the major edges. The major edges involve those of the boundaries of the vehicle and/or the boundaries of the vehicle's windshield. The resulting image can be converted to a binary mask having only 2 colour intensity values of 255 and 0. This helps us in reducing the complexity of the algorithm because the number of colour values we are going to work with are only 255 and 0. The number of search queries will also be around only 255 and 0, thereby considerably reducing the pattern matching time and enhancing our proposed algorithm's efficiency by a great amount. The number of vehicles is counted by counting number of closed figures that resemble a 3:1 aspect ratio similar to the ones found at windshields of vehicles in a trapezium to rectangle form. The number of vehicles is approximated because there can be some problems pertaining to long distant vision but the traffic scheduling algorithm has been designed such that it will not compromise on the result of the output produced by the traffic scheduler. The various sub steps involved are Simplified Edge Detection followed by Closed Figure Identification which is then executed by Traffic Scheduling Algorithm to obtain the results.

B. Moderated Canny Edge Detection

The frames obtained from video surveillance are subjected to a Kernel-based Gaussian Blur in order to reduce the minor edges and noise. We propose to work only on the major horizontal and vertical edges because the car edges are not minor edges. The windshields of cars also have a major colour variation; hence they cannot be classified as minor edges. Thus, we are reducing a large number of verifiable colour pixels by applying the kernel matrix to apply Gaussian Blur. The kernel matrix for Gaussian Blur is given as follows:

$$M_g = \begin{bmatrix} 0.111 & 0.111 & 0.111 \\ 0.111 & 0.111 & 0.111 \\ 0.111 & 0.111 & 0.111 \end{bmatrix}$$

The matrix used is a 3x3 one. This matrix was chosen based on a series of research that we conducted to identify the most optimal matrix by keeping the light intensity same for frames extracted from a 2 MP resolution CCTV camera having a frame rate of 15 frames per second. A convolve operation is performed which converts the given image into a blurred version after excluding the minor edges while keeping the major edges intact. The process of summing the elements of the image to its local neighbours is known as convolution, weighted by the kernel. This is related to a form of mathematical convolution.

The mathematical formula to apply Gaussian Blur using kernel is given as follows:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \text{; where, } \sigma = \sqrt{\frac{2}{\pi}} \quad (1)$$

The next sub process involved in edge detection is to apply Canny Edge Detection [9]. This method is applied because it was found to produce the most optimal values for a 2 MP resolution CCTV camera having a frame rate of 15 frames per second at a decent computational complexity.

After the application of Gaussian Blur and Canny Edge Detection, we propose to convert the entire image into 2 colour intensity values of 255 and 0. This sub step is instrumental in providing a very sleek system which has low computational complexity while not compromising on quality of the image. The binary mask is constructed using simple entry control commands which check each pixel if the colour intensity value is greater than 0. This indicates that it is an edge, thus changing the value of that pixel intensity value to 255. The control statement is composed as follows:

if $^{\text{intensity}}\text{Img}_{i,j} > 0$,

then, $^{\text{intensity}}\text{Img}_{i,j} = 255$

C. Closed Figure Identification by Chain Linking

After obtaining the binary masked images through the application of Simplified Edge Detection, the resulting step aims to identify the closed contours that resemble a windshield of a car or a truck or a bus. By this method, we can count the number of windshields in the given road and thereby count the number of vehicles. This process is achieved by the help of our proposed Closed Figure Identification algorithm which is similar to chain linking in Digital Image Processing followed by pattern matching to verify if the identified closed contour matches with a sample windshield of a car or a truck or a bus. The closed figure identification follows the path of neighbour identification to trace out the path of the closed contour.

$$\text{Perimeter} = \sum_{i=0,j=0}^{n,n} \sum_{k=1}^{n} \binom{intensity}{i,\ j} img^k \quad (2)$$

Each $\text{Img}_{i,j}$ is subjected to a number of neighbours check using the following test cases - $\text{Img}_{i-1,j}$, $\text{Img}_{i,j-1}$, $\text{Img}_{i+1,j}$, $\text{Img}_{i,j+1}$. The number of neighbours is noted and the same function is recursively called to find out the number of neighbours each $\text{Img}_{i,j}$ has and the area occupied by the vehicles in each road is computed. The flow diagram is shown in figure 1, explaining the various looping conditions involved in order to achieve our objective.

The recursive function is used to identify the number of neighbouring pixels which have the intensity value of 255. If there are no neighbours having intensity value 255, then the loop is continued to iterate to the next j value and subsequently the next i value to obtain the next set of neighbours in the image pool. The iteration is stopped and the vehicle_count variable is incremented only when the i, j value of $^{\text{intiial}}\text{Img}$ and $^{\text{final}}\text{Img}$. This signifies that a loop has been established which invokes the aspect ratio condition of 3:1 of a windshield.

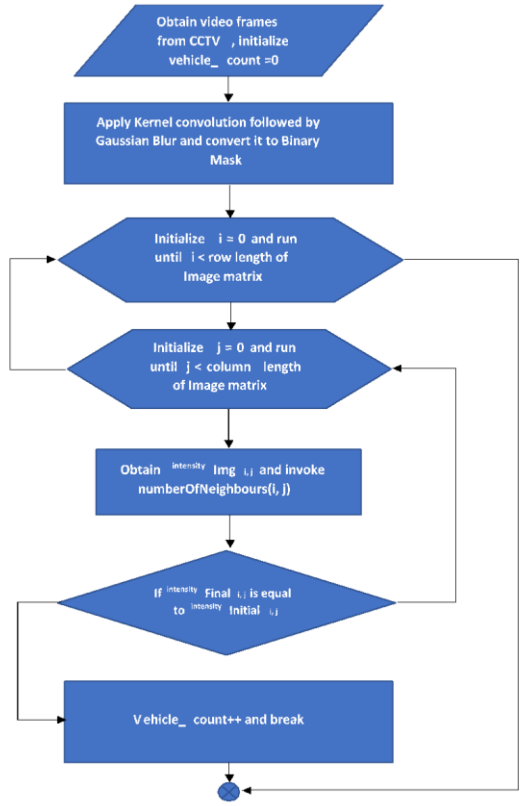


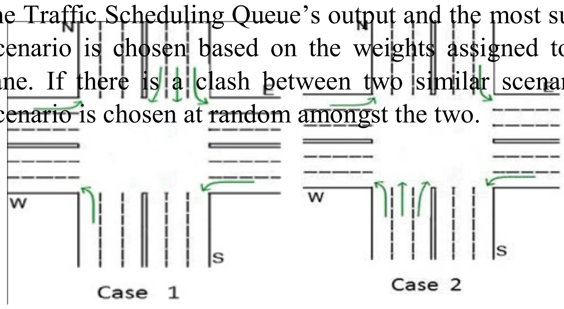
Fig 1 – Flowchart to identify closed figures

D. Traffic Scheduling Algorithm

The number of vehicles in each lane is collected by the traffic scheduling algorithm and the lanes are sorted in descending order of number of vehicles in each lane. This queue is termed as Traffic Scheduling Queue. Dynamic Time Slicing is performed based on the priority assigned in this queue by using the formula

Ti = $\{R_i/(R_{avg})*90 + 30\}$; where R_i = No. of vehicles in lane i and R_{avg} = Average no. of vehicles per lane (3)

Based on the queue generated by the Traffic Scheduling Queue, the various scenarios of traffic at a traffic junction can be of 12 types. Taking each direction, North (N), East (E), South (S), West (W) into account each vehicle will be able to either Turn Left (L), Go Straight (G), or Turn Right (R). The 12 scenarios are shown in figure 2.



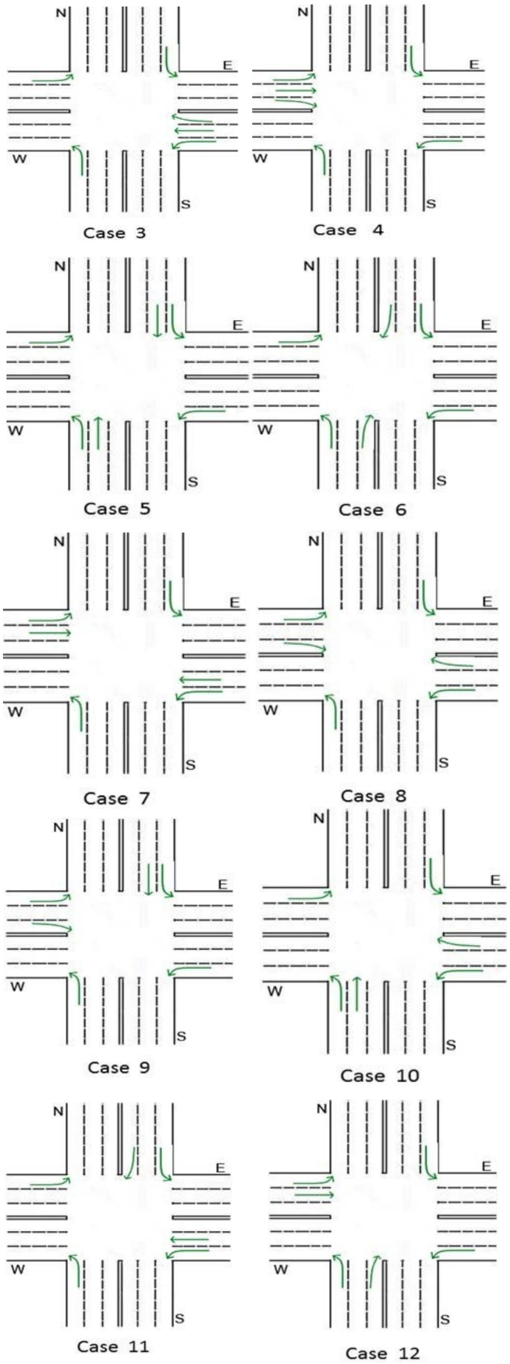


Fig 2 – The twelve possible scenarios at a traffic cross-section with 3 lanes each

The most optimum scenario is computed based on the Traffic Scheduling Queue's output and the most suitable scenario is chosen based on the weights assigned to each lane. If there is a clash between two similar scenarios, a scenario is chosen at random amongst the two.

IV. ALGORITHM

A. Vehicle Detection

Input: CCTV Video Frame

Output: Number of vehicles present in the given frame

Begin.

Step 1: Obtain the frame from CCTV surveillance.

Step 2: Apply Kernel-based convolution to perform Gaussian Blur.

Step 3: Find edges using Kernel matrix and convert to binary mask.

Step 4: Initialize loops i and j to traverse through Img matrix.

Step 5: Obtain $^{intensity}Img_{i,\ j\ and}$ invoke numberOfNeighbours(i, j) function to count number of neighbours around $Img_{i,\ j}$ and recursively call the function to proceed further.

Step 6: Execute Step 5 until $Final_{i,\ j}$ is equal to $Initial_{i,\ j}$ and calculate the area of the closed contour.

Step 7: Output the number of closed contours matching that of a windshield of a car, or a truck, or a bus and increment the car_counter variable.

End.

B. Traffic Scheduling Algorithm

Input: Queue length of n roads

Output: Dynamic green light duration for each road

Begin.

Step 1: Receive queue length data from Vehicle Counter for Roads $\sum_{i=1}^{n}(R_i)$.

Step 2: Set up a round robin of time slice varying between 30 and 120 seconds based on {Ri/(Ravg)*90 + 30} and traverse through circular queue.

Step 3: Set up queue in descending order and begin Round Robin scheduling.

Step 4: Update the queue lengths by invoking Vehicle Counter again and go to Step 2.

Step 5: Extract current video frames and invoke vehicle counting algorithm and go to Step 1.

End.

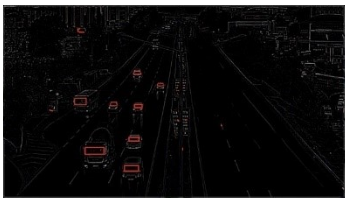
V. RESULT ANALYSIS

The frames obtained from 2 MP CCTV Cameras were subjected to the proposed method and the results obtained are shown as follows. Further observations were made on over 50 video frames extracted from a CCTV surveillance footage of a 4-road traffic cross-section. A sample is shown in Figure 3A. The result of Moderated Canny Edge Detection is shown in Figure 3C. Our proposed scheme returned the number of vehicles as 8 as shown in Figure 3D. The results were found to be mostly accurate. The computational time was found to be less than 1 second and it is optimum because the traffic wait time is matched by the computation time. The second half of the proposed system implements the Traffic Scheduling Algorithm with dynamic time slicing as mentioned in (3). The starvation time is also critically low which indicates that the algorithm is working positively. The overall system is found to be extremely efficient in processing the output signals in real-time.

3A. Frame Extracted from CCTV

3B. Gaussian Blur

3C. Edge Detected Image



3D. Closed Figure Identification

Fig 3A. Frame Extracted from CCTV – original image frame which is subjected to the proposed method. Fig 3B. Gaussian Blur – Gaussian Blur is applied with kernel matrix of 3x3. Fig 3C. Edges are detected after blur convolution is applied. Fig 3D. Close figures (windshields) are identified and the perimeter and the number of windshields is counted.

VI. CONCLUSION

In this paper, a novel method is applied to count the number of vehicles in each lane at a traffic cross-section. The points taken into consideration are low complexity, greater efficiency, and faster process time without compromising on the accuracy of the results. The obtained results are encouraging and can be scaled to broader roads with more lanes. The outputs obtained from our proposed method can be used to process the Traffic Scheduling Algorithm, thereby achieving the objective of intelligent traffic monitoring system. The resources used are 2 MP CCTV cameras having a frame rate of 15 frames per second. In that respect, the proposed system is found to be extremely efficient.

VII. REFERENCES

[1] Yinan Yu, Kaiqi Huang, Wei Chen, and Tieniu Tan, “A Novel Algorithm for View and Illumination Invariant Image Matching”, IEEE Transactions on Image Processing Year: 2012, Volume: 21, Issue: 1, Pages: 229 – 240.

[2] Raul Humberto Pena-Gonzalez, Marco Aurelio Nuno-Maganda, “Computer vision based real-time vehicle tracking and classification system”, 2014 IEEE 57th International Midwest Symposium on Circuits and Systems(MWSCAS), Year: 2014, Pages: 679 – 682.

[3] Prem Kumar Bhaskar, Suet-Peng Yong, “Image Processing Based Vehicle Detection and Tracking Method”, 2014 International Conference on Computer and Information Sciences (ICCOINS) Year: 2014, Pages: 1 – 5.

[4] Ihtisham Ali, Arsalan Malik & Waqas Ahmed, Sheraz Ali Khan, “Real-time Vehicle Recognition and Improved Traffic Congestion Resolution”, 2015 13th International Conference on Frontiers of Information Technology (FIT), Year: 2015, Pages: 228 – 233.

[5] Anurag Kanungo, Ayush Sharma, Chetan Singla, “Smart Traffic Lights Switching and Traffic Density Calculation using Video Processing”, 2014 Recent Advances in Engineering and Computational Sciences (RAECS), Year: 2014, Pages: 1 – 6.

[6] Tousif Osman, Shahreen Shahjahan Psyche, J. M. Shafi Ferdous, Hasan U. Zaman, “Intelligent Traffic Management System for Cross Section of Roads Using Computer Vision”, 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), Year: 2017, Pages: 1 – 7.

[7] Taqi Tahmid and Eklas Hossain, “Density Based Smart Traffic Control System Using Canny Edge Detection Algorithm for Congregating Traffic Information”, 2017 3rd International Conference on Electrical Information and Communication Technology (EICT), Year: 2017, Pages: 1 – 5.

[8] Juan Isaac Engel, Juan Martín, and Raquel Barco, “A Low-Complexity Vision-Based System for Real-Time Traffic Monitoring”, IEEE Transactions on Intelligent Transportation Systems, Year: 2017, Volume: 18, Issue: 5, Pages: 1279 – 1288.

[9] John Canny, “A Computational Approach to Edge Detection”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Year: 1986, Volume: PAMI-8, Issue: 6, Pages: 679 – 698.