Video Based Vehicle Detection and Speed Tracking System

Akash Bhadange
Department of Electronics and Telecommunication
Vishwakarma Institute of Technology, Pune
Pune, India
akash.bhadange20@vit.edu

Resham Bhattad
Department of Electronics and Telecommunication
Vishwakarma Institute of Technology, Pune
Pune, India
resham.bhattad20@vit.edu

Pratik Daga
Department of Electronics and Telecommunication
Vishwakarma Institute of Technology, Pune
Pune, India
pratik.daga20@vit.edu

Prof.(Dr.) Swati Shilaskar
Department of Electronics and Telecommunication
Vishwakarma Institute of Technology, Pune
Pune, India
swati.shilaskar@vit.edu

Abstract-The expansion of a programmed entity for measuring vehicle speed is a key part of the solution to the problem with the traffic surveillance system. It is advantageous for safe traveling, preventing traffic accidents, etc. Numerous technologies or approaches have been created for speed assessment, however in this study, we follow crucial procedures such as background modeling, foreground recognition, and vehicle tracking to mitigate the shortcomings of the earlier methods. In this study, a method for tracking and measuring vehicle speed is presented. Vehicle tracking was accomplished via blob detection across all frames, while the speed is calculated by making use of a technique related to background modeling. This method is adaptable and used for additional purposes. Using the number of frames, frame rate, and distance traveled by the vehicle, the speed was estimated mathematically. This experiment was conducted using several vehicle types traveling at various speeds.

***Keywords*— background modeling, vehicle tracking, frame rate, foreground recognition**

I. Introduction

One of the dangerous issues facing contemporary society and the economy is traffic management. The traffic surveillance system offers information on a variety of traffic factors, including vehicle count and speed. As speed is thought to be a major contributing factor in auto accidents, every effort is being made to identify and regulate vehicle speed. There are numerous tools available for speed detection. In recent years, a number of speed estimating techniques have been proposed. Every strategy makes an effort to increase precision. When a vehicle is tracked using video, more information is revealed that is difficult to learn using loop detectors. When used in conjunction with real-time surveillance, such a system is useful. Frames subtraction and masking techniques are utilized for vehicle segmentation [1]. Modern object detectors, transfer learning, and convolutional neural network (CNN) classifiers with projective geometry were employed to identify vehicles, pedestrians, and bikers in monocular films.information to classify vehicles [2]. For the categorization and identification of vehicles, one of the most prominent methods is the Support Vector Machine (SVM) classifier [3]. Background Subtraction The video-based vehicle recognition, categorization, counting, and speed measurement adaptive system for traffic data gathering uses the MoG2 algorithm, OpenCV, and the Java SE development kit [4]. MATLAB platform is used for programming [5].

In this study, we identify moving automobiles in video streams using OpenCV and Deep Learning, follow them, and use speed estimates to determine their speed.Two sequential frames are fed into the algorithm to determine the speed of moving vehicles on a road. The frame rate, which will be calculated from video, is multiplied by the total number of frames. This information is used to determine the overall trip time by vehicle. The distance needed to measure speed is constant. The speed of the corresponding vehicle is calculated using the detected vehicle's distance and duration of travel.

In the proposed system, we track the car using blob tracking and absolute differencing. When a vehicle is tracked, the centroid of the blob is tracked to determine the distance travelled by the vehicle. To lessen the noise, further thresholding operations are performed.

II. Literature Survey

For detecting vehicle speed in the past, inductive loops, RADAR guns, LASER guns, or manual counts were employed. The authors of this research suggested a system for detecting vehicle speed utilizing image and video processing methods. For vehicle segmentation, they have used frame subtraction and masking techniques. Speed is determined using the time between frames and the segmented item crossed in those frames. Frame masking is additionally used to distinguish between one or more cars. Results for speed detection have an inaccuracy of +/- 2 km/h on average [6]. The authors offer a vision-based pipeline with modules for counting vehicles in specific lanes, classifying vehicles in specific lanes, and counting pedestrians and cyclists in specific lanes from monocular films. The proposed pipeline may produce high-quality metadata for future traffic analysis while analysing recorded movies at 60 frames per second [7].

Linear, Quadratic, Cubic, and Gaussian SVM classifier models were trained utilising HOG traits from the collected pictures.The trial findings showed that the Cubic SVM classifier model, with an accuracy of 94.29%, provided the best output for detection and classification in comparison to the other SVM models [8].To estimate the speed, they have suggested a combination of Gaussian techniques [9]. They have used the blob detection technique used to display the vehicle movement which provides coordinates in the form of centroid [10]. To locate foreground items in a video clip, they employed the Background subtraction approach [11]. They have used the MATLAB platform for programming [12].In this study, image and video processing techniques are used to estimate the velocity of moving objects. Furthermore, the camera calibrations weren't used [13]. For extracting moving objects, they have employed three frame difference approaches and a backdrop difference method. [14]. For multiple object detection they have used a faster R-CNN algorithm, for multiple object tracking they have used histogram comparison and speed conversion they have used warping [15].The authors put out a deep learning-based study for calculating vehicle speed from drone footage. To identify the different types of passenger cars, the deep learning library You Only Look Once (YOLO) was utilised [16].system can count vehicles, categorise them, and calculate their speeds,They paired a SORT tracker with a two-stage Faster R-CNN detector to complete the assigned aim [17]. After the background image in the movie has been fixed, the background information is used to generate the backdrop model [18].

In the area of automotive object identification, deep convolutional networks (CNNs) have had incredible success.CNNs can perform a wide range of related tasks, including classification and bounding box regression, and are skilled at learning image features [19]. Two broad categories may be made for the detecting techniques. The two-stage method employs a convolutional neural network to categorise the object after creating a candidate box of it using a variety of techniques.Instead of developing a candidate box, the one-stage technique instantly converts the positioning problem of the object bounding box into a regression problem for processing. Region-CNN is used in the two-stage approach (R-CNN) [20].

YOLO is a single stage object detector that can run at 45 frames per second on a Titan X GPU. It was first proposed in 2016. YOLO contains 24 convolutional layers and two fully linked layers. Instead of 24, a quicker version of YOLO has 9 convolutional layers. A 7730 tensor is the network's predicted ultimate output. Depending on the video resolution, a later version of YOLO called YOLOv2 (also known as YOLO9000) [22] can process up to 67 frames per second. The third iteration of YOLO, version 3 [23], is yet another quick object detector. It contains some minor upgrades to YOLOv2. Liu et al Single .'s Shot MultiBox Detector (SSD) technique. [24] is another one stage approach where the object classes and the group of default anchor boxes are projected together with their bounding boxes.

Convolutional neural network (CNN) characteristics are generated for each area proposal in the R-CNN two stage detector that Girshick et al. [25] propose. The SVM learned for each item class is used to score the feature vectors. After that, a non-maximum suppression phase removes overlapping regions from the final scored regions. By choosing the region-of-interest (ROIs) from feature maps derived using convolution, Fast R-CNN [26], a later method from the same group of authors, expands R-CNN. The report demonstrates how their approach increases accuracy and speed. Faster R-CNN [27], which can operate at a speed of 5 frames per second, considerably enhances Fast R-CNN with an RPN. When compared to YOLOv2, this frame rate is still rather low.

A popular method for tracking several objects is tracking via detection (MOT). Tracking by detection techniques define the MOT job as a process of associating the observed bounding boxes with succeeding video frames. As a result, one way to conceptualise the problem of the proper assignment events in MOT is as a data association problem between the targets detected in the previous frames and the detections in the current frame. [28], [29]. Contrary to single object tracking, MOT approaches must be capable of handling the identity switch issue when the monitored items have a similar look.

Perera et al. [30] evaluate each potential hypothesis throughout the trajectory splitting and merging process. They use the Stauffer-Grimson backdrop modeling approach and their tracking algorithm

III. Methodology

The major phase of the recommended procedure is depicted as shown in Fig, 1. Background modeling, which provides the removal of subsequent frames, is used to extract the featured portion of the video stream. The location of the vehicle is determined through Foreground detection. And then the speed of the vehicle is calculated.

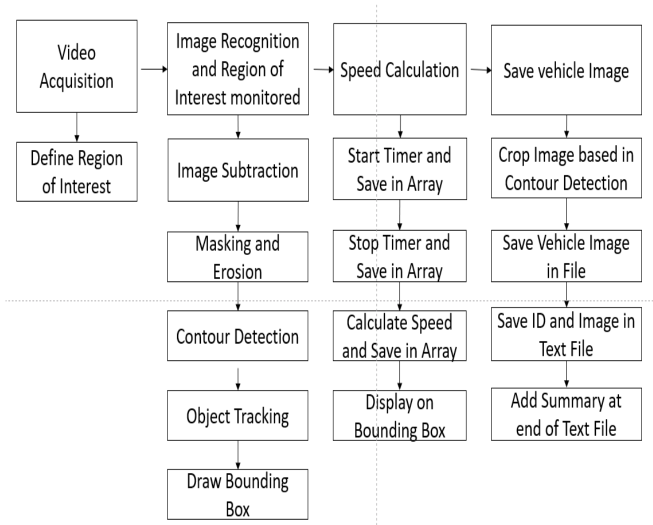


Fig. 1 Block diagram of Vehicle detection and speed tracking technology

A. Video Acquisition

Pre-processing establishes the many video properties, such as Bits per pixel, frame rate, number of frames, and video formats. Frames are also reduced in size and converted to double data format.

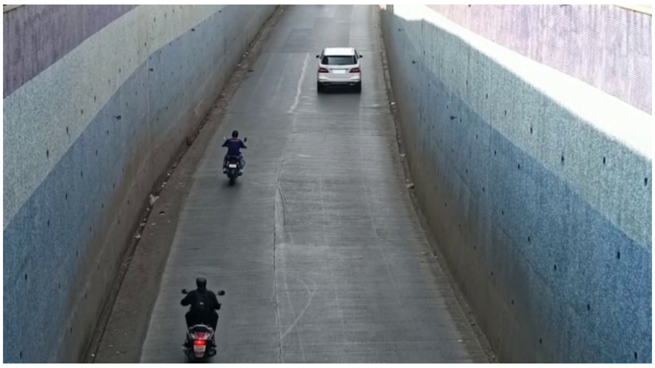


Fig. 2 Sample Video Screenshot

B. Background Modeling and foreground detection

Background subtraction is a component of the background modelling. It is an important element of this technique. It is customary to use a background image without a moving car, but in this case, we are using the first frame. We create an image with just stationary objects by subtracting the current frame from the background frame in this manner.

$$M(k, p) = 1(c, p) - O(p) \qquad (1)$$

In equation (1), The background picture from the first frame is represented by O(p), the absolute difference image is represented by M(C, p), and the current frame is represented by 1(c, p) in the kth frame. The original image is then multiplied by each frame before being converted from RGB to grey level.

$$N(p) = M(c, p) \times V(p) \qquad (2)$$

Equation (2) shows,V (p) which is the colour RGB picture of O (p), is multiplied by N (p), an image that has had its original pixels subtracted (p). It helps with vehicle detection accuracy. The gradual changes are caused by the typical difference between frames. As a result, the original frame is multiplied by the absolute difference between the background frame and the current frame in order to swiftly detect cars. Only one similar portion of the image is identified during foreground detection.

C. Masking and Region of Interest

of InteA smaller percentage of the original footage is taken up by Region of Interest (ROI). Image subtraction is used on this ROI to find moving vehicles. (Image subtraction aids in determining the distinction between two frames.) Masking is used to make the background black and the moving cars appear white.

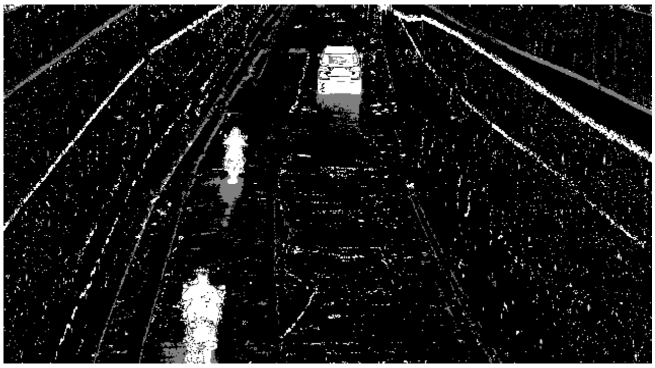


Fig. 3 Masked Image

D. Contour Detection and Object Tracking

The contours are identified based on the area threshold of the number of pixels. The threshold prevents the contours of smaller moving objects that are not automobiles from being detected. Based on the separation between two contours between frames, the object is tracked. Each contour is allocated an ID.

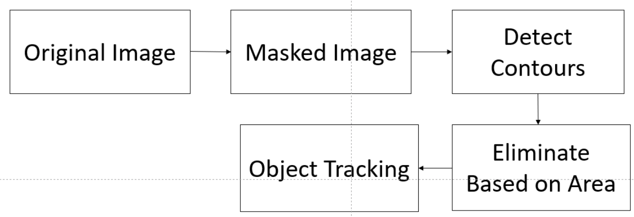


Fig. 4 Object Tracking

E. Vehicle tracking

By separating the foreground image from the background and taking each video frame out individually, it is possible to obtain a moving object, such a car travelling along a road. Finding analogous sites to trace the journey over a number of picture frames is done using the vehicle that was identified after thresholding and morphological processes as show in Fig(5). Results from applying thresholding to the image may contain noise that is further susceptible to being mistaken for a vehicle. Therefore, it is necessary to filter the blobs according to their size; blobs larger than a certain size are regarded as vehicles and are represented by bounding boxes.

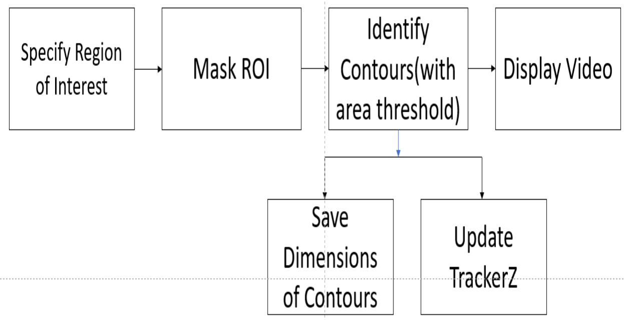


Fig. 5 Block Diagram (Vehicle Detection)

Picture equivalency is the process of locating a certain point in one frame of an image and then relocating it in the subsequent frame. Follow confined objects that can be

closely inspected inside the track window using the centroid tracking technique. The image was initially changed into a bi-level image by turning the object pixels black and the image pixels white. The locations of the pixels within the object are then averaged to determine the centroid.
Which can be calculated as,

$$X_d = (\sum X_i)/n \qquad (3)$$

$$Y_d = (\sum Y_i)/n \qquad (4)$$

In equation (3) and (4),
X_d = Vehicle Centre Co-ordinate on the X-axis
Y_d = Vehicle Centre Co-ordinate on the Y-axis
X_i & Y_i = the coordinates of a spot inside the vehicle-limited region on one of the n images

The centroid provides the vehicle's Centre pixel location, which is then utilized to track the entire object. Any feature-tracking method must have both accuracy and resilience, which are crucial components. For the tracking to be considered effective, the displacement of the observed vehicle's Center point across two succeeding frames must be smaller than the separation between its Center and the Center of another vehicle. The next step of tracking involves matching the collection of feature points in each frame of the sequence of photos using a matching technique. We must utilise the coordinates of their centres to calculate all the distances dk between the vehicle in frame n and each subsequent vehicle in order to track it.

$$D_k = \sqrt{} (X_k - X_d)^2 + (Y_k - Y_d)^2 \qquad (5)$$

We locate the tracked vehicle in the following frame by calculating D = min (D_k) as shown in equation (5).

F. Vehicle Speed Measurement

The speed and time differential between one vehicle's location and another are calculated using a formula. As shown in fig.6, when the car crosses the first line, the timer begins, and when it crosses the second line, it stops. When the car crosses both lines, the speed is only displayed above the enclosing box. The detailed speed formula is shown in below speed function,

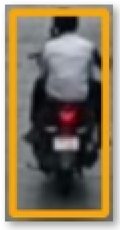
```
def getsp(self,id):
    if (self.s[id]!=0):
        s = D/self.s[id]
    else:
        s = 0
        return s
```

The easiest way to calculate the value of D is by taking a reference vehicle first, and calculate the speed backwards. A more complicated approach would be to take into consideration the distance between the lines, time taken, frame rate and lag due to computation. You can evaluate the precision of speed computation using frame rate.

IV. RESULT

By utilizing the formulas in equation(8), this project is efficiently able to track vehicles and determine their speed. Depending on the speed at which the application is executed, the speed estimate may vary by 0 to 2 km/h.
The vehicle detection model, detects the vehicle and draws the green color bounding box around the vehicle as shown in fig 6, by centroid tracking method. Here we have given the threshold speed as 80km/hr. If the detected vehicle exceeded the speed by their threshold value Then this detected vehicle marked by orange box as shown in Fig. 7.

Fig. 6 Vehicle Detection



4_speed_81

Fig. 7 Vehicle which cross speed limit

System is successfully able to track the vehicle and detect its speed as shown in Fig. 8.
Multiple vehicles and their speeds can be detected. However, if two cars are moving very closely beside one another, it can be mistaken for a single item. In order to differentiate vehicles from the background, this duty calls for as much steadiness as is practical.

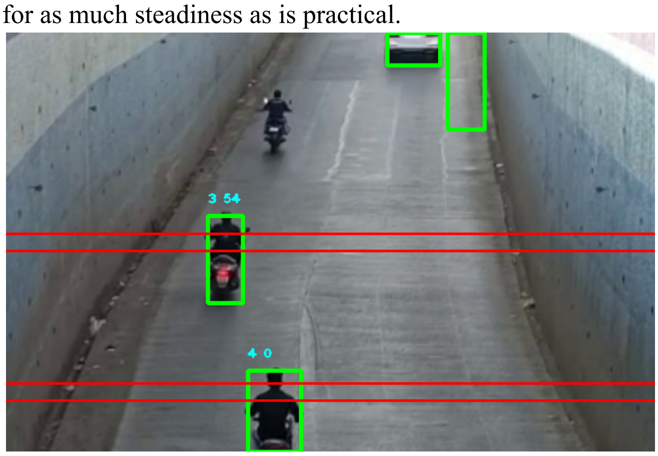


Fig. 8 Speed Estimation

A video stream of a car being driven at the right pace in accordance with the project's parameters is used to calculate the speed, and it is then compared to the speed determined using the technique used in this work. The result is displayed in Table 1

Table 1. Result Verification

Sr. No.	Real Speed(km/hr)	Measured Speed(km/hr)	Error Rate
1	43	45.5	+1.5
2	52	50	-2.0
3	35	38	+3.0
4	45	42.5	-2.5

V. Conclusion

In this study, road safety is a key concern. Furthermore, it is our duty as citizens to verify the law and ensure road safety.This project has the capacity to calculate vehicle speeds and save vehicle information.

In this study, we outline a system for evaluating vehicle speed and tracking it using videos. Speed estimation using image processing is more accurate and cost-effective than using traditional radar, and it may take use of its many benefits. In this study, moving targets are extracted using the three-frame difference approach and background difference. The centroid position of the moving object is examined along with the mapping relationship to estimate the approximate velocity of the vehicle. Although the procedure is reliable and very practicable, there is a little amount of inaccuracy.

VI. Future Scope

There have been numerous approaches put out thus far, but none of them meet the requirements for precision, effectiveness, and dependability. A system with a calibrated and high-resolution camera is something we can design for accurate results so that it will successfully help us avoid traffic accidents under low lighting, especially at night.

VIII. References

[1] K. V. K. Kumar, P. Chandrakant, S. Kumar and K. J. Kushal, "Vehicle Speed Detection Using Corner Detection," 2014 Fifth International Conference on Signal and Image Processing, 2014, pp. 253-258, doi: 10.1109/ICSIP.2014.46.

[2] C. Liu, D. Q. Huynh, Y. Sun, M. Reynolds and S. Atkinson, "A Vision-Based Pipeline for Vehicle Counting, Speed Estimation, and Classification," in IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 12, pp. 7547-7560, Dec. 2021, doi: 10.1109/TITS.2020.3004066.

[3] J. J. P. Belen, J. C. V. Caysido, A. B. Llena, E. J. O. Samonte, G. N. Vicente and E. A. Roxas, "Vision based classification and speed estimation of vehicles using forward camera," 2018 IEEE 14th International Colloquium on Signal Processing & Its Applications (CSPA), 2018, pp. 227-232, doi: 10.1109/CSPA.2018.8368717.

[4] P. M. Dhulavvagol, A. Desai and R. Ganiger, "Vehical Tracking and Speed Estimation of Moving Vehicles for Traffic Surveillance Applications," 2017 International Conference on Current Trends in Computer, Electrical, Electronics and Communication (CTCEEC), 2017, pp. 373-377, doi: 10.1109/CTCEEC.2017.8455043.

[5] R. M. Waregaonkar, S. M. Deokar, S. S. Patankar and J. V. Kulkarni, "Development of prototype for vehicle speed measurement," 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2017, pp. 945-949, doi: 10.1109/RTEICT.2017.8256737.

[6] N. Seenouvong, U. Watchareeruetai, C. Nuthong, K. Khongsomboon and N. Ohnishi, "A computer vision based vehicle detection and counting system," 2016 8th International Conference on Knowledge and Smart Technology (KST), 2016, pp. 224-227, doi: 10.1109/KST.2016 7440510.

[7] S. S. Wardha, S. M. Deokar, S. S. Patankar and J. V. Kulkarni, "Development of automated technique for vehicle speed estimation and tracking in video stream," 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2017, pp. 940-944, doi: 10.1109/RTEICT.2017.8256736.

[8] K. V. K. Kumar, P. Chandrakant, S. Kumar and K. J. Kushal, "Vehicle Speed Detection Using Corner Detection," 2014 Fifth International Conference on Signal and Image Processing, 2014, pp. 253-258, doi: 10.1109/ICSIP.2014.46.

[9] D. C. Luvizon, B. T. Nassu and R. Minetto, "A Video-Based System for Vehicle Speed Measurement in Urban Roadways," in IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 6, pp. 1393-1404, June 2017, doi: 10.1109/TITS.2016.2606369.

[10] J. -x. Wang, "Research of vehicle speed detection algorithm in video surveillance," 2016 International Conference on Audio, Language and Image Processing (ICALIP), 2016, pp. 349-352, doi: 10.1109/ICALIP.2016.7846573.

[11] T. Huang, "Traffic Speed Estimation from Surveillance Video Data: For the 2nd NVIDIA AI City Challenge Track 1," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2018, pp. 161-1614, doi: 10.1109/CVPRW.2018.00029.

[12]A. Ghosh, M. S. Sabuj, H. H. Sonet, S. Shatabda and D. M. Farid, "An Adaptive Video-based Vehicle Detection, Classification, Counting, and Speed-measurement System for Real-time Traffic Data Collection," 2019 IEEE Region 10 Symposium (TENSYMP), 2019, pp. 541-546, doi: 10.1109/ TENSYMP46218.2019.8971196

[13] Song, H., Liang, H., Li, H. et al. Vision-based vehicle detection and counting system using deep learning in highway scenes. Eur. Transp. Res. Rev. 11, 51 (2019). https://doi.org/10.1186/s12544-019-0390-4

[14] A. L. Siregar, A. Kurniawan, S. Juniastuti and E. M. Yuniarno, "Vehicle Speed Calculation from Drone Video Based on Deep Learning," 2020 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM), 2020, pp. 229-233, doi: 10.1109/CENIM51130.2020.9297844.

[15] Alexander Grents, Vitalii Varkentin, Nikolay Goryaev, Determining vehicle speed based on video using convolutional neural network, Transportation Research Procedia, Volume 50, 2020, Pages 192-200, ISSN 2352-1465, https://doi.org/10.1016/j.trpro.2020.10.024

[16] S. Javadi, M. Dahl, M. I. Pettersson and W. J. Kulesza, "Design of A Video-Based Vehicle Speed Measurement System - An Uncertainty Approach," 2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR), 2018, pp. 44-49, doi: 10.1109/ICIEV.2018.8640964.

[17] Z. Dai et al., "Video-Based Vehicle Counting Framework," in IEEE Access, vol. 7, pp. 64460-64470, 2019, doi: 10.1109/ACCESS.2019.2914254.

[18] Ghosh, M. S. Sabuj, H. H. Sonet, S. Shatabda and D. M. Farid, "An Adaptive Video-based Vehicle Detection, Classification, Counting, and Speed-measurement System for Real-time Traffic Data Collection," 2019 IEEE Region 10 Symposium (TENSYMP), 2019, pp. 541-546, doi: 10.1109/TENSYMP46218.2019.8971196.

[19] J. Mirthubashini and V. Santhi, "Video Based Vehicle Counting Using Deep Learning Algorithms," 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), 2020, pp. 142-147, doi: 10.1109/ICACCS48705.2020.9074280.

[20] Park, K., Lee, D., Park, Y. (2007). Video-based detection of street-parking violation, In International Conference on Image Processing. https://www.tib.eu/en/search/id/BLCP%3ACN066390870/Video-based-detection of street-parking-violation, vol. 1 (pp. 152–156).

[21] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), June. 2016, pp. 779–788.

[22] J. Redmon and A. Farhadi, “YOLO9000: Better, faster, stronger,” in Proc. CVPR, Jul. 2017, pp. 6517–6525.

[23] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement,” 2018, arXiv:1804.02767. [Online]. Available: http://arxiv.org/abs/ 1804.02767

[24] W. Liu et al., “SSD: Single shot MultiBox detector,” in Proc. Eur. Conf. Comput. Vis. Cham, Switzerland: Springer, 2016, pp. 21–37.

[25] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2014, pp. 580–587.

[26] R. Girshick, “Fast R-CNN,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 1440–1448.

[27] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real time object detection with region proposal networks,” in Proc. Adv. Neural Inf. Process. Syst., 2015, pp. 91–99

[28] Y. Bar-Shalom and T. E. Fortmann, Tracking and Data Association. London, U.K.: Academic, 1988.

[29] S. S. Blackman, “Multiple hypothesis tracking for multiple target tracking,” IEEE Aerosp. Electron. Syst. Mag., vol. 19, no. 1, pp. 5–18, Jan. 2004

[30] A. G. A. Perera, C. Srinivas, A. Hoogs, G. Brooksby, and W. Hu, “Multiobject tracking through simultaneous long occlusions and split-merge conditions,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR), vol. 1, Jun. 2006, pp. 666–673.

[31] Las Vegas: IEEE. Zhao, Z.Q., Zheng, P., Xu, S.T., Wu, X. (2018). Object detection with deep learning: A review. arXiv e-prints, arXiv:1807.05511.