Vehicle Detection and Counting System

Prof. Sonali Guhe
Asst. Prof Dept of Information Technology
G.H. Raisoni College of Engineering, Nagpur.

Pranoti Ghotekar
Department of IT
GHRCE
Nagpur
pmghotekar123@gmail.com

Shreya Khapekar
Department of IT
GHRCE
Nagpur
shreyakhapekar24@gmail.com

Lavesh Lingayat
Department of IT
GHRCE
Nagpur
lingayat.lavesh125@gmail.com

ABSTRACT –

In traffic surveillance systems, where effective traffic control and security are the main considerations, vehicle recognition and tracking play a vital and effective role. We explore and handle the problem of extracting vehicle and traffic information from video frames in this report. Even though numerous studies have been conducted and numerous approaches have been used, there is still potential for advancement in this field. Despite the fact that numerous studies and methodologies have been adopted in this field, there is still space for improvement. For observing, organising, and managing traffic flow, passing car identification, examining, and counting are vital.

In contrast to previous methods, a video-based solution is simple to deploy and does not impede traffic. In this project, a video-based solution is developed after analysing a series of traffic footage captured by a camera and using adaptive subtracted background technology, virtual detector technology, and blob tracking technology. Experiments were carried out using Visual Python code and OpenCV development kits, and the results show that the proposed method may precisely identify, follow, and quantity moving autos.

Keywords:

Image Processing, Vehicle Detection, Vehicle Tracking, Vehicle Counting, Traffic control.

INTRODUCTION-

Vehicle detection and tracking are useful and important in traffic monitoring devices where effective traffic control and security are the main objectives. The topic of extracting car and traffic information from video frames is covered in detail.

Although numerous studies have been conducted in this field and numerous methodologies have been used, there is still potential for advancement. It is suggested to create a special algorithm for tracking and identifying vehicle data in order to make improvements. Understanding the background first aids in separating the forefront and backstory in frames. To build rectilinear areas around each discovered object, a binary calculation is used to recognize the object in the front-facing detector.

I. LITERATURE SURVEY

A vehicle detection system is an object detection system application in which the

major focus is on tracking the vehicle using coordinates in photos or videos. This system can be implemented using a variety of tools, and in the following part, we will do so with the OpenCV library.

The Open Source Computer Vision (OpenCV) Library is a Python library that includes coding functions for implementing real-time computer vision and its applications.

Vehicle speed measurement, traffic flow prediction, and traffic collision identification are some real-world initiatives employing vehicle detection in the Intelligent Transportation System (ITS). Such a technology seeks to identify the coordinates of the car and concentrate on that specific vehicle in the video. For instance, have a look at the image frame, where a bounding box is generated to enclose the car and identify its coordinates.

II. PROPOSED METHODOLOGY

System Requirement Analysis:
The analysis of the system's needs is the first phase in the process methodology for a vehicle detection and counting system. The study should cover the function of the system, the kinds of cars that need to be found, the setting in which it will be used, and the accuracy standards. The cost and scalability of the system should also be taken into account throughout the analysis.

Hardware Selection:
The selection of the auxiliary equipment comes after the analysis of the system needs. The system prerequisites, such as the camera's resolution, the computer's processing speed, and the memory needs, should be considered when selecting hardware equipment. When selecting the hardware, the price and availability of the components should be considered.

Software Development:
The software for the vehicle detection and counting system must be created next. The hardware components chosen and the needs of the system should serve as the basis for developing the software. Image processing algorithms like background subtraction, edge identification, and object detection algorithms like SVM or CNN should all be included in the software. Algorithms for tracking and counting vehicles should also be included in the software.

System Integration and Testing:
The system testing and hardware and software integration process come next after the software development. The seamless operation of all hardware and software components should be guaranteed by the integration. To verify the system's accuracy and dependability, testing should take into account a variety of scenarios, such as diverse lighting conditions and vehicle kinds.

Deployment:
The final step is to deploy the system after it has been integrated and tested. The system's accuracy requirements should be met, and the deployment process should take into account the environment in which the system will be used. The deployment should also specify the necessary user training and maintenance.

Continuous Improvement:
A vehicle detection and counting system's process approach should include a component for continual improvement. Regular monitoring of the system is

necessary to spot any problems and enhance its precision and dependability.

In order to increase the system's performance and capabilities, continuous improvement should incorporate new technologies and methodologies.

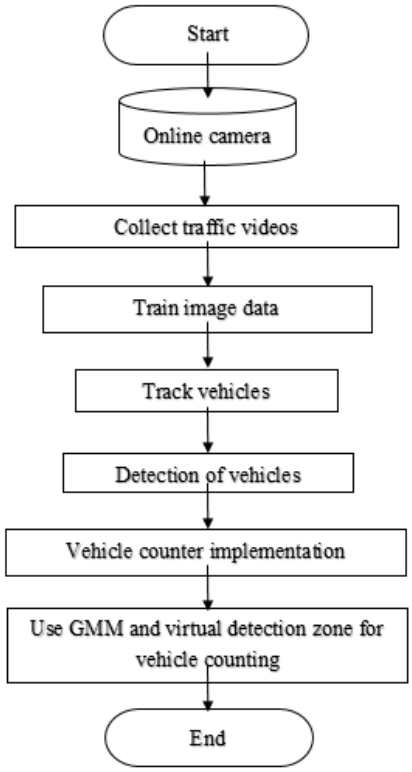

Fig: Flowchart for counting, recognizing and following vehicle

III. SYSTEM FRAMEWORK

The common components of a framework for a camera-based vehicle detection and counting system comprise a camera, image processing, object tracking, vehicle counting, user interface, database, and system administration components. The system framework must be properly built and set to suit the system's requirements because each component is essential to the system's accuracy and dependability. Continuous testing, monitoring, and optimization can raise the system's performance.

IV. CONCLUSION

We began by downloading the video and then worked on the aforementioned algorithm. We learned how to detect objects using OpenCV and Python, as well as how to identify vehicles using various techniques.

This has a broad scope and can be used for object detection. It has the ability to recognise items, and we can also customise it to work with particular objects.

V. REFERENCE

1. N. K. Kanhere, S. T. Birchfield, W. A. Sarasua, and T. C. Whitney, “Real-Time detection and tracking of Vehicle base fronts for measuring traffic counts and speeds on highways, “Transportation Research Record, No. 1993, 200

2. Al-Smadi, M., Abdulrahim, K., Salam, R.A. (2016). Traffic surveillance: A review of vision based vehicle detection, recognition and tracking. *International Journal of Applied Engineering Research*, *11*(1), 713–726.

3. Hortovanyi, N. (2017, February 24). Vehicle Detection and Tracking. Towards Sata Science.

4. Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), 273–297.

5. Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V.,

Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., Vanderplas, J., Joly, A., Holt, B. And Varoquaux, G. (2013). API Design For Machine Learning Software: Experiences From The Scikit-Learn Project, 108–122.