

Original Article

A Cost-Effective Object-Oriented Approach for Multiple Vehicle Tracking using Optical Flow: Comparative Review and Insights

Nikesh Patel¹, Keyur Brhambhatt², Mahasweta Jayantbhai Joshi³

¹Gujarat Technological University, Ahmedabad, Gujarat, India.

²Department of Information Technology, B.V.M. Engineering College, V.V. Nagar, Gujarat, India.

³Department of Computer Engineering, B.V.M. Engineering College, V.V. Nagar, Gujarat, India.

¹Corresponding Author : nikeshapatel107@gmail.com

Received: 05 November 2025

Revised: 18 August 2026

Accepted: 12 September 2026

Published: 27 October 2026

Abstract - Intelligent and dynamic traffic management is the key component for smart city development. Vehicle tracking plays a vital role in providing enhancements in urban mobility and city planning for smart city development. Real-time tracking helps authorities to manage the traffic flow, to estimate the speed of vehicles, to maintain dynamic traffic control systems, etc. Computer Vision transforms the way traffic surveillance is conducted from manual to automatic, also by making the process static and automatic. This paper provides a video processing technique to track multiple vehicles using Computer Vision.

Keywords - Traffic surveillance, Vehicle tracking, Computer vision, Motion estimation, Optical flow.

1. Introduction

Traffic Surveillance is one of the major development domains in the development of smart cities in India. India is facing a rapidly urbanizing population and an increase in traffic density. For India, in 2024, the initiative-“Sangam: Digital Twin” was launched by the Department of Telecommunication to leverage technological advancements to anticipate the real-world effects of implementing innovative infrastructure planning solutions, assisting in formulating a future roadmap for sustainable and scalable development [29]. The initiative by the government took place for discussion about the transportation sector development for making Intelligent and Dynamic Traffic management of the city, which includes various measurements to be optimized, like the flow of vehicles, the average speed of vehicles [29].

Vehicle tracking is one of the important and crucial applications for traffic surveillance, combined with various technologies such as GPS, RFID and Computer Vision. Computer Vision plays an important role in intelligent traffic surveillance systems. Many traffic regulation tasks can be converted from manual work to automation by integrating computer vision-based intelligent systems. Vehicle tracks can be useful for many applications, such as Traffic Congestion, Vehicle Speed Estimation, and urban planning. This paper presents a comparative analysis of tracking vehicles using CCTV footage by combining Computer Vision technologies. This paper presents a comparative analysis of various

approaches for vehicle tracking, alongside an innovative yet space-efficient method for vehicle tracking. This method innovatively improves the verification of vehicles by storing vehicle positions by utilizing a simple array technique. This approach not only records the positions but also stores various vehicle attributes, such as contour, vehicle track in terms of their centroid pixels track, and centroid. As a result, it facilitates easier implementation and verification in subsequent video frames. Motion representation in computer vision encompasses various algorithms. Motion tracking for moving objects typically involves a two-phase process: object detection followed by object tracking. This paper presents a comparative analysis of different algorithms for vehicle tracking, specifically identifying which methods are suitable for vehicle tracking and which are not. A dataset specific to vehicles is utilized, and the implementation of various technologies is compared using this dataset. The remainder of the paper is structured as follows: Section 2 elucidates different methods for object tracking. Section 3 presents the implementation and results. Section 4 provides a comparative analysis post-implementation.

2. Multimodal vs Single Camera Approach for Vehicle Tracking

Vision-only (single camera) systems are being studied widely because of their low cost and the amount of information they provide. In contrast, a multi-modal tracking



system combined a camera with different sensors. Real-world deployment still faces hurdles in generalization [30]. LiDAR employed systems are cost-effective compared to RADAR-based systems [29]. Recent growth on RADAR underscores its efficiency to ensure robust perception under real-time environment scenarios [29]. As per the information for Camera Installation for vehicle tracking purposes, it has to be cost-effective. Also, the multi-modal systems are very useful for autonomous driving systems [30], but for traffic surveillance, single-camera systems are feasible and cost-effective.

3. Vehicle Tracking Techniques

The video analysis using image processing involves the initial step of identifying a moving object from two consecutive frames. Many applications necessitate precise detection and location data of the object during the continuous tracking process. Computer vision offers several techniques for object detection in video datasets. For Vehicle, techniques can be broadly categorized into two types: 1) motion-based and 2) learning-based. Motion-based techniques are used to estimate motion within a sequence of images. A fundamental method for detecting motion involves subtracting two consecutive frames, known as the frame-differencing method, which results in the detection of motion through pixel distribution. However, object identification cannot be achieved solely using this technique. Features such as edges and corners can be utilized as inputs for detection algorithms. Corners reduce the search time for motion tracking and can be used for subsequent tracking processes. Edges provide valuable information for highlighting the boundaries of moving vehicles. The basic algorithms for edge detection include Canny, Sobel, Prewitt, and Roberts with their kernels for horizontal and vertical edges [3]. Both horizontal and vertical edges can be considered [20]. A fixed ratio of horizontal and vertical edges is used for regions containing the vehicle as a target, such as a number plate, if vehicle [3, 20].

All these methods are conventional for detecting an object from an image; however, they are insufficient for achieving the tracking goal. They can be used for detection purposes only. Another method for object detection involves color detection using a simple histogram calculation method [28]. However, color detection is not suitable for vehicle-specific applications because of the greater number of different vehicles and having the same colors in real time. Features of vehicle targets may include multiple vehicle features (e.g., number plates). An alternative approach for object detection and measurement of distance is stereo vision, which involves computing a disparity map for two images captured by a stereo camera of the same vehicle. Although this approach is reliable, the cost of implementation increases because two cameras must be used.

Learning-based methodologies are increasingly being used for object classification, including vehicle detection, using various techniques. Vehicle detection can be implemented through segmentation and object localization processes by employing different features and classifiers for specific vehicle detections [28]. Principal Component Analysis (PCA) is another learning-based approach for vehicle detection.

The most relevant approach is the Neural Network approach, wherein a training- and testing-based model is developed for vehicle detection [1, 4]. Among the various deep learning techniques, You Only Look Once (YOLO) is a widely used model for specific vehicle detection [1, 4]. YOLO provides a higher frame rate than other models because it processes the entire frame in a single pass and detects vehicles from only two grids of the frame, resulting in lower accuracy. When applied to live data, learning-based approaches are faster but require higher configurations and greater costs, such as a GPU. Comparative study is represented in the following table (Table1).

Table 1. Comparison of different techniques for vehicle detection and tracking

Type	Technique	Methodology	Task/Goal(s)		
			Detection	Tracking	Classification
Motion based	Optical Flow [1]	Works on the Brightness Constancy Equation and assumes that neighboring pixels have similar motion	√	√	×
	Frame Difference	Subtraction (Absolute difference) of two consecutive frames	√	√	×
	Corners	Finding the corner pixel from a frame by considering a significant change of intensity in more than one direction	√	×	×
	Vertical and Horizontal Edges [3, 20]	Fixed Ratio of horizontal and vertical edged for regions with a target(vehicle)	√	√	×
	Colour [28]	Simple histogram calculation	√	×	×
	Stereo Camera [12, 13]	Computation of disparity map	√	√	×

	Feature Extraction [4]	Targets of multiple features of a vehicle (e.g., Numberplate)	√	√	×
Learning based	Features and classifiers [28]	Different classification techniques	√	×	√
	Principle Component Analysis [28]	Detection with Principal Component Analysis (PCA)	√	×	×
	Neural Network [1, 4]	Training and Testing-based Model generation of a vehicle	√	×	√

4. Learning-based Approach

There are mainly two categories of deep learning techniques: 1) Single step Algorithm and 2) Two-step Algorithm. For processing an image, two step vehicle detection algorithm can be used that considers a region-based approach. Object Detection is generally divided into a two-step procedure comprising the extraction of frames and applying classification or regression for detection.: Extracting a region proposal from a sequence of frames and classifying and reversion of frames to improve the model's detection accuracy [28]. Deep Learning techniques have been studied in [30], and by comparison, there are a few issues that are open for future work.

It is necessary to update the model for vehicle tracking in some aspects [30]. The aspect for which modification is needed is for detection or classification in Video. Traditional algorithms for object classification or for detection usually perform frame-by-frame vehicle detection, but do not perform the correlation of vehicles in consecutive frames. [30].

5. Multiple Vehicle Tracking Algorithm

In the realm of motion representation for moving objects, optical flow characterizes the pattern of apparent motion between two consecutive frames prompted by a moving object. Optical flow works on the principle of the Brightness Constancy Equation. It assumes that neighbouring pixels of feature points also have similar motions. Typically, optical flow is used with feature points identified through corner detection in the initial frame, and these points are subsequently tracked iteratively across all frames of a video.

The Lucas-Kanade method employs a 3×3 patch around a point to track motion, whereas dense optical flow calculates the optical flow for every point within the frame. As a result, the dense optical flow was slower than the Lucas-Kanade method. To effectively track a vehicle using the Lucas-Kanade method, an array of points, either corners or edges, must be identified in the initial frame as suitable points for tracking. Corner points reduce the computational overhead associated with calculating the optical flow for subsequent frames.

In this study, the optical flow calculation was optimized by focusing on the centroid points of the detected vehicle contours. To track multiple vehicles within a single frame, a

class-object approach was used to store each vehicle as an object within the frame.

The subsequent step involved verifying whether each vehicle was present in the previous frame, which was achieved through contour shape matching. For each new vehicle detected in a frame, its contour is added to the vehicle array, and its track is updated accordingly. Algorithm can be represented as follows:

1. Input Video
2. Set pre-defined Parameters for optical flow (Array p0 and p1)
3. Generate Background Subtractor and Vehicle Array
4. Generate Mask to draw the track
5. Create a vehicle Array and a vehicle object
6. For each frame :
 - a. If First frame
 - i. add vehicles for all contours found in first frame
 - b. Else:
 - i. For each contours in frame:
 1. Check whether the vehicle exists in the array or not
 2. If Exist then update the track
 - a. Get previous position and calculate distance
 - b. Calculate the speed of a particular vehicle
 3. Else, add the vehicle to the vehicle array by setting all vehicles
 4. Compute the optical flow for points
 - ii. Update p0 for optical flow
 - iii. Calculate Optical Flow and draw the track
7. Update p0 and p1
8. Repeat from step-6
9. Stop

6. Data Availability

The data that support the findings of this study are available from the Dataset from [26]: Sochor, Jakub and Juránek, Roman and Špaňhel, Jakub and Maršík, Lukáš and Šíroký, Adam and Herout, Adam and Zemčík, Pavel," Comprehensive Data Set for Automatic Single Camera Visual Speed Measurement", IEEE Transactions on Intelligent Transportation Systems, vol 20, pages=1633-1643, doi = 10.1109/TITS.2018.2825609,2019". Restrictions apply to the availability of these data, which were used under license for

this study. Data are available from the authors with their permission.

7. Evaluation Metric

The dataset that is used is the same dataset and experimental setup as described in [26] to ensure fair and consistent comparison. This work focus on classical non-deep learning tracking method for low operation complexity, so the work focus on the processing time as an evaluation metric. As such, metrics such as MOTA/MOTP that are driven by deep learning-based detection methods are not used.

8. Scalability and Real-Time Deployment

The proposed vehicle-tracking framework represents a breakthrough in computational efficiency, making it ideal for real-time use in intelligent transportation systems. By using geometric calculations based on optical flow tracking instead of resource-intensive deep learning models, this method requires significantly less processing power and memory. The distance and speed estimation stages rely on simple mathematical operations, ensuring minimal computational overhead. Consequently, the framework can handle video streams at real-time frame rates on standard computing

hardware without the need for dedicated GPU resources, offering a cost-effective solution.

Moreover, the approach is inherently scalable and ready to be adapted for multiple surveillance cameras across various traffic environments. Each camera functions independently, allowing seamless integration into distributed traffic monitoring architectures with minimal adjustments.

The framework's ability to process each tracked vehicle individually enables it to manage different traffic densities, while its scene-independent geometric formulation allows for deployment at diverse locations with minimal recalibration. These features position the proposed method as a practical and efficient solution for large-scale traffic monitoring and intelligent transportation applications.

9. Implementation and Results

For Comparison, other methods have been implemented, and a comparative study has been represented. Figure 1 indicates various algorithm implementations with their result-specific vehicle tracking. Figure 2 shows the continuous frames result for the proposed algorithm.

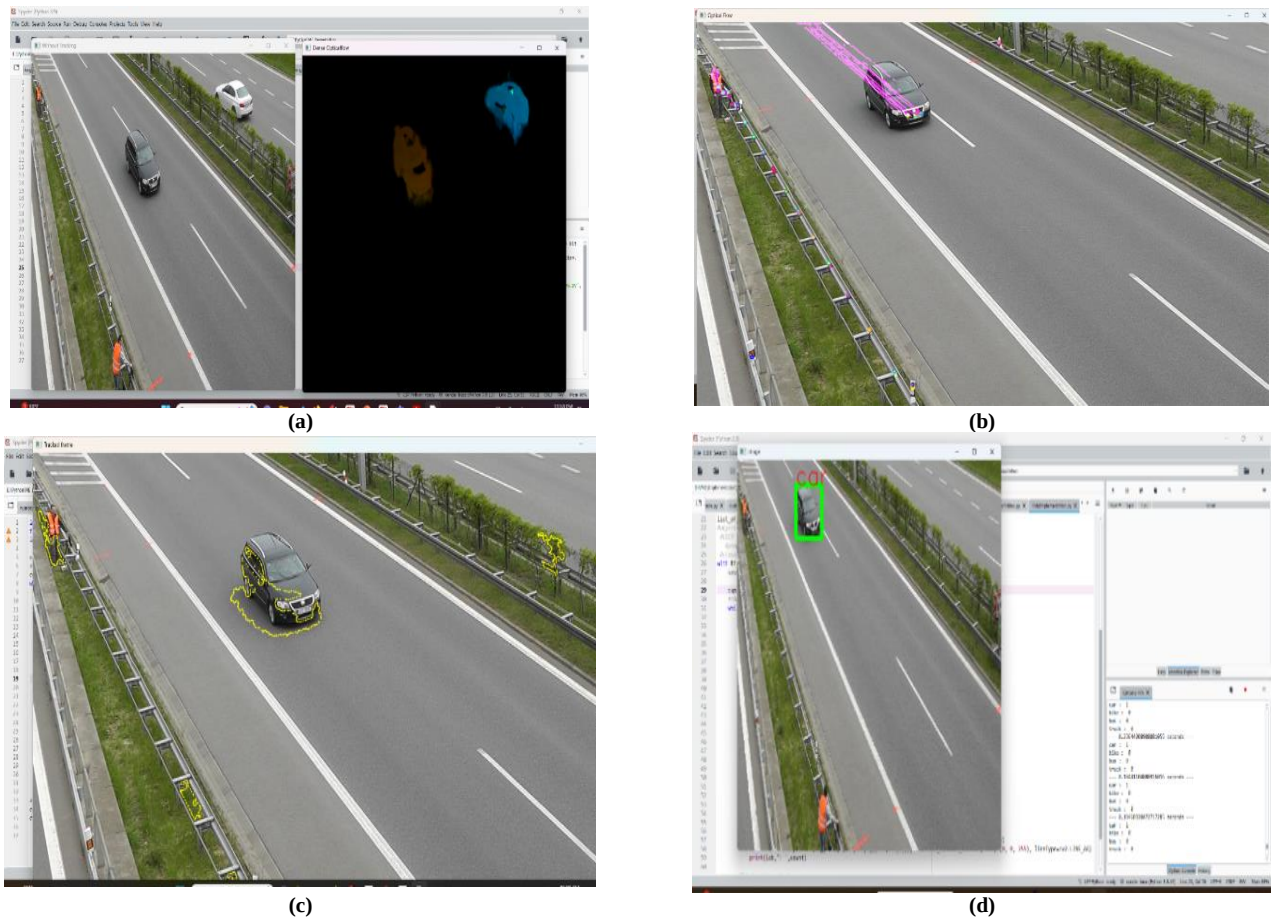


Fig. 1 Basic methods used for vehicle tracking, (a) Dense optical flow, (b) Optical flow with Shi-Tomasi corner detection, (c) Contour detection, and (d) Learning-based approach-YOLOv3.

9.1. Multiple Vehicle Tracking

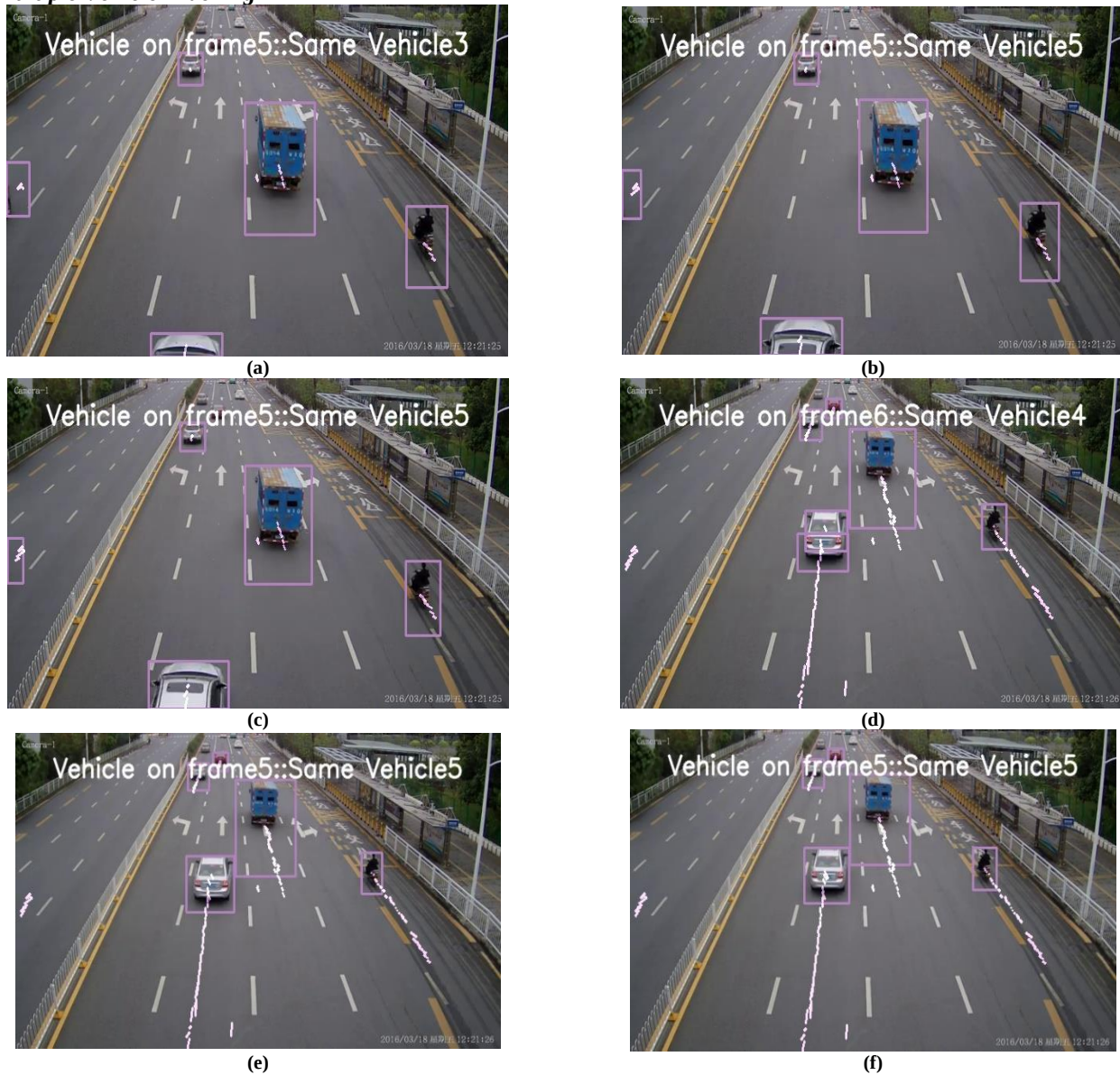


Fig. 2 Result analysis: (a-f) Multiple vehicle tracking using optical flow combined with centroid of contours.

10. Result Analysis

Table 2. Result analysis

Technique	OpenCV Function	Execution time (For each frame) (ms) (Average)	Detection Goal	Tracking Goal
Dense Optical flow	cv2. calcOpticalFlowFarneback()	547	True	√
Optical flow Lucas-Kanade	cv2.calcOpticalFlowPyrLK()	168	True	√
Multiple Vehicle Tracking Algorithm	cv2.calcOpticalFlowPyrLK(), cv2.goodFeaturesToTrack(frame_gray, mask = None, **feature_params)	87.5	True	√
Horn-Schunk Optical Flow	User-defined function to calculate Image derivatives using the Sobel Convolution kernel	560	True	√

SSD	MobileNetSSD	33	Partially True	X
Learning-based YOLO	Model generation using tensorflow library	23.1 (using GPU)	True	X
Blob Detection	cv2. simpleBlobDetector_create()	146	Partially True	X
Mean shift	cv2. meanShift()	6.5	False	X
Camshift	cv2.camShift()	8.6	False	X
Frame Difference	Consecutive frame subtraction	7.9	True	X
Contour Detection	cv2.findContours()	89.2	Partially True	√
Background Subtraction	cv2. createBackgroundSubtractorMOG2()	26.3	True	X

11. Conclusion

Major of existing on-road vehicle detection techniques use a “detection-then-track” approach. As per the result analysis, for vehicle tracking, optical flow with canny edge and shi-tomasi corner detection gives better execution time and detection. For real-time applications such as over-speed detection, there is a need to have low cost and less time-consuming calculation of speed and hence the tracking implementation is crucial part. The proposed algorithm gives less time as compared to other algorithms.

Conflicts of Interest

The authors declare that they have no known financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Author Contribution Statement

NP has conducted a literature review, experiment design, analysis, comparison and performed an initial draft. KB & MJ have supervised the study, contributed to making an interpretation of the results, and revised the manuscript for intellectual content.

References

- [1] Jahongir Azimjonov, and Ahmet Özmen, “A Real-Time Vehicle Detection and a Novel Vehicle Tracking Systems for Estimating and Monitoring Traffic Flow on Highways,” *Advanced Engineering Informatics*, vol. 50, pp. 1-12, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [2] Shaohua Wan, Songtao Ding, and Chen Chen, “Edge Computing Enabled Video Segmentation for Real-Time Traffic Monitoring in Internet of Vehicles,” *Pattern Recognition*, vol. 121, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [3] Elnaz Vakili, Maryam Shoraran, and Mohammad R. Sarmadi, “Single-Camera Vehicle Speed Measurement using the Geometry of the Imaging System,” *Multimedia Tools and Applications*, vol. 79, no. 27-28, pp. 19307-19307, 2020. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [4] Gaurav Dubey et al., “Vehicle Detection and Speed Tracking based on Object Detection,” *Proceedings of International Conference on Big Data, Machine Learning and their Applications*, Singapore, Springer, vol. 150, pp. 403-409, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [5] S. Sri Jamiya, and P. Esther Rani, “An Efficient Method for Moving Vehicle Detection in Real-Time Video Surveillance,” *Advances in Smart System Technologies*, Singapore, Springer, vol. 1163, pp. 577-585, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [6] Hari Krishnan P.M et al., “Fast Approach for Moving Vehicle Localization and Bounding Box Estimation in Highway Traffic Videos,” *Signal, Image and Video Processing*, vol. 15, no. 5, pp. 1041-1048, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [7] Shipra Ojha, and Sachin Sakhare, “Image Processing Techniques for Object Tracking in Video Surveillance: A Survey,” *2015 International Conference on Pervasive Computing (ICPC)*, Pune, India, pp. 1-6, 2015. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [8] Sen-Ching Cheung, and Chandirka Keumath, “Robust Techniques for Background Subtraction in Urban Traffic Video,” *Visual Communications and Image Processing, SPIE*, vol. 5308, 2004. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [9] D. Jeyabharathi, and D. Deje, “Vehicle Tracking and Speed Measurement System (VTSM) based on Novel Feature Descriptor: Diagonal Hexadecimal Pattern (DHP),” *Journal of Visual Communication and Image Processing*, vol. 40, pp. 816-830, 2016. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [10] Jinhui Lan et al., “Vehicle Speed Measurement based on Gray Constraint Optical Flow Algorithm,” *Optik*, vol. 125, no. 1, pp. 289-295, 2013. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [11] Diogo Carbonera Luvizon, Bogdan Tomoyuki Nassu, and Rodrigo Minetto, “A Video-based System for Vehicle Speed Measurement in Urban Roadways,” *IEEE Transaction of Intelligent Transportaation Systems*, vol. 18, no. 6, pp. 1393-1404, 2017. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]

- [12] Abdelmoghith Zaarane et al., “Distance Measurement System for Autonomous Vehicles using Stereo Camera,” *Array*, vol. 5, pp. 1-7, 2020. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [13] Yasir M. Mustafah et al., “Stereo Vision Images Processing for Real-time Object Distance and Size Measurements,” *2012 International Conference on Computer and Communication Engineering (ICCCE)*, Kuala Lumpur, Malaysia, pp. 659-663, 2012. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [14] Jakub Sochor, Roman Juránek, and Adam Herout, “Traffic Surveillance Camera Calibration by 3D Model Bounding Box Alignment for Accurate Vehicle Speed Measurement,” *Computer Vision and Image Understanding*, vol. 161, pp. 87-98, 2017. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [15] Daniil A. Loktev, and Alexey A. Loktev, “Estimation of Measurement of Distance to the Object by Analyzing the Blur of Its Image Series,” *2016 International Siberian Conference on Control and Communications (SIBCON)*, Moscow, Russia, pp. 1-6, 2016. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [16] Harasim Eugen, and Codrin Donciu, “Video Distance Measurement based on Focus,” *2020 International Conference and Exposition on Electrical and Power Engineering (EPE)*, Iasi, Romania, pp. 582-588, 2020. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [17] Viet Hoa Do et al., “A Simple Camera Calibration Method for Vehicle Velocity Estimation,” *2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, Hua Hin, Thailand, pp. 1-5, 2015. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [18] Wan-Ping Wu et al., “Design and Implementation of Vehicle Speed Estimation using Road Marking-based Perspective Transformation,” *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*, Helsinki, Finland, pp. 1-5, 2021. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [19] Genyuan Cheng et al., “Real Time Detection of Vehicle Speed based on Video Image,” *2020 12th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, Phuket, Thailand, pp. 313-317, 2020. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [20] Saleh Javadi, Mattias Dahl, and Mats I. Petterson, “Vehicle Speed Measurement Model for Video-based Systems,” *Computers and Electrical Engineering*, vol. 76, pp. 238-248, 2019. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [21] Manuel Ibarra Arenado et al., “Monovision-based Vehicle Detection, Distance and Relative Speed Measurement in Urban Traffic,” *IET Intelligent Transport Systems*, vol. 8, no. 8, pp. 655-664, 2014. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [22] Zhegguang Xu, Luyao Wang, and Jie Wang, “A Method for Distance Measurement of Moving Objects in a Monocular Image,” *2018 IEEE 3rd International Conference on Signal and Image Processing (ICSIP)*, Shenzhen, China, pp. 245-249, 2018. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [23] David Fernández Llorca, Antonio Hernández Martínez, and Iván García Daza, “Vision-based Vehicle Speed Estimation: A Survey,” *IET Intelligent Transportation Systems*, vol. 15, no. 8, pp. 987-1005, 2015. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [24] Emanuele Trucco, and Alessandro Verri, *Introductory Techniques for 3-D Computer Vision*, Prentice Hall, Englewood Cliffs, 1998. [[Google Scholar](#)] [[Publisher Link](#)]
- [25] Feature Extraction, 2026. [Online]. Available: <https://www.geeksforgeeks.org/machine-learning/what-is-feature-extraction/>
- [26] Jakub Sochor et al., “Comprehensive Data Set for Automatic Single Camera Visual Speed Measurement,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 5, pp. 1633-1643, 2018. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [27] D. Willersinn, and W. Enkelmann, “Robust Obstacle Detection and Tracking by Motion Analysis,” *Proceedings of Conference on Intelligent Transportation Systems*, Boston, MA, USA, pp. 717-722, 1997. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [28] Amir Mukhtar, Likun Xia, and Tong Boon Tang, “Vehicle Detection Techniques for Collision Avoidance Systems: A Review,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2318-2338, 2015. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [29] Nicolas Baumann et al., “CR3DT: Camera-RADAR Fusion for 3D Detection and Tracking,” *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Abu Dhabi, United Arab Emirates, pp. 4926-4933, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [30] Hui Qian et al., “A Review of Multi-Sensor Fusion in Autonomous Driving,” *Sensors*, vol. 25, no. 19, pp. 1-34, 2025. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]