Vehicle Detection, Tracking and Counting

Safoora Maqbool, Mehwish Khan, Jawaria Tahir, Abdul Jalil*
Department of Electrical and Electronics Engineering
International Islamic University
Islamabad, Pakistan
e-mail: abdul.jalil@iiu.edu.pk

Ahmad Ali*
Department of Electrical and Electronics Engineering
Pakistan Institute of Engineering and Applied Sciences
Islamabad, Pakistan
e-mail: ahmadali1655@hotmail.com

Javed Ahmad
Military College of Signals
National University of Sciences and Technology
Rawalpindi, Pakistan
e-mail: javed@mcs.edu.pk

***Abstract*—Traffic congestion and occlusions are major problems nowadays in metropolitan cities which leads to an ever growing traffic accidents. Therefore, the need of traffic flux management in order to avoid these congestions, unnecessary time wastage and tragic accidents is very important. Traffic regulation by optimizing timing of traffic control signals is one of the solutions for this purpose. This paper presents a low cost camera based algorithm in order to control traffic flow on a road. The algorithm is based on mainly three steps: vehicle detection, counting and tracking. Background subtraction is used to isolate vehicles from their background, Kalman filter is used to track the vehicles and Hungarian algorithm is exploited for association of labels to the tracked vehicles. This algorithm is implemented on both daytime and night time videos acquiered from CCTV camera and IR camera. Experimental results show the efficacy of the algorithm.**

Keywords-Computer vision, vehicle detection, counting, tracking

I. INTRODUCTION

Nowadays, people prefer to use their own vehicles for commutation rather than public transport and this trend causes a huge traffic flux on roads. It leads to problems like traffic congestion, air and noise pollution, peevishness in behavior, etc. Therefore, traffic flow management is important to alleviate such issues. With ubiquitously available cameras, vision based systems are more suitable and lower cost solutions than loop detectors (which require massive hardware to automate the traffic flow) and they also minimize the number of traffic wardens [1]. Moreover, the cameras may also be used for surveillance purposes.

Detection of vehicles in a sequence of frames is an active field of research in computer vision which helps in overcoming an ever growing complications in traffic surveillance and security. Fig. 1 shows the block diagram of the system based on computational analysis. Computer vision deals highly with the automatic extraction, understanding and analysis of the useful information from a single image or from a sequence of images. It implicates the theoretical and algorithmic based steps to achieve visual understanding automatically. In this paper, we propose a computer vision based vehicle detection, counting and tracking method that uses a Gaussian mixture model for background subtraction which yields a foreground mask (binary mask). The foreground mask thus obtained is then processed using the morphological operators (e.g., dilation, erosion, opening and closing) to eliminate noise. The BLOB analysis technique then helps in clearly discerning the vehicles from the background. It basically detects the cluster of connected pixels which may correspond to moving objects. Afterwards, a binary classifier helps in discerning a vehicle from pedestrians. This classifier makes use of the fact that the width to height ratio of vehicles is always greater than 1 and that for pedestrians is always less than 1. Kalman filter then helps predict the locations of vehicles during the next update interval. Hungarian algorithm is then exploited for associating labels to the tracked vehicles. In order to increase the computational speed, all of this processing is done within a region of interest (ROI) which is set in our video frames such that only those vehicles are detected which enters the ROI. Thus, the vehicle counter is incremented only when a vehicle enters the ROI.

According to Fig. 1, extraction of vehicles in frames of a video given as an input in the first step is achieved using Gaussian Mixture Model. After the extraction of vehicles, noise is removed from the processed binary image obtained as a result of second step. This noise filtering is taken care of using morphological operators. After the removal of noise, BLOB analysis is used to highlight the detected vehicles and counter is employed to achieve the counting of vehicles depending upon the highlighting of vehicles in Step 4. Counting (labelling) in Fig. 1, basically refers to cost estimation which is achieved in this frame work using a Hungarian model. This cost estimation model helps in assigning tracks to the corresponding associations.

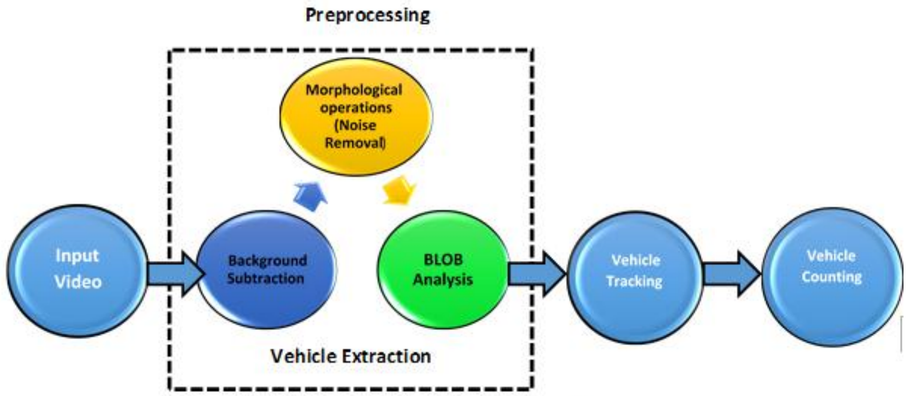

Figure 1. Block diagram of the proposed system

Rest of the paper is organized as follows: Section II discusses related work and Section III describes the explanation of proposed solution, Section IV explains the implementation and results, and finally Section V elaborates the conclusions and future work.

II. RELATED WORK

Jang Hyeok et al. in [2] employs Gaussian Mixture Background Model along with Pyramidal Lucas Kanade method to first extract the objects and then measure the displacement of an object in two consecutive frames of a video. The Lucas Kanade method is a differential method used for optical flow estimation. In this method, since one of the two frames is taken as a reference, the problems like aperture focusing and therefore correspondence issues arise which may yield results that are quite uncertain. In [3], for background subtraction and image segmentation, first frame in a video file is assumed to be a reference background. For the elimination of background, video input is first converted to frames and then the difference between the current frame and background is calculated. This difference is then used to eradicate the pixels having same values. This algorithm tracks every object that appears to be displaced in the next frame as a vehicle even if it is not a vehicle thus yielding uncertainty. In [4], frame differencing method for vehicle detection is discussed in which sub-features of a vehicle account for the detection of vehicle. Feature extraction method is followed upon using Haar wavelets technique. A wavelet analysis is almost similar to Fourier analysis in that the target function is represented in terms of orthonormal bases thus providing a non-redundant representation of an input image. In wavelet pyramidal decomposition process discussed in [5], each frame of an input video, after passing through series of filters (high pass and low pass) and down sampling, is compressed. This provides a richer model along with spatial resolution and is even more suitable in complex patterns capturing [5]. The complexity of this algorithm increases when attributes belonging to same class having same absolute values are detected as two different features. In [6], different algorithms for tracking are discussed which are classified into three main categories namely tracking based on region, tracking based on active contour and tracking based on features and combined methods. Of these three, in tracking based on region method labels are assigned to vehicles detected in order to track them through cross correlation measurements over time. However, the limitation of this approach includes the fact that this algorithm is applicable for large size objects. Furthermore, this algorithm is applicable only for fewer vehicles on the road and thus cannot in general handle occlusion reliably. In tracking, based on active contour method, the main idea is presentation of environmental contour of a vehicle and its dynamic updating [6]. The problem of this algorithm is that occlusion handling is almost inevitable because tracking precision at contour position is limited. In tracking, based on feature technique, features of intended vehicles are used as a key to tracking that vehicle. For this, in each frame, the vehicle under observation is subjected to feature extraction. Using the extracted features, vehicle tracking takes place by analyzing these features in consecutive frames of a video sequence. This method is used widely in many systems [7]. For monitoring the status of a traffic, vehicle counting is usually taken into account in order to estimate traffic flow. In [8], normalized color and edge map method was discussed for vehicle counting. Normalized color and edge map technique basically uses a color transform model to find vehicle color to efficiently locate possible candidates of a vehicle [9]. In [10], cascade Haar with pyramidal Kanade Lucas Tomasi tracker is used for vehicle counting. An object is said to be matched with an object if a threshold percentage of its points are contained in the detection window [10]. Moreover, in order to reduce false counts, an object, which is not matched with detection in maximum number of frames, is ignored in the final count [10]. When a right threshold percentage of points associated with an object go out of the window, the object is counted [10].

The system proposed in this paper uses Gaussian mixture model which has an advantage of detecting more minor details in foreground extraction because this method basically computes the PDF corresponding to every pixel in a frame which means that it is more flexible in terms of cluster covariance. Although there are many other filters better than Kalman filter, the Kalman filter is employed for tracking in proposed system since it has an advantage over all other trackers that it can correct its assumptions based on prior and posterior knowledge. This tracker also takes into account the quantities that are partially or completely neglected in other techniques and its recursive nature can help in real time execution of an algorithm without storing observations or past estimates.

III. THE PROPOSED METHOD

A. Background Subtraction Using Gaussian Mixture Model

Background subtraction is the simplest method for motion detection in video frames. We use Gaussian Mixture Model for achieving the objective of background subtraction in our system. Gaussian mixture model is like a probability distribution. This method at initial stage computes the variance, covariance and mean of every pixel in a frame. Upon the arrival of new frame, the variance, covariance and mean are again calculated and cumulative average is obtained. If the difference between the values (of above mentioned quantities) of the current frame and the cumulative average is larger than the product of actual value and standard deviation, then it is classified as a foreground [11]. Each pixel is modelled as a mixture of k normal distributions. Detail of GMM can be studied in [12].

Once approximation is done, weights are normalized. Time constant is achieved by taking the reciprocal of the learning rate which helps in determining the distribution parameter variation. GMM helps in dealing effectively with background situations because the learning rate determines the speed of adaptation to the illumination changes in the background [12]. Fig. 3, shows the results of background subtraction achieved through GMM when applied to input video frame in Fig. 2.

B. Morphological Operations

As it is evident in Fig. 3, there exists a little noise (stray foreground pixels) in the binary image obtained through GMM. To eradicate this noise, we make use of Morphological operations.

Morphology is basically described as the theory and techniques employed for the study of object's structure and its form. It helps calculate the geometrical structures of an object. To a morphological operator, a binary image along with a structuring element is given as an input. Commonly used morphological operators are dilation, erosion, closing and opening. Erosion and dilation are two basic operators. A structuring element helps define an arbitrary neighborhood structure. Fig. 4 shows the final results of morphology applied in this algorithm. Flat morphological structuring element is employed which forms an essential part of erosion and dilation. This structuring element consists of a binary valued neighborhood in which only the pixels having true value are included in morphological computation while those having false values are exempted from morphological computations. Center pixel, the origin, acts as a pixel identifier in an image being processed.

Figure 2. Input Video Frame

Figure 3. Foreground Detection (result of GMM)

C. Blob (Binarly Large OBject) Analysis

One of the fundamental technique of computer vision is the Blob Analysis. In this technique, the object under observation is made clearly discernible from background. This technique provides good performance and better flexibility. Blob is basically a connected region and region is any subset of image pixel [13]. The basic concept of this technique requires that firstly the region corresponding to the object of interest must be acquired using image thresholding techniques (background subtraction in our case). Secondly by applying morphological operators, noise is removed so that the region is enhanced. Finally, the refined region is subjected to mathematical measurements and computations to obtain final results (see Fig. 5) These measurements and computations are achieved through the use of a cascade classifier that makes use of the fact that the ratio of width to height for vehicles is always greater than 1 and that for human beings this ratio is always less than 1. These measurements and computations help in discerning moving objects as vehicles rather than a human being. BLOB detection is applied only for the vehicles entering in the region of interest that is the detection field defined (see Fig. 5) for the results of blob detections.

D. Hungarian Algorithm

Hungarian algorithm is basically a cost estimation model in which an assignment is based on the basis of cost. In most cases depending upon a requirement, it can be a maximum cost estimation or minimum cost estimation. Hungarian algorithm can be best understood by matrix interpretation.

Matrix Interpretation of Hungarian Algorithm: Suppose that we want to assign four tracks on the basis of minimum cost assignment to four detections. Writing in matrix form:

m1	m2	m3	m4
n1	n2	n3	n4
o1	o2	o3	o4
p1	p2	p3	p4

Here, we have four tracks m, n, o and p and total four detections. The task is to find the least cost of assigning the tracks to the detections. Since the rows and columns of a matrix are equal, we can infer that at a time only one track can be assigned to a single detection.

Figure 4. Results of applying morphological operations (Clean Foreground)

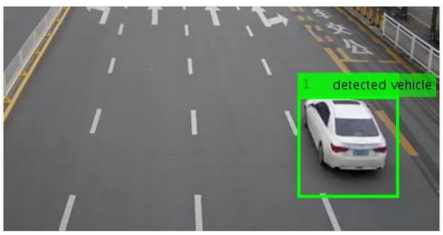

Figure 5. Blob Detection (BLOB analysis)

In the first step, we will subtract the least element of the entire row from the elements of that same row. This results in at least one zero in an entire row. The process is repeated for all the rows of a matrix.

0	m2	m3	m4
n1	n2	**0**	n4
o1	o2	o3	**0**
p1	**0**	p3	p4

So, in this way cost assignment is done in Hungarian model. For higher order matrix, if at some point the row and column operations do not yield the assignment plan, we keep on proceeding further that is applying row and column operations till we are able to distinguish among tracks in terms of least cost [14]. Thus, using this algorithm for assigning tracks to the vehicles being detected helps to distinguish between detected and undetected vehicles in a video frame. The main problem of data association is to determine the correspondence between detections and multiple tracked vehicles.

E. Kalman Filter

A Kalman filter is used to predict the systems state during the upcoming intervals when it cannot be measured directly.

We assume our system to be linear, so the system equation can be defined as:

$$x_k = Ax_{k-1} + w_{k-1} \tag{1}$$

Where w_{k-1} is the white noise (also called as the process noise) that occurs due to inaccuracy in measurement of updated position of the vehicle and from computational tasks performed by the computer, A is an nxn matrix defined as the state transition matrix [15]. Note that k-1 represents the previous state.

IV. IMPLEMENTATION AND RESULTS

The above mentioned algorithm for vehicle detection is implemented using a MATLAB 2017a software .The results of the aforementioned algorithm are compared with the ground truth values of vehicles in frames.

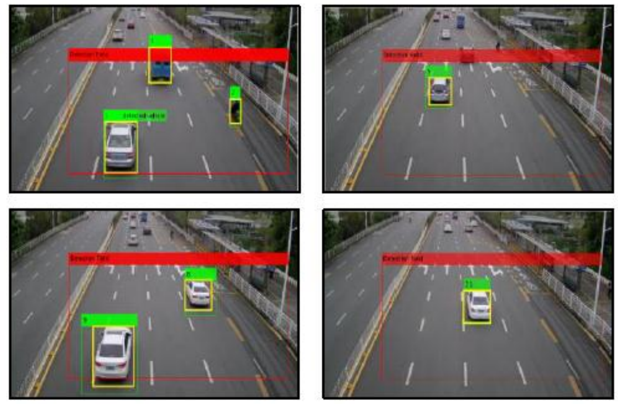

Figure 6. Comparison of results with the ground truth (green boxes show actual detection through this algorithm whereas yellow boxes represent ground truth positions of cars in a frame)

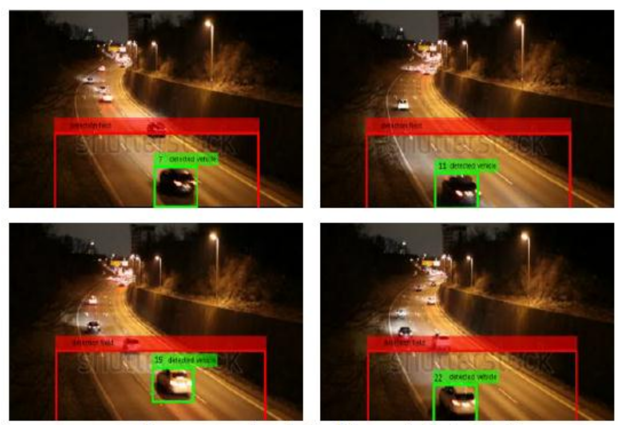

Figure 7. Results of algorithm at night time [21].

The implementation of the above discussed algorithm yields results with efficient tracking and detection of vehicles in a video. The cost of assignment of detections using Hungarian Algorithm yields accurate counting of vehicles in a video. Only the vehicles entering the region of interest (detection field) are detected, tracked and counted by this algorithm.

The detection obtained through this framework when compared with the ground truth value gives minute error which shows the accuracy of this frame work. To account for the accuracy of object detector we will use an evaluation metric Intersection Over Union (IoU) method which requires two things namely ground truth bounding boxes and predicted bounding boxes. Intersection over union is merely a ratio in which $A \cap B$ corresponds to the area of overlap between the predicted (that is actually computed bounding box through this frame work) and the ground truth bounding box whereas $A \cup B$ corresponds to the area encompassed by both the predicted and the ground truth bounding boxes [16]. Fig. 6 shows the comparision of results with the ground truth values.

The above mentioned algorithm when run at on other times of a day, for example night time (CCTV camera), yields results as shown in Fig. 7

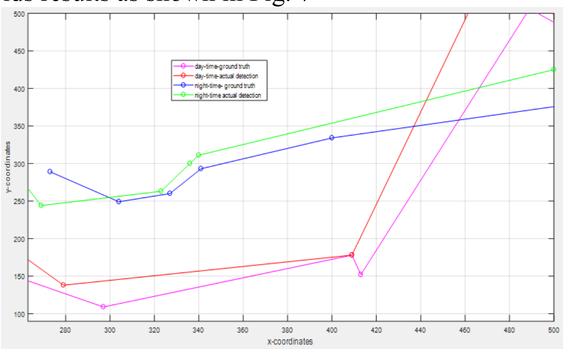

Figure 8. Difference between the plots of coordinates of ACTUAL DETECTION (through this frame work) and that of GROUND TRUTH coordinates

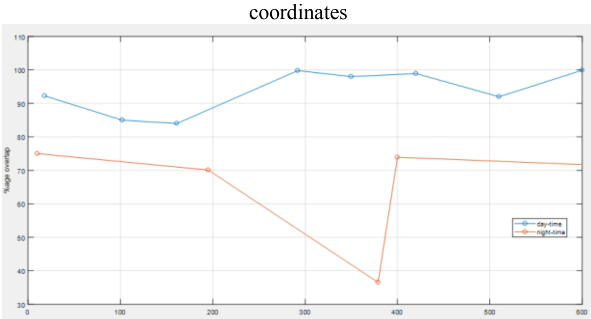

Figure 9. Results in the form of percentage overlap (day and night time)

Figure 10. Algorithm implemented on IR camera

Figure 11. Original frame and processed frame (from left to right) in low traffic density.

Figure 12. Original frame and processed frame (from left to right) in medium traffic density.

Graphs (see Fig. 8 and Fig. 9) show the results of percentage overlap and variations in the coordinates of ground truth and actual detection obtained through above mentioned algorithm.

TABLE I and TABLE II show the accuracy of system through percentage overlap.

TABLE I. RESULTS OF DETECTIONS DURING NIGHT-TIME
(as percentage overlap between the ground truth and the actual predictions)

Frame No.	Detections	%age overlap
frame#195	Car#7	70.12%
frame#379	Car#11	56.55%
frame#400	Car#19	73.93%
frame#683	Car#22	70.82%

TABLE II. RESULTS OF DETECTIONS DURING DAY-TIME
(as percentage overlap between the ground truth and the actual predictions)

Frame No.	Detections	%age overlap
frame#18	Detected Car#1	89.02%
	Detected Car#2	89.2%
	Detected Car#3	80%
frame#102	Detected Car#5	85%
frame#161	Detected Car#8	87%
	Detectetd Car#9	80%
frame#287	Detected Car#21	89.2%

The above table predicts that the system stabilizes giving more accurate results as video progresses. Since this algorithm also detects shadows of the vehicle that is why for vehicle having larger shadow will have larger bounding box around it and as a result its Intersection over Union will be lower. This accounts for the reason as to why some vehicles have higher IOU's compared to the others. Also, the headlamps of light at night just like shadows also forms part of vehicle because of the use of GMM that is why IOU for detected vehicle number 11 at night is very low. Whenever a vehicle leaves the detection field, the corresponding bounding box assigned to it also vanishes and whenever a vehicle enters the detection field, it is assigned a bounding box upon its detection with the total number of vehicle counter incremented.

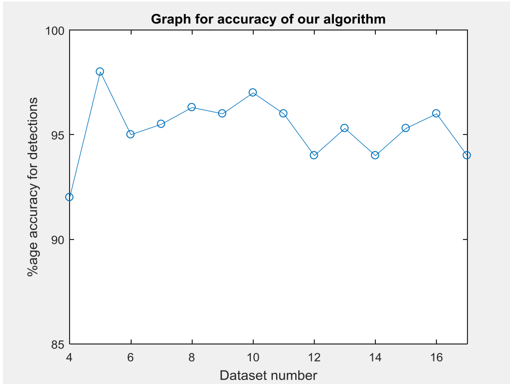

Figure 13. Graph of accuracy of our algorithm based detections

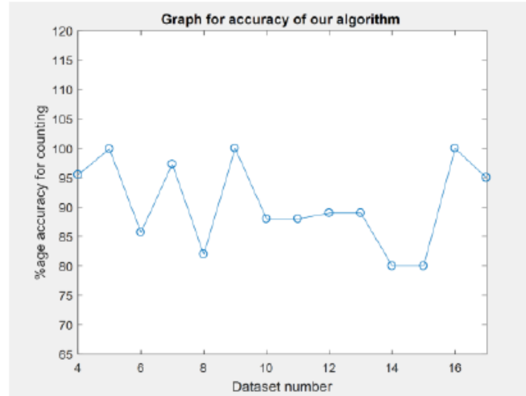

Figure 14. Graph of accuracy of our algorithm based countings.

Above mentioned algorithm can also be implemented on videos acquired from IR-camera. The implementation of algorithm on IR-camera yield results that are close to the ground truth position of vehicles such that the accuracy of approximately 85% (average value) is achieved when intersection over union is applied on ground truth and actual detections. Fig. 10 shows the results.

TABLE III. RESULTS OF IMPLEMENTATION OF OUR ALGORITHM ON BENCHMARK VIDEOS [19] for (low traffic density)

Results of	**Dataset#1**	**Dataset#2**	**Dataset#3**	**Dataset#4**	**Dataset#5**	**Dataset#6**	**Dataset#7**	**Dataset#8**	**Dataset#9**	**Dataset#10**	**Dataset#11**	**Dataset#12**	**Dataset#13**	**Dataset#14**	**Dataset#15**	**Dataset#16**	**Dataset#17**	**Dataset#18**	**Dataset#19**	**Dataset#20**
Detection (%age)	96.5	96.3	96	92	98	95	95.5	96.3	96	97	96	94	95.3	94	95.3	96	94	94	91	92
Counting (%age)	95.5	100	90	95	99.9	85.7	97.3	82	100	88	88	89	89	80	80	100	95	87	80	70

TABLE IV. COMPARISON OF RESULTS WITH THOSE OBTAINED IN [18]

Reference No.	**%age accuracy for detections** (Average value)	**%age accuracy for counting** (Average value)
Ours	94.77≈95%	89.77%
[18]	94.22%	-

In [17], detection of vehicle using a triangle thresholding method was proposed. According to [17], this thresholding technique is determined through histogram by normalizing its brightness values and the dynamic range of its intensity values. The differenciation line drawn between maximum and minimum of the histogram brightness values is used to determine the threshold value. Brightness value having max distance is set as a threshold [17]. Similarly in [8] as discussed before, normalized color and edge map method was discussed for vehicle counting and in [18], pixel domain methods were employed to achieve the objective of vehicle detection. To account for the accuracy of our proposed algorithm, benchmark videos used in [18] were used which were obtained from a UCSD database [19]. The accuracy of our algorithm compared with that of [18] is depicted in figures (see Fig. 11 and Fig. 12)

TABLE III. shows the results of applying our algorithm, on bemchmark videos in [19]. Detections are represented in the form of Intersection Over Union percentage and counting accuracy of our algorithm on each dataset is also represented in the form of percentage. Through our algorithm we attain detection accuracy of 94.77% which is very close to 95% and is better than the results in [18]. Counting accuracy of our algorithm comes out to be 89.57% which is approximately equal to 90% as TABLE IV depicts. The lower percentage of counting in some datasets results due to error in first frame of videos in benchmark. Fig. 13 and Fig. 14 show the graphs of our results discussed in TABLE III and TABLE IV.

Dataset**:**

Due to lack of augmented dataset online for vehicle detection in videos taken with stationary camera, groundtruth for videos was self constructed using object annotation tools. Day time video (using CCTV camera) used in this algorithm was acquired from a youtube source [20] whereas night time video and its dataset was self constructed using online annotation tools LabelMe to obtain the ground truth positions of car in a video frame. The daytime video has a frame rate of 25 fps. Night time video was acquired online from [21] and its groundtruth was also self constructed using object annotation tools. The night time video (CCTV camera) has a frame rate of 29.97fps. IR-camera video was acquired from database in [22]. This video has a frame rate of 30fps. Benchmark videos in [18] which were used for comparision of our algorithm were obtained from UCSD database [19]. Each video in dataset has a frame rate of 10fps.

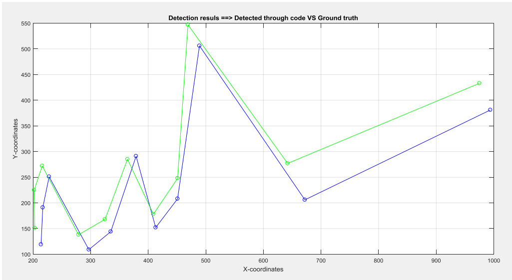

Figure 15. Difference between the plots of coordinates of ACTUAL DETECTION (through this frame work) and that of GROUND TRUTH coordinates

V. CONCLUSION AND FUTURE WORK

In this paper, a framework for vehicle detection, tracking and counting was proposed. Its key functions are detecting vehicles which form the foreground part of a frame using Gaussian Mixture model [11], [12] which gives a binary mask. This binary mask is then subjected to morphological operators [23].

Morphology is basically applied in order to overcome noise in binary foreground detected image. After the removal of noise, connected regions are detected using Blob Analysis [13] and these regions are then assigned detections using Hungarian Algorithm [14] on the basis of cost estimation. Next, for tracking the trajectory of the vehicle within the next update interval, KALMAN filter is employed. Comparing the results of our algorithm with the one discussed in [8] and [15] we conclude that our system is more efficient and accurate since it yields results that are much closer to the ground truth values of vehicles. In future, the proposed algorithm can be implemented using OpenCV software to work online.

The limitations of this paper include: vehicle shadows which become a part of foreground, nonstationary camera, greater velocity of vehicles and intense sunlight which causes reflection from car windows thus making extra objects a part of foreground. These issues can be resolved using some techniques like thresholding the frames of a video to remove noise or by using histogram of gradient (HOG) for vehicle detection. For fast speed vehicles, model of Kalman filter can be improved so that it can detect fast moving vehicles.

REFERENCES

[1] Gupte, Surendra, et al. “Detection and Classification of vehicles,” ITS Institute of Minnesota, 2002.

[2] Jang, Hyeok, In-Su Won, and Dong-Seok Jeong. “Automatic Vehicle Detection and Counting Algorithm,” IJCSNS International Journal of Computer Science and Network Security, vol.14, (2014).

[3] P.M. Daigavane, P.R. Bajaj. “Real Time Vehicle detection and Counting Method for Unsupervised Traffic Video on Highways,” IJCSNS International Journal of Computer Science and Network Security, vol.10, (2010).

[4] Hadi Raad Ahmed, Sulong Ghazali, George Loay Edwar. “Vehicle Detection and Tracking Techniques: A concise review,” International Journal (SIPIJ), Vol. 5, No.1, (2014).

[5] Wen Xuezhi, et al. “Improved wavelet feature extraction methods based on HSV space for vehicle detection,” IAPR conference on machine vision application, Tokyo (2007).

[6] Farahani, Gholamreza, “Dynamic and Robust Method for Detection and Locating Vehicles in the Video Images Sequences with the Use of Image Processing Algorithm.”, EURASIP journal on image and video processing, 2017.

[7] B Han, L Davis, “Object Tracking by Adaptive Feature Extraction.” International Conference on Image Processing (Institute of Electrical and Electronics Engineers (IEEE), Singapore, 2004.

[8] Yang Zi, Pun-Cheng S.C Lilian, “Vehicle Detection in Intelligent Transportation Systems and Its aplications under varying environments,” The Department of Land Surveying and Geo-Informatics, The Hong Kong Polytechnic University, Hong Kong (2017)

[9] Luo-Wei Tsai, Jun-Wei Hsieh, Kuo-Chin Fan, “Vehicle Detection using Normalized Color and Edge Map.”, IEEE transactions on Image Processing, 2007.

[10] Patel, Chirag I., Patel Ripal. “Counting Cars in Traffic using Cascade Haar with KLP.”, International Journal of Computer and Electrical Engineering, 2013

[11] Hardas, Adesh, et al. “Moving Object Detection using Background Subtraction Shadow Removal and Post Processing,” International Journal of computer applications, 0975 – 8887(2015).

[12] Alandkar, Lajari, Gengaje, Sachin R. “Delaing Background issues in Object Detection using GMM: A Survey,” International Journal of Computer Applications, vol.150-No.5, 0975-8887(2016).

[13] “Machine vision guide”, Retrieved from: https://docs.adaptive-vision.com/current/studio/machine_vision_guide/BlobAnalysis.html, last accessed 2018/01/05.

[14] “The free encyclopedia”, Retrieved from: https://en.wikipedia.org/wiki/Hungarian_algorithm, last accessed 2018/01/05.

[15] Rezaei, Mahdi, “Computer vision for road safety: A system for simultaneous monitoring of driver behavior and road hazards,” University of Auckland, New Zealand (2014).

[16] PYimagesearch – “Machine learning,” https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/, last accessed 2018/02/12.

[17] Mohamed A. El-Khoreby, Syed Abd Rehman Abu Bakr, “Vehicle Detection and Counting in Complex Weather Conditions.”, IEEE International Conference on Signal and Image Processing, Malaysia, 2017.

[18] Xu Liu, ZileiWang, et al., “Highway Vehicle Counting in Compressed Domain.”, CVPR, 3016-3024, 2015.

[19] Statistical Visual Computing Lab-“UCScanDiego”, retrieved from: http://www.svcl.ucsd.edu/projects/traffic/, last accessed 2018/04/23.

[20] Day time video using CCTV camera, Retrieved from: https://www.youtube.com/watch?v=eO19UTm93GQ.

[21] Shutterstock, Inc [US] – “Cars run low speed in late nighttraffic on motorway”, retrieved from: https://www.shutterstock.com/video/clip-3756275-stock-footage-cars-run-low-speed-in-late-night-traffic-on-motorway-dusseldorf-germany-area.html?src=rel/3756290:3

[22] OTCBVS Benchmark Dataset Collection, Dataset#11, Retrieved from: http://vcipl-okstate.org/pbvs/bench/, 10th May, 2018.

[23] “Image processing basics”, Retrieved from: http://www.coe.utah.edu/~cs4640/slides/Lecture11.pdf, last accessed 2018/01/05.