RAPID VEHICLE SUPERVISION AND DETECTION IN A TERRITORY

Lakshmi T[1], SafiyaBegam G[2], AsisJovin[3]

[1, 3]*Department of AI & DS,KarpagaVinayaga College of Engineering and Technology,Tamil Nadu,India*
[2]*Department of Computer Science,B.S.AbdurRahman Crescent Institute of Science and Technology,Tamil Nadu,India*

ABSTRACT:

Due to recent significant advancements in the processing power of basic computers, deep learning algorithms are now frequently utilized to analyze traffic video footage. Estimating traffic flow, spotting anomalies, identifying vehicles, including video vehicle tracking are some of the fundamental elements of traffic analysis. Predictions of flow of traffic or estimates of speed have been one of the most prominent study topics in recent years. Good solutions to this issue may help decrease accidents on the roads and improve roadway construction by more accurately estimating transit demand. In this paper, we present a number of novel methods for measuring the speed of moving objects, as well as a solution to the monitoring problem.

KEYWORDS: Deep Learning, Vehicle Tracking, Traffic, Speed Monitoring.

1.INTRODUCTION:

Technology growth has transformed many parts of our life, including communication and security. One of the most difficulties that authorities and organizations confront is effectively monitoring and controlling cars in a specific zone. Whether for road safety, law enforcement, border security, or critical infrastructure monitoring, the necessity for precise and timely vehicle supervision and detection has become vital.Rapid Vehicle Supervision and Detection is the process of monitoring, identifying, and tracking cars in a given region in real time utilizing modern technology and intelligent systems. This strategy involves fusing advanced software and machine-learning algorithms with cutting-edge hardware like cameras, sensors, and radars.Quick automobile supervision is based on cutting-edge technologies including deep learning, vision, and machine learning. These systems can manage enormous volumes of data from numerous sources, including CCTV cameras, satellite imagery, and other sensors, to detect and identify autos successfully.

Background modeling yields the highlighted portion of the video stream by subtracting consecutive frames. The exact location of the vehicle is detected through foreground detection. Centroids and a histogram of the removed picture are used to track the vehicle. This technique is also useful for determining vehicle velocity. The initial step in estimating vehicle speed is to accurately recognize the cars in the video.

For years, computer vision techniques have been used to study object detection. Traditional computer vision approaches typically evaluate the shape, contrast, and other properties of an item to detect it. The deep convolution neural network (CNN), in particular, has improved object detection thanks to the efficient application of computational intelligence technology. Approaches for identifying objects that remove context and differ in frames can be categorized.

1.1 Analyzing the Threshold value:

In this case, we will evaluate the hue level of an item while excluding the backdrop sceneries. We shall be able to distinguish between vehicle and background by taking threshold intensities into account for an area. As a result, determining the appropriate threshold value becomes critical. If this threshold value is too low, we may miss the specific car. And if the threshold value is higher, there is a risk that the noise may be detected together with the car. Using this method, determining the minimum and maximum intensities is quite challenging. For object detection, gray-scale morphological techniques have been applied.

Background subtraction using a Gaussian mixture model is the best strategy for object detection. There are several approaches available, but the most effective is the Gaussian Mixture Model (GMM). The GMM acts at the pixel level and is most commonly used for still backgrounds. The difference between two successive frames is used to detect foreground items in the background using a subtraction approach.

To get the necessary results, frame transistors have to be eliminated from the needed video using the model developed by the GMM for background removal. It should respond to frame changes caused by the object's start and stop. Vehicle tracking from a video with automobiles in motion is a difficult undertaking. Many challenges can develop during vehicle tracking, such as a lack of hard vehicle structure, variations in appetite pattern, and occlusions connecting objects-to-scene. In general, tracking is the challenge of detecting an object's trajectory motion as it travels about in a scene. The object tracker labels all of the tracked cars here. In other words, a tracker can also offer centric-information, vehicle orientation, and vehicle shape.

So many techniques have been presented, but few of them are best suited to the requirement of employing best at night. As a result, you are leaving this section for future expansion. For accurate findings, we can develop a system that includes a calibrated camera with a high resolution that can assist us in low-light situations.

.

2.LITERATURE SURVEY:

RADAR systems were traditionally used to monitor vehicle speeds. The main disadvantage of RADAR is its high cost and lack of precision. The technology is highly costly, less precise, and requires uninterrupted line of sight connectivity between the RADAR and the cars. To circumvent the limitations of existing approaches, the suggested system employs the Gaussian methodology. As a result, numerous methods of image processing are applied to conquer the limitations of existing technologies. In general, tracking is the challenge of detecting an object's trajectory motion as it travels about in a scene.

Due to the creation of a bounding box using the KLT algorithm, it does not maintain the speed estimation. As a result, that algorithm lacks the consistency factor of speed estimation. The accuracy factor of creating bounding boxes while tracking vehicles is not working. It only accepts two frames as input. That algorithm also lacks the efficiency factor.

2.1. AUTOMATED TECHNIQUE DEVELOPMENT FOR MOTOR RATE PREDICTION AND MONITORING IN THE MEDIA STREAM.

To estimate vehicle speed in the past, inductance cycles, RADAR weapons, Lasers guns, or human counts were used. Due to their expensive price and poor precision, radars are not suited for traffic research systems and inductive loops have large installation and maintenance costs. The spherical projection equation's estimation of the car's speed set the motion in motion. According to previously proposed vehicle detection research, multiple cameras are used to record objects that the first camera is unable to capture. The technique occasionally uses as many as twenty monitors to examine moving cars. Likewise, a brand-new characteristic called "vehicle predictability" has been established which classifies the vehicle type.

2. 2.EVALUATION OF VEHICLE SPEED FROM SURVEILLANCE VIDEO INFORMATION.

Traffic data is essential in Intelligent Transportation Systems (ITS) research and implementations. Current approaches, such as deploying roadside radar sensors upon roads electromagnetic loop detectors at junctions, GPS data acquired from probe fleets, and so on, have made a wealth in information available for roadway monitoring. Aside from such technologies, surveillance cameras are frequently utilized but have not yet been employed to collect traffic data. It might be a worthwhile task to switch the camera's use beyond inspection by hand to regular surveillance. Agencies may deploy surveillance cameras for a variety of purposes, including speed data collection and traffic flow analysis in addition to visual confirmation of events. The following are some advantages: a) faster R-CNN multi-object identification; b) summary comparison multi-object tracking; and c) warping using a linear viewpoint distortion for speed conversion. It would be a good challenge to switch the camera's use from periodic inspection to ongoing surveillance.

2.3. MOTOR SPEED IDENTIFICATION ALGORITHM ANALYSIS IN CAMERA SURVEILLANCE

Automobile recognition is the most fundamental and crucial component of intelligent transportation. Infrared detection, inducement line detection, ultrasonic detection, acoustic array detection, radars, video image detection systems, and other traditional detection methods are included in this research. Various novel vehicle recognition algorithms in video surveillance have been discovered during the previous few decades. Vehicle acceleration, flow, kind of vehicle, and other useful data can be gathered from cctv pictures using methods for the identification of moving vehicles that are both inexpensive and very effective.

The research investigated the vehicle speed identification technique in video based on shifting target detection. The primary tasks are motion detection, vehicle tracking, and vehicle speed estimate.The vehicle's approximation velocity is computed by examining the centroids location of the moving target and the mapping connection.The approach has high resilience and practicability, however there is some mistake. The next step is to integrate the speed computation with more detailed information about visual motion.

3. PROPOSED WORK:

The suggested method estimates the speed of the vehicle approaching the camera by following its motion over a succession of photos. The suggested system comprises mostly of five components. The collected video is transformed into frames during preprocessing, and noise is eliminated using the Median Filter approach. The background Subtraction (Mixture of Gaussian) approach is utilized in moving vehicle identification. We are analyzing films collected by stationary cameras and using an optical-flow based approach to recognize several cars. Multiple pictures are detected at various moments using this approach, and the optical-flow-based technique identifies barriers indirectly by evaluating the velocity field.The identified foreground is cleared using morphological techniques such as opening. To detect each item, the Connected Component Method is employed. The ShiTomasi approach is used to detect corners in feature extraction. Vehicles are tracked using the optical flow approach in vehicle tracking. The centroids of each vehicle are computed during speed estimate. Centroids are monitored throughout a number of frames. The time and distance traveled by the vehicle are used to compute velocity.

3.1 STEPS INVOLVED IN PROPOSED SYSTEM:

1. The vehicle's speed is measured via tracking, and an image sequence is acquired.
2. The collected video is transformed into images and the unwanted information is eliminated using the Median Filter method during preprocessing.
3. The CNN algorithm, or Convolution Neural Network, is employed for this.
4. Vehicle velocity is computed using the distance traveled and the time consumed.
5. Pixels are deemed foreground when the separation between the backdrop and the present picture is large enough.

3.2 VEHICLE TRACKING:

Several techniques can be used to recognize vehicles from captured footage. It is challenging to recognize moving automobiles in video with accuracy. The moving vehicle detection classifies each of the pixels in the movie's frames as either salient or background data. Frame differencing and the background removal technique are two ways for vehicle detection. The frame differencing approach can be used to exclude static items (those that are not moving) and consider moving things. The fundamental disadvantage of this strategy is that slow moving items are not identified, necessitating improvisation.

Backdrop subtraction is the process of removing the backdrop or static scene from a video picture. Because the camera is stationary, each pixel in the picture has an associated number that is essentially constant across time. The goal of subtracting backgrounds is to determine the background value of each picture point. We utilized the Gaussian Mixture approach to detect the vehicle's motion since it overcomes the limitations of the other methodologies. The best method for eliminating the backdrop and considering the car from the video is background subtraction, in which each frame of the footage is compared with a reference or backgrounds model, and the pixels where divergence from the background is noticed are deemed to be the vehicles.

Accuracy of creating bounding boxes is better than existing System. Speed estimation is also taken a lesser time while compare to KLT algorithm.Efficiency of tracking a vehicle from captured video is also high in that proposed techniques.More often using object detection technique in vehicle tracking is better than using RADAR techniques for traffic signal monitoring.

Estimating vehicle speed is frequently beneficial in system as well as other processing of images application. For better results, the video quality is initially enhanced by applying noise reduction techniques such as the Median filter to the input footage. The next step is to remove the backdrop or static scene, leaving only the foreground. There are several strategies available here, and the particular one we utilized is a Gaussian mixture. The resulting foreground is cleaned up using morphological approaches such as opening. Optical flow is used to track vehicles in the next phase. And the axis centers of all objects are determined at each occurrence to anticipate their future motion. Finally, the speed of moving vehicles is approximated based on the distance traveled.

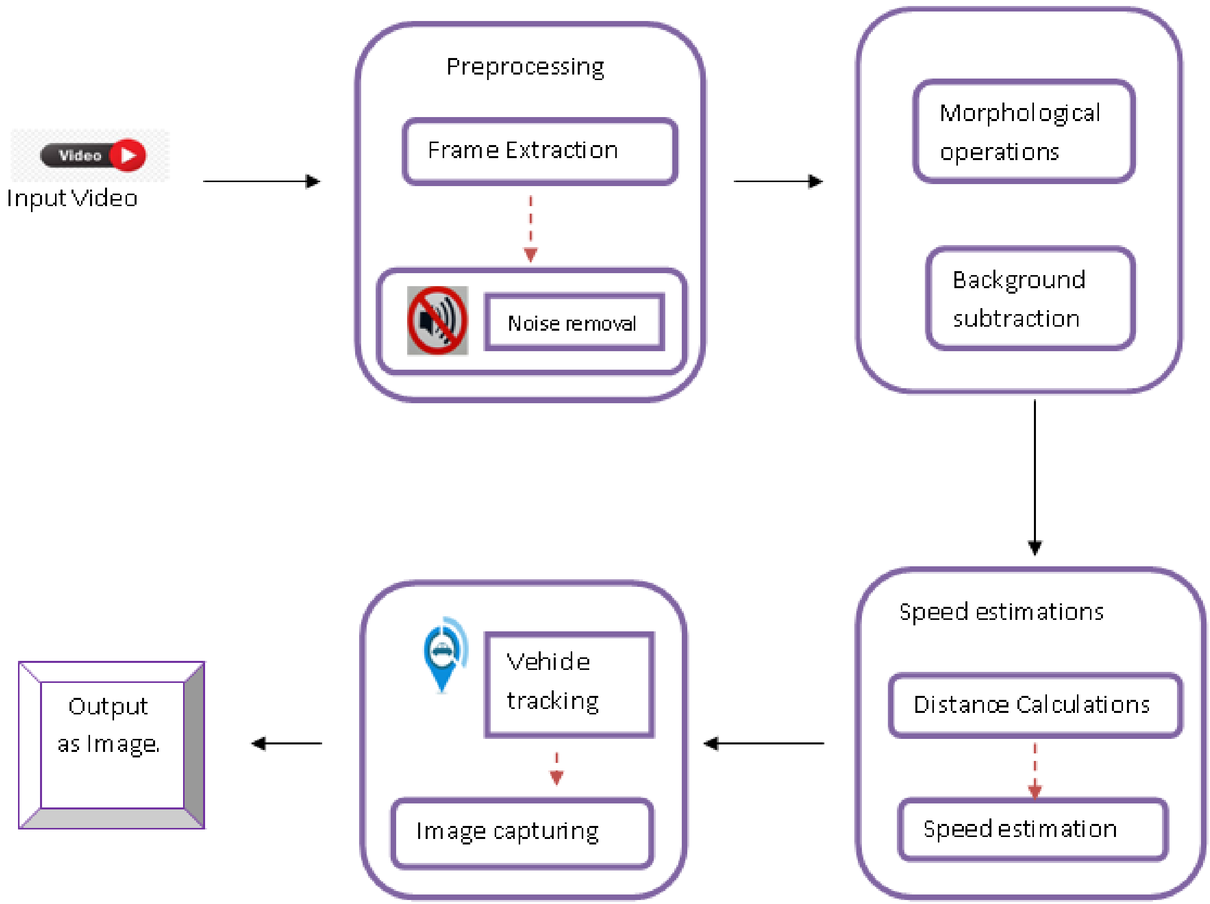

Figure 1: SYSTEM ARCHITECTURE

3.2.1.Preprocessing:

The first stage is video preparation, which includes frame segmentation and video enhancement. The provided video is transformed into n frames and the frame rate is retrieved during preprocessing. In the following stage, the video quality is increased by minimizing background noise. The median filter method is used for this. This is a filtering nonlinear technique that is frequently used to eliminate noise from the background and so produce a better foreground. This approach is frequently used as an initial process to increase the accuracy for further processing. Although there are other approaches, we favor the median filtered since it keeps the edges while reducing noise.

3.2.2. Detection of Moving Vehicles:

Background subtraction is the next stage. It is a method in the realm of computational imaging that extracts the foreground of a picture. It is also known as Background Detection. Vehicles in the foreground are the image's regions of interest. This approach is frequently used in object localization. It is assumed here that the intensity values of each pixel in the video may be described using a model consisting of a Gaussian mixture (MOG). A general rule defines which intensities belong to the background and which belong to the foreground. Foreground pixels are those that are incompatible with any of these. The pixels in the foreground are categorized using 8 connectivity linked component analysis.

3.2.3. Extraction of Characteristics:

Corner identification is a popular method for extracting specific types of features. It is mostly used to determine the content of a picture. There are certain corner selection criteria in the Harris corner detector. A score is assigned to each pixel in the pictorial representation, and if the score exceeds a certain threshold, the pixel is designated as a corner; otherwise, it is not.

3.2.4.Vehicle Monitoring:

The practice of tracing a moving item in relation to frames is referred to as vehicle tracking. Tracking is accomplished by extracting features (Shi-tomasitechnique) from elements in a frame then discovering the items in order. We calculate the position relative to the moving item through looking at the relative position values of the object in each frame. It is the pattern of movement of picture objects between two successive frames produced by camera or object movement. It is a 2D vector field, with each vector representing the progression of points from one frame to subsequent.

3.2.5. Speed Prediction:

The previous and next frame's centroids are used to determine distance. The distance determined by Euclid is used to calculate distance. As the car enters the ROI, time is computed. The speed is initially determined using the formula. Distance travelled= (x2-x1)2+ (y2-y1)2 Speed=Distance travelled/Time taken if (x1, y1) are the centroids of the vehicle in the first frame and (x2, y2) are the centroids of the second frame.

3.3 TEST CASE DIAGRAM:

A case for testing is a documentation that has been created for a specific test situation to assess compliance against a specified requirement. It contains a set of test knowledge, requirements, desired outcomes, and post conditions.

3.3.1. USECASE DIAGRAM:

A diagram of uses is a representation of a user's interaction among the software that shows how the user interacts with all of the applications in which they are involved.

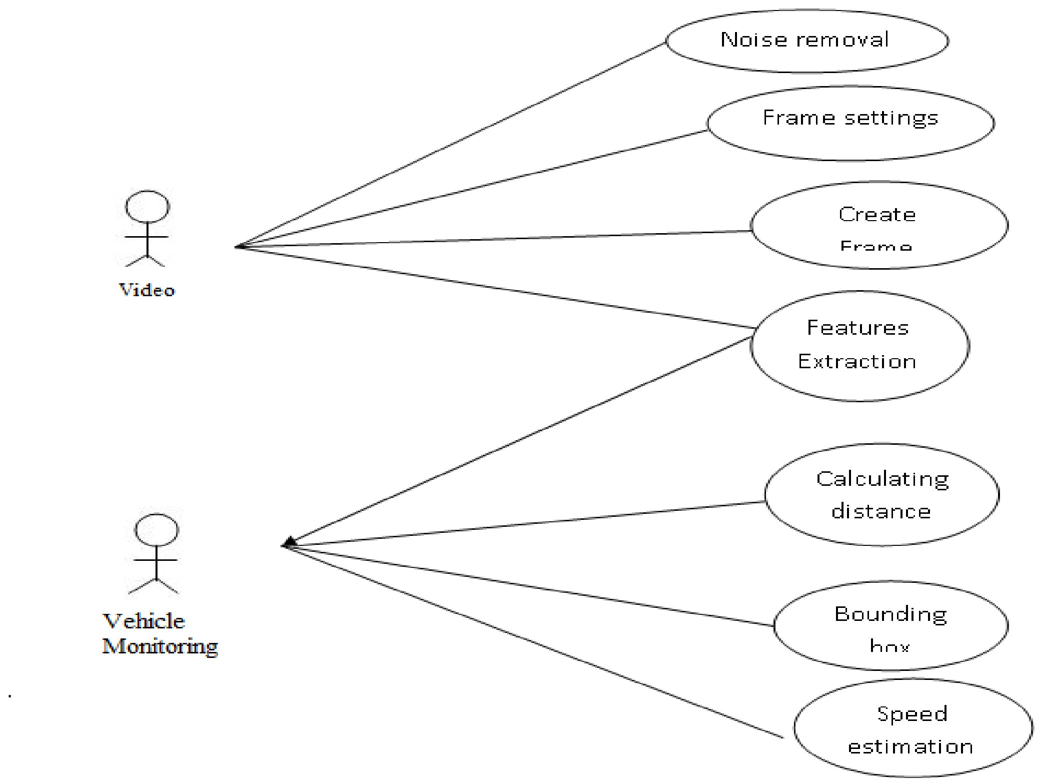

Figure 2: USE CASE DIAGRAM

3.3.2.SEQUENCE DIAGRAM:

In the sequence graph, the interactions between objects are shown in chronological order. It provides examples of the kinds of objects and classes that participate in the situation along with the organized group of messages delivered between the various objects needed to implement the capability of the situation at hand.

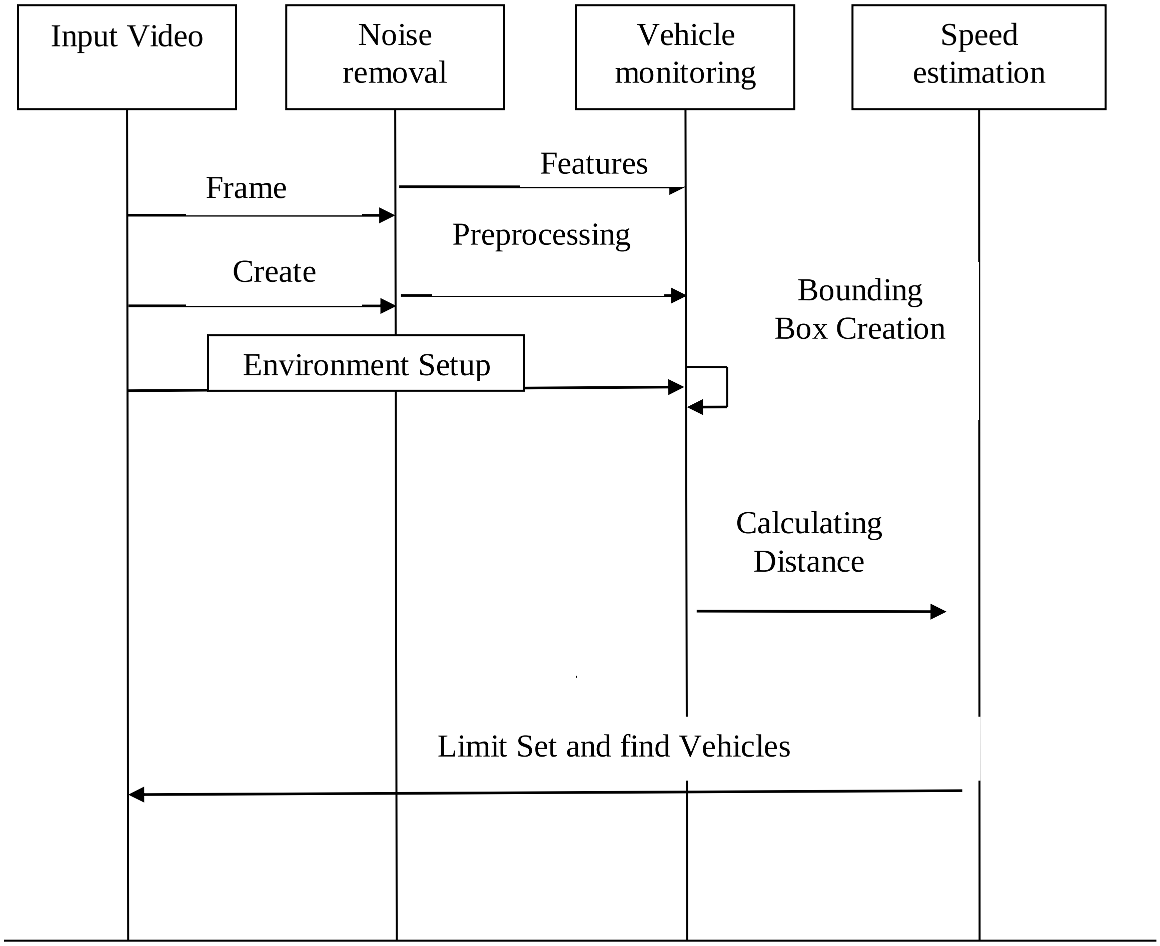

Figure 3: Sequence Diagram

<u>3.3.3.COLLABORATION DIAGRAM</u>

A conversation diagram in the unified model language 2.0 is a reduced version of a collaboration diagram in UML

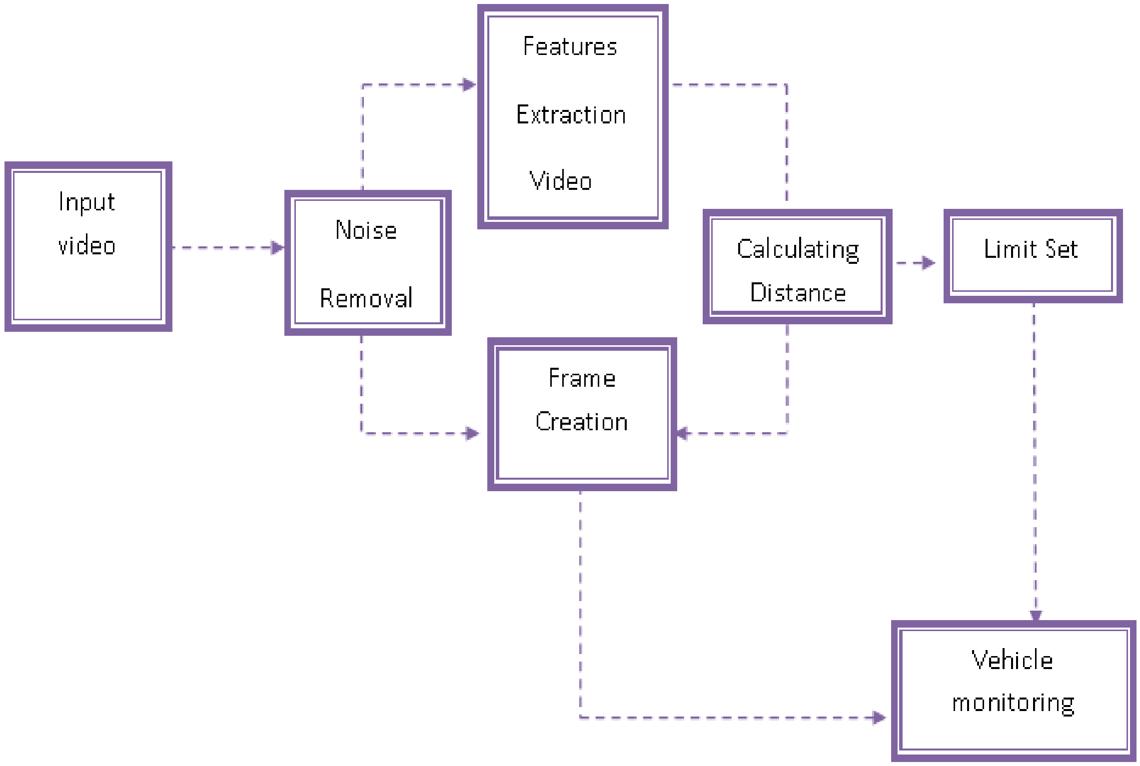

Figure 4: Collaboration Diagram

4. EXPERIMENTAL RESULTS:

Errors and prejudice can enter a system of monitoring at any point. Because records from surveillance are used for determining high-risk populations, focus treatments, and assess interventions, it is critical to understand the system's strengths and weaknesses. So far, the discussion of characteristics has focused on the information gathered for cases, although morbidity and death rates are estimated in many monitoring systems. The numbers used for these rate computations are frequently collected from a wholly independent data system administered by another organization, such as the Census Bureau. Consider the comparability of the categories (for example, race, age, and residency) on which the numerator and denominator of rate computations are based.

4.1 TESTING STRATEGIES:

Relating technology with customer criteria and hardware requirements is an aspect of software testing. Testing can take place at the level of the modules of the executable code or at the process level in the software development life cycle. A part of software testing is assurance and confirmation.

4.2 TESTING LEVELS:

1. TESTING OF UNITS
2. TESTING OF INTEGRATION
3. SYSTEM EVALUATION
4. EVALUATION OF FUNCTIONALITY
5. EVALUATION OF PERFORMANCE
6. SECURITIES AND PORTABILITY
7. TESTING FOR ACCEPTANCE
8. ALPHA TESTING
9. BETA TESTING
10. REGRESSION TESTING

1. TESTING OF UNITS

While coding, a programmer tests the unit of software to see if it is error-free. The white-box testing technique is used for testing. Unit testing allows developers to ensure that individual software units are operating properly and without errors.

2.TESTING OF INTEGRATION

Although the application elements function effectively on their own, it is important to establish whether the components will function properly when combined. Taking argument transmission and data updating as examples.

3. SYSTEM EVALUATION

After being packaged as a product, the software is tested in its entirety. Any combination of the examinations that follow may be used to do this:

4. EVALUATION OF FUNCTIONALITY

Checks all of the applications features against the specification.

5. EVALUATION OF PERFORMANCE

This test illustrates the program's effectiveness. It assesses the effectiveness of the program and the typical time required to complete the intended task. Performance testing techniques called load and stress assessments subject the application to heavy user and information loads across a range of conditions.

6. SECURITIES AND PORTABILITY

Whenever something is designed to run across a variety of technologies and be used by a sizable audience, these tests are carried out.

7. TESTING FOR ACCEPTANCE

The program must go through one final round of evaluation, that involves test for user engagement and response, before it is prepared to be delivered to the client. This is important because, regardless of whether the application satisfies all the user's requirements, it could still be rejected if the user dislikes the way it appears or behaves.

8. ALPHA TESTING

In order to simulate a functioning environment, the development team performs alpha testing on the system. They make an effort to predict how a user will react to software operations and the way the system will react to user inputs.

9.BETA TESTING

The program is distributed to subscribers to be examined primarily in their manufacturing setting after it has through internal testing. This doesn't constitute the final item. Customers are likely to bring little problems that were disregarded in order to participate at this point, according to developers.

10.REGRESSION TESTING

A software update that adds additional code, an option, or performance is thoroughly evaluated to determine whether the added code having adverse consequences. Regression testing is what's done in this situation.

4.3. BLACK-BOX TESTING:

It's done to evaluate how well the program works. Testing that is 'Behavioral' is another name for it. In this case, the anticipated outcomes and a set of the inputs are provided to the examiner. The program is tested as "ok" when input is provided if its outcome matches the desired results, and as "problematic" when it does not. In this kind of testing approach, testing technicians and end users test the software; the tester is not aware of the design or arrangement of the code.

4.3.1TECHNIQUES FOR BLACK-BOX TESTING:

1.The equivalent class - The data are grouped into comparable categories. A class is assumed to pass if even one of its components passes the test.

2.Threshold values - High and lower limits are segregated from the input. It is assumed that if those values pass the test, then every value in between will also pass.

3.Cause-effect diagramming - In both of the previous methods, just one input value is looked at a time. A cause-and-effect testing strategy performs systematic tests on all possible combinations of input values.

4.Pair-wise Testing- Multiple parameters influence software behavior. Multiple parameters are examined for their distinct values in pair wise testing.

5. State-based testing - When input is provided, the system's state changes. The states and input of these systems are used to test them.

4.2 WHITE-BOX TESTING:

It is utilized to test the software's functionality and how it is implemented in an effort to improve the structure or efficiency of the code. 'Structural' testing is another name for it. When employing this testing methodology, the individual performing the test has an understanding of the architecture and structure of the code. Programmers in charge of writing code perform this exam.

4.2.1. WHITE-BOX TESTING TECHNIQUES:

1. Control-flow testing - The objective of control-flow monitoring is to develop test cases that cover each and every circumstance involving a statement and a branch. The corporation's conditions are checked both as correct and incorrect information to make sure each of the claims are addressed.

2. Data-flow screening - This testing strategy is centered on monitoring all of the software's data variables. It looks at the places where variables were declared, defined, used, or changed.

4.3 VALIDATION AND VERIFICATION:

Validation of Software: This test process of validating software involves deciding whether or not it satisfies the user's requirements. At the end of the SDLC, it is finished. If the software complies with the requirements under which it was designed, it is verified.Validation confirms that the product being developed meets user requirements.

Verification of software: Confirmed is the method of confirming that software is built in compliance with the right standards and methodologies and that it satisfies business needs.Verification establishes whether the finished product satisfies the design requirements. The question of "Are we making this product while following strictly every one of the design specifications?" is answered via verification.The emphasis of verifications is on the design and system requirements.

Target of the test:

Errors- are genuine coding errors committed by developers. Furthermore, a disparity between the output generated by the program and the expected output is called an error.

Fault - A fault happens when there is a mistake. A fault, commonly referred to as a bug, is a mistake that can cause a system to fail.

Failure - is defined as the system's inability to fulfill the required job. When a problem exists in the system, failure happens.

4.4 FINDING THE VEHICLE AND ITS SPEED, ALONG WITH AN IMAGE:

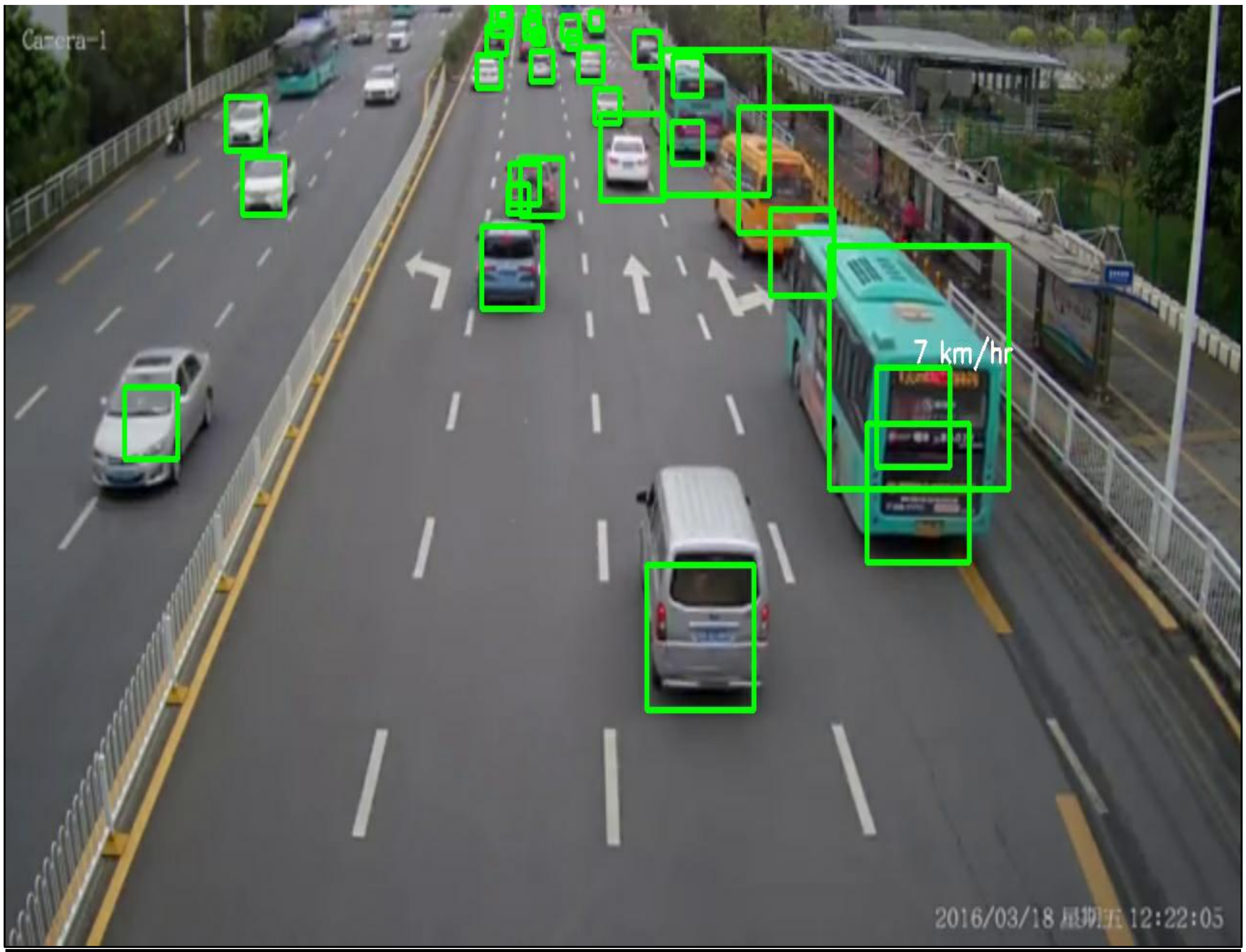

Figure 5: Vehicles and its speed

From figure 5 -Tracking is used to determine the vehicle's speed, and a picture sequence is taken. Speed estimation using image processing is more effective and cost-effective than using traditional radar technology and makes use of its many advantages.

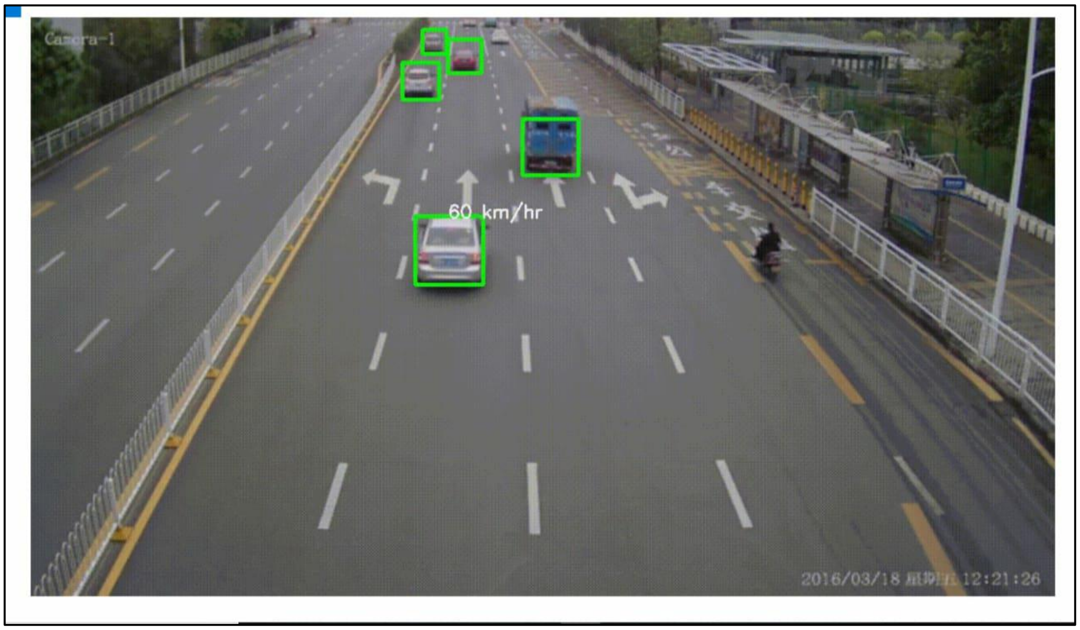

Figure 6: Finding Speed Vehicle

From Figure 6: Distance measurements are made using Euclidean distance and vehicle speed estimate using camera data. The removal of a video's static scene or backdrop is referred to as "backdrop subtraction." Since the camera is fixed, a number is assigned to each and every pixel in the image that is virtually constant over time. The underlying value associated with every picture point is calculated by deducting backgrounds.

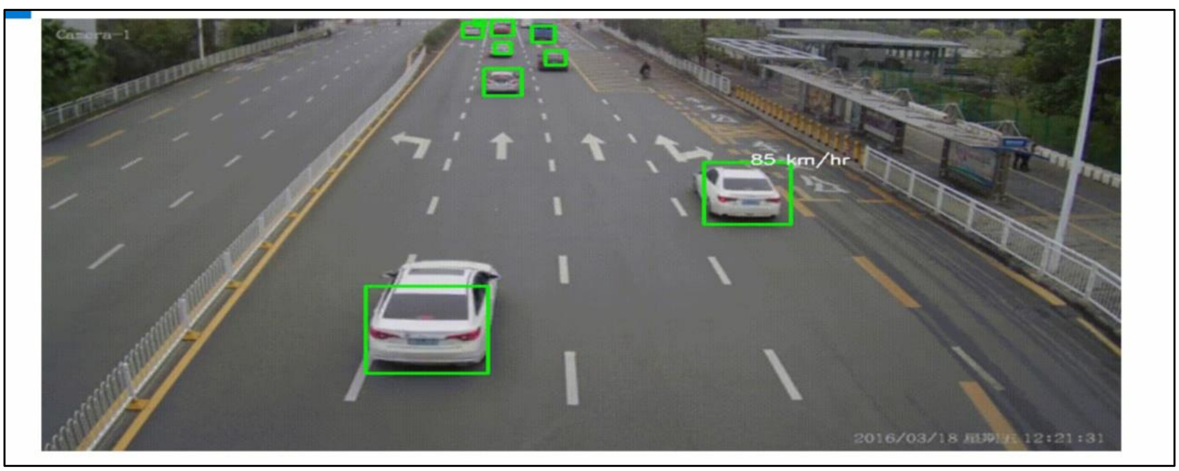

Figure 7: Speed vehicle Image Capture

From figure 7: Overcomes the limitations of the other methodologies, the Gaussian Mixture approach is used to detect vehicle speed and track the vehicle.

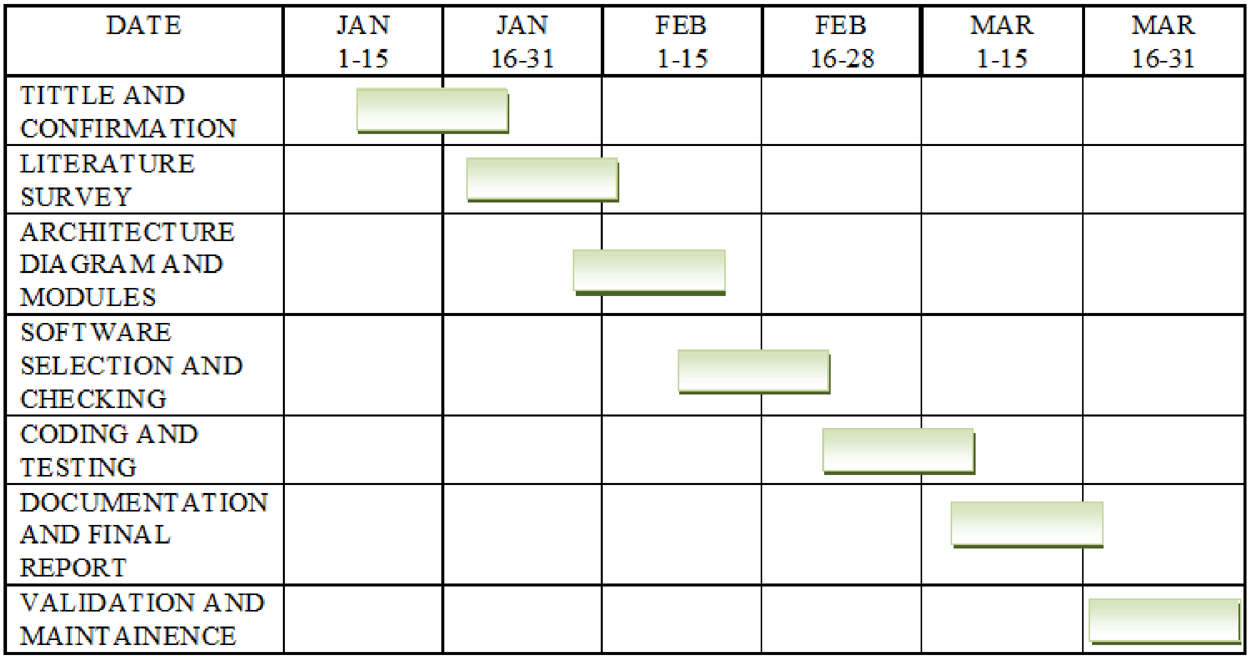

Figure 8: Gantt chart for proposed work

4.5 CONCLUSION:

Image processing-based speed estimate is more efficient and cost-effective than traditional radar technology, and it takes advantage of its extensive capabilities.Making effective sensors out of inexpensive cameras is a worthwhile project. This work uses rapidly evolving visual analysis as well as advanced learning techniques to attempt to extract car speeds from monitoring video data. The next phase is to include more precise visual motion data while calculating speed.This software project will be expanded further in the future, with the assistance of suitable algorithms for future advancements.Additional applications for determining vehicle speed can also be created. Image processing-based speed estimate is more efficient and cost-effective than traditional radar technology, and it takes advantage of its extensive capabilities.Why? Because we used the Mat laboratory platform to program, high-quality movies took a lot longer to process than regular video, making the system slow. So, for this project, standard-sized videos are advised. The next stage is to include more precise visual motion data to estimate speed. With the help of suitable algorithms, the software being developed will eventually be improved and enlarged. It is also possible to develop additional applications for measuring vehicle speed.

REFERENCES:

1. https://www.researchgate.net/publication/322514557_Development_of_automated_technique_for_vehicle_speed_estimation_and_tracking_in_video_stream
2. https://www.researchgate.net/publication/325768038_Traffic_Speed_Estimation_from_Surveillance_Video_Data
3. https://www.researchgate.net/publication/313542492_Research_of_vehicle_speed_detection_algorithm_in_video_surveillance
4. https://www.researchgate.net/publication/322511911_Development_of_prototype_for_vehicle_speed_measurement
5. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4883290/
6. Mahmood, Z.; Haneef, O.; Muhammad, N.; Khattak, S. Towards a Fully Automated Car Parking System. IET Intell. Transp. Syst. 2018, 13, 293–302.
7. Xiaohong, H.; Chang, J.; Wang, K. Real-time object detection based on YOLO-v2 for tiny vehicle object. ProcediaComput. Sci. 2021, 183, 61–72.
8. Rani, E. LittleYOLO-SPP: A delicate real-time vehicle detection algorithm. Optik 2021, 225, 165818. [CrossRef]
9. Tajar, T.; Ramazani, A.; Mansoorizadeh, M. A lightweight Tiny-YOLOv3 vehicle detection approach. J. Real-Time Image Process. 2021, 18, 2389–2401. [CrossRef]
10. Mahmood, Z.; Bibi, N.; Usman, M.; Khan, U.; Muhammad, N. Mobile Cloud based Framework for Sports Applications. Multidimens. Syst. Signal Process. 2019, 30, 1991–2019. [CrossRef]
11. Hamsa, S.; Panthakkan, A.; Al Mansoori, S.; Alahamed, H. Automatic Vehicle Detection from Aerial Images using Cascaded Support Vector Machine and Gaussian Mixture Model. In Proceedings of the 2018 International Conference on Signal Processing and Information Security (ICSPIS), Dubai, United Arab Emirates, 7–8 November 2018; pp. 1–4.
12. Mikaty, M.; Stathaki, T. Detection of Cars in HighResolution Aerial Images of Complex Urban Environments. IEEE Trans. Geosci. Remote Sens. 2017, 55, 5913–5924.
13. Arı, Ç.;Aksoy, S. Detection of Compound Structures Using a Gaussian Mixture Model With Spectral and Spatial Constraints. IEEE Trans. Geosci. Remote Sens. 2014, 52, 6627–6638. [CrossRef]
14. Hbaieb, A.; Rezgui, J.; Chaari, L. Pedestrian Detection for Autonomous Driving within Cooperative Communication System. In Proceedings of the 2019 IEEE Wireless Communications and Networking Conference (WCNC), Marrakesh, Morocco, 15–18 April

2019; pp. 1–6.

15. Xiong, L.; Yue, W.; Xu, Q.; Zhu, Z.; Chen, Z. High Speed Front-Vehicle Detection Based on Video Multi-feature Fusion. In Proceedings of the 2020 IEEE 10th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 17–19 July 2020; pp. 348–351.
16. Yawen, T.; Jinxu, G. Research on Vehicle Detection Technology Based on SIFT Feature. In Proceedings of the 8th International Conf on Electronics Info. and Emergency Communication (ICEIEC), Beijing, China, 15–17 June 2018; pp. 274–278.
17. Li, Y.; Wang, H.; Dang, L.M.; Nguyen, T.N.; Han, D.; Lee, A.; Jang, I.; Moon, H. A Deep Learning-Based Hybrid Framework for Object Detection and Recognition in Autonomous Driving. IEEE Access 2020, 8, 194228–194239. [CrossRef]
18. Li, Y.; Li, S.; Du, H.; Chen, L.; Zhang, D.; Li, Y. YOLO-ACN: Focusing on small target and occluded object detection. IEEE Access 2020, 8, 227288–227303. [CrossRef]
19. Cai, Z.; Vasconcelos, N. Cascade R-CNN: Delving into High Quality Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6154–6162.
20. Wang, C.; Wang, H.; Yu, F.; Xia, W. A High-Precision Fast Smoky Vehicle Detection Method Based on Improved Yolov5 Network. In Proceedings of the 2021 IEEE International Conference on Artificial Intelligence and Industrial Design (AIID), Guangzhou, China, 28–30 May 2021; pp. 255–259. [CrossRef]
21. Miao, Y.; Liu, F.; Hou, T.; Liu, L.; Liu, Y. A Nighttime Vehicle Detection Method Based on YOLO v3. In Proceedings of the 2020 Chinese Automation Congress (CAC), Shanghai, China, 6–8 November 2020; pp. 6617–6621. [CrossRef]
22. Sarda, A.; Dixit, S.; Bhan, A. Object Detection for Autonomous Driving using YOLO [You Only Look Once] algorithm. In Proceedings of the 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), Tirunelveli, India, 4–6 February 2021; pp. 1370–1374. [CrossRef]
23. Zhao, S.; You, F. Vehicle Detection Based on Improved Yolov3 Algorithm. In Proceedings of the 2020 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS), Vientiane, Laos, 11–12 January 2020; pp. 76–79. [CrossRef]
24. Corovi´c, A.; Ili´c, V.; Ðuri´c, S.; Marijan, M.; Pavkovi´c, B. The Real-Time Detection of Traffic Participants Using YOLO Algorithm. In ´ Proceedings of the 2018 26th Telecommunications Forum (TELFOR), Belgrade, Serbia, 20–21 November 2018; pp. 1–4. [CrossRef]

25. Lou, L.; Zhang, Q.; Liu, C.; Sheng, M.; Zheng, Y.; Liu, X. Vehicles Detection of Traffic Flow Video Using Deep Learning. In Proceedings of the 2019 IEEE 8th Data Driven Control and Learning Systems Conference (DDCLS), Dali, China, 24–27 May 2019; pp. 1012–1017. [CrossRef]