Journal of Science, FAS-SEUSL (2020) 01(01) 12-21
ISSN:2738-2184

Highways Traffic Surveillance using Internet Protocol Cameras and Open Source Computer Vision library

H. M. M. Naleer and J. A.G. G. Jayasinghe

Department of Mathematical Sciences, Faculty of Applied Sciences, South Eastern University of Sri Lanka, Sammanthurai, Sri Lanka

Abstract
Because of road traffic and traffic congestion, the development of traffic surveillance systems with multifunctional techniques has received increasing attention. Vehicle detection, tracking, classification and tally is extremely necessary for military, civilian and government applications, like road watching, traffic prediction, toll assortment and traffic flow. For traffic management, vehicle detection is that the vital step. This paper presents a real-time management and control system that serves to analyze road traffic using an IP camera. The programming method enforced with python artificial language with functional programming of OpenCV which could be operated under both, Windows and Linux OS. During this paper, we tend to gift cheap, transportable and Computer Vision primarily based systems for moving vehicle detection and tally. Image from video sequence is taken to observe moving vehicles. The system is enforced mistreatment OpenCV image development kits and experimental results are incontestable from video dataset. The traffic counting method has been developed by background subtraction, image filtering, image binary and segmentation ways are used. this method is additionally capable of tally moving vehicles from videos. This paper will also examine the result of the computer vision programming under GNU Linux. The experimental results show that the proposed method can achieve more than 97% accuracy of vehicle counting.

Keywords: Vehicle detection, Tracking, Tally, OpenCV.

1. Introduction

Traffic watch is one in all the key half in Intelligent Traffic Systems. Traffic watching offers solutions to the majority of the issues moon-faced by individuals. It conjointly makes policing and dominant of traffic easier. Video police investigation is one in all the technologies which may be used for traffic watching. Video police investigation is also the foremost economical choice that doesn't involve major prices or infrastructural changes

Road traffic is one in all the key issues in Ceylon and worldwide. traffic jam is seen all the time in countries across the globe. Researchers around the globe area unit analyzing to solve this drawback. varied alternatives and ways area unit looked at to solve or at least cut back this drawback. Traffic police investigation is additionally seen as the choice to solve the matter. Intelligent Traffic Systems solves the matter like incident detection, traffic watch, traffic rules violations, live traffic updates, automotive vehicle mated traffic signal. Intelligent traffic system management and higher access to real-time on with historic info helps commuters to arrange their route. It may facilitate to cut back congestion. Loop detectors, video cameras, mobile sensors like

hmmnaleer@gmail.com

GPS will be used for traffic watching. Putting in loop detectors involve high installation price, high maintenance price conjointly changes in road infrastructure. Mobile sensors like GPS etc. involve high price issues from the commuter purpose of read [1]. In distinction of these, putting in video cameras and watching traffic mistreatment it's the higher choice because it doesn't need high installation, maintenance price conjointly no personal price for the commuters.

Object recognition and detection is an important part of this research. The basic goal is to find an object in static images or video frames. for complex object behaviors, such as illegal vehicle movement, it is hard to find features and heuristics that will handle the huge variety of instances of the object class [2]. Thus, vehicle detection in a cluttered environment is an open problem. If the appearance is polluted by noise, recognition errors will be produced [3].

The use of implicit models is explained in [4]. In [4] the author supposes to use a learning Ada Boost algorithm which is robust and fast by making a lot of cascaded weak decisions. In [4] they train a support vector machine with the SIFT descriptors of selected cars and non-cars. But both of these approaches have to be trained with lots of positive and negative samples before working independently. Additionally, it is not easy to cover all cases of illumination and the environment. That's why many learning algorithms have to be trained for every situation separately. Another easy approach for the detection of moving cars without using any model is explained in [5], where they detect all moving objects in adjacent images by computing the normalized difference image. But as the geo-registration of the images often is less exact than the pixel size, the images have to be co-registered first. On the other hand, only moving objects can be detected while traffic jams or queues in front of a traffic light would be ignored [6]

In this paper, we propose a vehicle surveillance system to detect, track, and recognize vehicles from different video sequences. In this system, only one camera, still mounted on a pole and looking down on the observed highway, is used to monitor various vehicles. In the beginning, different vehicles are extracted from video sequences using the technique of image subtraction. Since this technique is sensitive to lighting changes, an adaptive background updating method is used for modeling the background.

To count the vehicle by implementing an algorithm programmed in OpenCV (Intel® open-source Computer vision library), Tracking objects will be achieved by using a smart camera which captures the frames and detecting the illegal driving of the vehicles from highway and motorway then send the images signals to the main server. Images might also be stored in a server for future usage and at the same time, the smart camera will perform image processing for potential dangers. From all above the target of the system will be; Direct streaming, updating background directly frame by frame, real-time tracking all moving vehicles by put them inside a box then detect and save all the illegal driving of the vehicles. And don't forget that the system could work in any weather and during any time of the day.

The main goal of this project was to detect and count vehicles from an IP camera feed and to ultimately give an idea of what the real-time on-street situation is across the road network. Then save video data set and detailed data set to future works like traffic prediction, toll assortment and traffic flow.

This paper consists of the following sections, literature review, methodology, results and discussion followed by Conclusion. The section literature review discusses about the related works. The section methodology deals with focuses on architecture, vehicle detection and vehicle counting. The experimental results are discussed in the result & discussion. Finally, we give a brief conclusion.

2. Literature Review

A variety of research papers and articles have been published regarding the traffic analysis in the highway. In [7] has shown that in Anuradhapura Sri Lanka causes for accidents included wrong driving/riding 54(25%), other vehicle collided 46(22%), animal crossing road 39(18%), mechanical failure 14(7%), poor road 18(9%), glare 4(2%), man crossing road 8(4%), garment trapping the wheel 5(2%), rain 6(3%). They are studied from 214 consecutive patients with an age range of 01- 75 years. Also in [8]. has enhanced that factors are influencing on total traffic accident in varies statistical amount such as overtaking 31%, speeding 9%, drunken driving 14%, carless of pedestrians 9%, road conditions 4% carless of drivers 7% and others 9%. In [9], an advanced road vehicle counting system is to replace the old systems based on ILD was developed. They suggest that Multicore programming allows achieve real-time performances with only a piece of software. In [10] has suggested that a vehicle recognition system is a good manner to help solving some of the problems like as safety (crime prevention, surveillance problems), Intelligent Transport Systems (driver assistance systems, intelligent parking systems) and Traffic Management (Traffic Parameters). In [11] has suggested that a vehicle recognition system is a good manner to help solving some of the problems like as safety (crime prevention, surveillance problems), Intelligent Transport Systems (driver assistance systems, intelligent parking systems) and Traffic Management (Traffic Parameters). In [12] the vehicle tracking scheme is achieved by using readings from two 3-axes magnetic sensors. In this research, they state that real-life experiments showed a good performance for vehicles on the same side of the road as the installation spot of the sensors, while they gave less performance for vehicles on the other lane. Also, Large vehicles created difficulties for the tracking scheme due to the invalidation of a single dipole moment model for such vehicles. In [13] Used discrete Fourier transform (DFT) and principal component analysis (PCA) to automatically extract features from magnetic signatures. In [14], the researchers explored the effect of the number of GCP, camera viewing angle and the deflection angle on the accuracy of the detected vehicle trajectories. An important issue that was addressed is how error in video capture is affected by the field of view (horizontal camera viewing angle) and the sightline deflection from the perpendicular concerning the roadway. The results suggest that the camera viewing angle did not have a significant effect on tracking error. On the other hand, the deflection angle was found to have a significant effect on this error for the angles chosen in the survey. Also, they found an error in the video image processing algorithm is caused by camera placement (height and offset distance) and by the deflection of the line of sight concerning the roadway. In [15] Vehicle detection is performed using either motion silhouettes or 3DHOG (3D extended Histograms of Oriented Gradients). [16] Paper presents a real-time management and control system that serves to analyze road traffic using a stationary camera. In [17] Detection algorithm was designed using both Canny and Gaussian protocols. The system is developed using different processor speed and the system developed under GNU-Linux with C programming integrity with OpenCV functions. The results are beyond the expectation as its purpose for traffic surveillance system.

3. Methodology

3.1. Synopsis

In this work, we have developed a system that automatically detects and counts vehicles. The proposed system consists of four main functions: capture footage, vehicle detection, vehicle counting and saving the details. The input of the system is, for instance, a video footage (in the current version of the system, we use a prerecorded video), while the output of the system is an absolute number of vehicles. The following sections describe the different processing steps of the system.

3.2 Architecture

The brain of an IP camera is a special processing module that performs application-specific information processing. The design of a smart camera as an embedded system is challenging because video processing has an insatiable demand for performance and power, but at the same time, embedded systems place considerable constraints on the design. In this system, the IP camera (Internet protocol camera) used to surveillance the traffic on the highway. The IP camera is a type of digital video camera that receives control data and sends image data via the internet. Unlike analog CCTV cameras, they require no local recording devices. The IP camera captures the footage and sends it to the system for the object detection and vehicle counting process.

The system of Traffic surveillance developed by using the IP camera in real-time large-scale computerized city motorway system and highway road monitoring and controlling. This could be achieved by installing camera sensors on the highway and at certain junctions to monitor highway users & pedestrians. However, it monitors road illumination and other environmental and security problems. The camera processor units able to monitor the highway and the information from the cameras will be sent to the main server which is responsible for monitoring the highways of different parts of the city is installed. This station is placed as a base station where it receives the camera displays through necessary switching units.

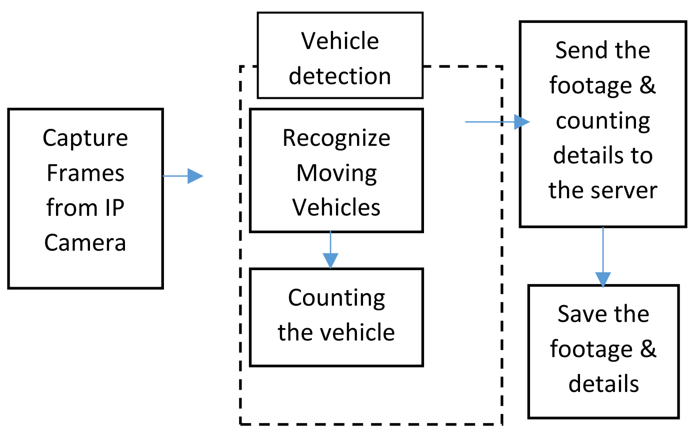


Fig. 1. Architecture of the system

Figure 1 shows the architecture of the system used. The steps used to solve the problems found in the system process are as follows:

1. Distribute a bunch of IP cameras on route.
2. Check the ability and energy of every node by causing signal to the server that's accountable of checking the nodes state of affairs
3. Use some recognitions protocols and functions like blob detection to acknowledge the sides of the objects. Then get means to classify the objects within the road.

4. Install new programming methodology that may be a massive library of intelligent algorithms OpenCV and GNU-Linux underneath Ubuntu to acknowledge the abnormal objects state of affairs within the image.
5. Send the image information to the bottom station by causing signals to the closest node that results in the bottom station.
6. Analyze the signal and information received then send that individual signal (image of the object and its information) from the bottom station to the users.

3.3 Object Detection

To count vehicles, we tend to observe them in a picture. this can be pretty straightforward for an individual to select out however tougher to implement within the machine world. However, if we have a tendency to contemplate that a picture is simply Associate in Nursing array of numbers (one price per pixel), we have a tendency to could also be ready to use this to work out what a vehicle sounds like and what we'd expect to check once there is not a vehicle there. we will use OpenCV to seem at however the worth of sure pixels' changes for these 2 conditions, as shown within the image below. To do this, we tend to should first translate our image from RGB channels (Red, inexperienced Blue) to HSV (Hue,Saturation, Value) and examine every channel to check if it will tell America one thing.

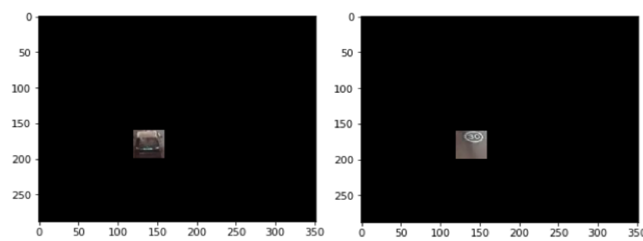


Fig. 2. Object detection

As we can see from figure 3 the Hue channel doesn't supply a lot of data, whereas each the Saturation and price channels clearly show a distinction between the Vehicle/No Vehicle conditions so we can use these channels in our detection algorithmic rule.

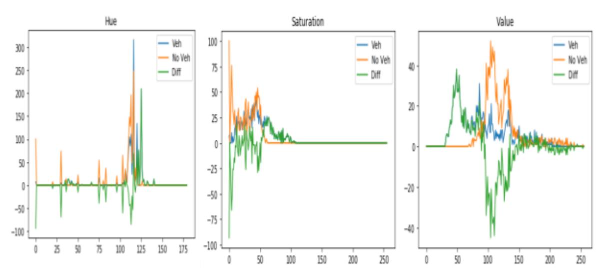


Fig. 3. The bar graph plots

We can then use this data to work out what's background and what's a vehicle. Within the case shown here, it's troublesome to get a transparent image, but we can use OpenCV to average between many frames and make our background image.

Fig. 4. Highway image in grayscale

Now that we have a background image or AN array of default background values, we can use OpenCV to notice once these prices go higher than a precise value (or 'threshold value'). we tend to assume that this happens once there's a vehicle at intervals that constituent, so use OpenCV to line the pixels that meet the edge criteria to most brightness.

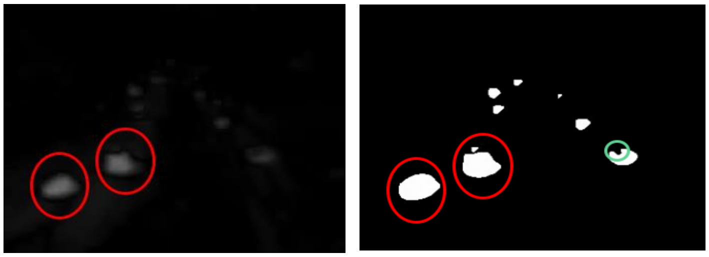


Fig. 5. Blob detection

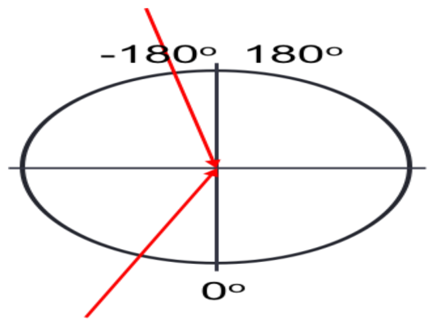
The images in figure 5 were higher than the pixels that meet the edge criteria (left) and therefore the ensuing shapes once setting those pixels to most value/brightness (right). conjointly highlighted (green) is gaps in our objects wherever dark areas (windscreens, grills, etc) might not meet our threshold criteria. this might cause a haul soon thus we tend to attempt to fill in these gaps victimization the erosion and dilation functions from the OpenCV library.

Once we tend to square measure proud of the shapes created, we should then check the shapes (or contours) to work out that square measure most wishes to be vehicles before dismissing those who don't seem to be. we can square measure to try this be implementing a condition wherever we tend to are solely inquisitive about the detected contours if they're over a precise size.

3.4 Counting Vehicles

The vehicle counter is split into 2 category objects, one named vehicle which is employed to outline every vehicle object, and also the other vehicle counter that determines which 'vehicles' are valid before counting them (or not). Vehicle is comparatively easy and offers data concerning every detected object like a caterpillar-tracked position in every frame, what percentage frames it's appeared in (and what percentage it's not been seen for if we tend to quickly lose track of it), whether or not we've got counted the vehicle nevertheless and what direction we tend to believe the vehicle to be travel in. we will additionally acquire the last position and also the position before that to calculate several values at intervals our vehicle counter algorithm.

Vehicle Counter is a lot of complicated and serves many functions. we will use it to work out the vector movement of every self-propelled vehicle from frame to border, giving associate indicator of what movements are true and that are false matches. we tend to try this to form certain we're not incorrectly matching vehicles and thus obtaining the foremost correct count doable. during this case, we tend to solely expect vehicles to travel from the highest of the image to rock bottom paw corner or the reverse. this suggests we tend to solely have a precise vary of allowable vector movements supported the angle that the vehicle has touched - this could be seen from the figure 6. The image shows the expected vector movements (highlighted in red) and also the figure 7 shows a chart of distance touched vs the angle - those classed as allowable movements are highlighted by the inexperienced box.



.

Fig. 6. Vector movement

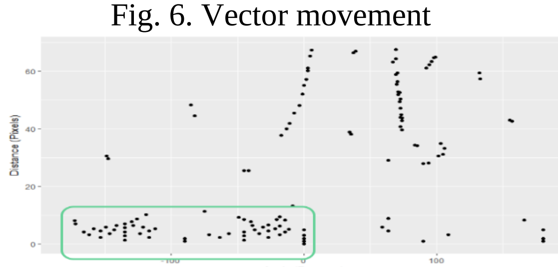


Fig. 7. Touched vs angle chart

If a vehicle object satisfies the on top of criteria, we then need to visualize what direction it's occupancy before then passing it to the counter. we will then use this data to work out whether or not the vehicle ought to be counted and so whether or not the count applies to the left lanes (up direction) or right lanes (down direction). Once glad, we tend to update the counter and print it to the output frame. If a vehicle has not been seen for a minute, we take away it from the list of tracked objects because it isn't any longer of interest.

4. Results and Discussion

This section discusses the results achieved by using the surveillance system, one of the results that get after running the program that coded by OpenCV, is counting the numbers of moving vehicles in the view by tracking the moving objects for all frames helps in surveillance system and solve the road capacity problem. They are tested with footage obtain from Beckton west road traffic surveillance. As you can see in figure 9 the system is tested with both day and night time footages.

Fig. 8. PNG format image in grayscale

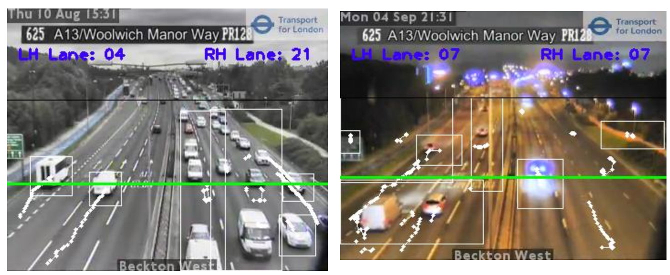


Fig. 9. Counting vehicle in the system

Figure 8 shows the converting of the image from RGB to Grayscale. Figure 9 shows how the result of the counting vehicle describes in the system running situation. The first image shows the real-time traffic analyzing in the day time and the second image shows night time traffic analyzing. After the camera connected has been detected, the capturing operation will be started and the system will be established. The results above had proved based on OpenCV programming under the GNU-Linux Ubuntu operating system.

An analytical study was conducted to calculate the vehicle counting accuracy to test the efficiency of our system. Table 1 presents the results calculated from day time video data.

Table 1: Vehicle counting

Sequence	Actual Number of vehicle		Recognized By system		Accuracy	
Lane	RH	LH	RH	LH	RH	LH
Video 1 Day	6	9	6	9	100%	100%
Video 2 Day	203	182	199	176	98.02%	96.70%
Video 3 Day	1149	996	1126	958	97.99%	96.18%

Total Day	**1358**	**1187**	**1331**	**1143**	**98.01%**	**96.29%**
Video 4 Night	13	15	12	14	92.37%	93.33%
Video 5 Night	32	32	29	29	90.62%	90.62%
Total Night	**45**	**47**	**41**	**43**	**91.11%**	**91.48%**

The average accuracy measured in the test conducted on three different footages at day time in the right hand (RH) side had as value 98.01% and the left hand (LH) side is 96.29%. The average accuracy measured in the test conducted on three different footages at night time in the right hand (RH) side had as value 91.11% and the left hand (LH) side is 91.48%. The Total average accuracy of day time is 97.21% and the total average accuracy of night time is 91.30%. The detection results of two-wheeled vehicles are worse than other vehicles. The number of false negatives can be explained by several factors. Their small size, their high speed, and their no rectilinear trajectories make them difficult to track. The night time results show 8.69% of error. This can be explained by vehicle lights and road lights.

In spite of difficulties induced by road marking and shadows due to road panels, moving tree branches, and vehicle lights, the detection ratio remains very satisfactory, 97.00%, with 2558 vehicles detected out of 2637. Regarding the classification performance, despite the simplicity of the criterion (width of the blob), results are rather satisfactory. For all the datasets.

4. Conclusion

This paper has proposed a highway traffic surveillance system. The proposed system can detect video sequences. In other words, the system can measure the number of vehicles in real-time based on segmentation, detection. The videos were captured by the IP cameras. OpenCV library was used to detect the vehicle. In the system, the video background was directly updated frame by frame, which is useful for doing background subtraction. In the testing stage, the system showed 93% accuracy. Despite the considerable success in the vehicle detections, there are some problems during the experiments. In testing of the night time, the system shows some difficulties to identify vehicle due to road light and vehicle light. The System is developed under GNU-Linux with python programming integrity with OpenCV functions. The result as attach in this paper beyond the expectation as its purpose for traffic surveillance system.

As future work, we intend to explore a new algorithm to further optimize (in terms of speed) vehicle detection and explore new algorithms to license plate detection. Additionally, we plan to explore the traffic prediction model as our new dataset provides such information.

References

1. Bruzzone, D. F. P. S. B. S. L. (1999) A neural-statistical approach to multitemporal and multisource remote-sensing image classification. IEEETrans.Geosci.RemoteSens., 37(3), 1350–1359.
2. Intel, learning_vision. Intel Technologies Journal., 09 (2), 04.
3. Bertozzi, A. B. a. A. F. M. (2000) Vision-based intelligent vehicles: State of the art and perspectives. Robot. Auton. Syst., 32(1), 1–16.
4. Grabner, C. H. (2008) Semi-supervised On-Line Boosting for Robust Tracking. Berlin.
5. Reinartz, P. (2006) Traffic monitoring with serial images from airborne cameras. ISPRS Journal of Photogrammetry and Remote Sensing., 61(3-4), 149-158.
6. Wafi, R. P. Z. (2010) Highways Traffic Surveillance System (HTSS) using OpenCV. in 2010 IEEE Control and System Graduate Research Colloquium (ICSGRC 2010).
7. Weerawardena, I. P. R. S. N. (2013) Analysis of patients admitted with history of road traffic accidents to surgical unit B Teaching Hospital Anuradhapura, Sri Lanka. Anuradhapura Medical Journal., 7(1), 2-5.
8. Aslam, M. (2015) An econometric analysis of trafic accident in Sri Lanka. in Proceeding of International Conference on Contemporary Management - 2015 (ICCM-2015).
9. Crouzil, L. A. (2016) Automatic Vehicle Counting System for Traffic Monitoring. Journal of Electronic Imaging, SPIE and IST., 25(5), 1-12.
10. Santos, D. (2008) Automatic Vehicle Recognition System. in Extended abstract of Master's degree dissertation.
11. Nemade, B. (2016) Automatic Traffic Surveillance using Video Tracking. Procedia Computer Science., 79, 402 – 409.
12. W. B. a. T. G. Ali M. H. Kadhi m. (2013) Evaluation of Vehicle Tracking for Traffic Monitoring Based on Road Surface Mounted Magnetic Sensors. in 3rd IFAC Symposium on Telematics Applications The International Federation of Automatic Control, Seoul, Korea.
13. Chang Xu, Y. W. X. B. a. F. L. (2018) Vehicle Classification Using an Imbalanced Dataset Based on a Single Magnetic Sensor. Sensors., 18, 1690.
14. Giuseppe Guidoa, A. V. F. F. S. V. A. V. G. (2013) Vehicle tracking system based on videotaping data. in 3rd IFAC Symposium on Telematics Applications the International Federation of Automatic Control November. Seoul, Korea.
15. Norbert Buch, M. C. J. O. a. S. A. V. (2009) Vehicle location and classification in urban cctv streams. in ITS World Congress 2009.
16. Zakaria Moutakki, I. M. O. K. A. A. A. (2018) Transport and Telecommunication., 19(2), 93–102.
17. Zainab Nazar Khalil Wafi, R. A. P. M. (2010) Highways Traffic Surveillance System (HTSS) using OpenCV. in 2010 IEEE Control and System Graduate Research Colloquiu.