e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
( Peer-Reviewed, Open Access, Fully Refereed International Journal )
Volume:06/Issue:04/April-2024 Impact Factor- 7.868 www.irjmets.com

# NEXT-GENERATION TRAFFIC SURVEILLANCE: VEHICLE DETECTION AND COUNTING WITH NEURAL NETWORKS

**Shaik Munawar*1, Vinod Kumar Bhukya*2, Bhavana Bommana*3**

*1Assistant Professor, Department of Computer Science and Technology, Kakatiya Institute of Technology and Science, Warangal, India

*2,3Department of Computer Science & Engineering, Kakatiya Institute of Technology and Science, Warangal, India

*DOI : https://www.doi.org/10.56726/IRJMETS55232

## ABSTRACT

Effective highway management systems require intelligent vehicle recognition and counting; however, there are still issues since different vehicle sizes affect detection accuracy. This study describes a vision-based system for vehicle recognition and counting that uses input video streams to create YOLOv8 with lessons from YOLOv7.A unique segmentation approach is presented to extract the highway road surface from video frames, addressing the problem of different vehicle sizes. By concentrating on pertinent regions, this segmentation improves detection accuracy. The separated areas are then processed with YOLOv8, which incorporates improvements from YOLOv7, to efficiently identify the kind and location of vehicles.

Post-processing techniques and a maximum suppression algorithm are used to modify detection findings in real-time, improving precision, especially in situations where there are overlapping or tiny cars. Additionally, vehicle trajectories are obtained from identified vehicles, allowing for the identification of various vehicle kinds and the estimation of driving directions. To confirm the efficacy of the suggested methodologies, a variety of highway surveillance movies are used for experimental assessments. The results show noteworthy progress in determining driving directions and counting cars, as well as considerable increases in the accuracy of detecting tiny vehicle objects.

**Keywords:** vehicle counting,yolo, post-processing step, non-maximum suppression, simple deep convolution.

## I. INTRODUCTION

Counting and detecting vehicles has become a critical problem in several fields, including traffic management, surveillance, and autonomous driving. Typical techniques frequently have issues with speed, accuracy, and vehicle type recognition. Advanced algorithms like "You Only Look Once" (YOLO) have become more popular as a solution to these problems.

YOLO achieved great accuracy and real-time performance, revolutionizing object identification. In contrast to conventional techniques, which require many processing steps, YOLO effectively makes predictions by processing the complete image through a neural network in a single pass.

The basic idea behind YOLO is to anticipate bounding boxes and class probabilities for each grid cell in the input picture by first splitting it into a grid. Vehicle localization and categorization may be done precisely with this grid-based method.

In this project, we aim to implement a neural network-based vehicle detection and counting system using YOLOv8, which builds upon the strengths of previous versions while integrating novel features and optimizations. By leveraging the efficiency and accuracy of YOLOv8, we endeavor to address the challenges associated with vehicle detection and provide a robust solution for real-world applications. Subsequent iterations of YOLO, such as YOLOv2, YOLOv3, YOLOv4, and YOLOv5, have been introduced in order to improve vehicle detection capabilities and overcome limitations in earlier iterations.

Accurate vehicle identification and counting is essential to many applications in contemporary transportation systems, from intelligent transportation systems to traffic monitoring. The accuracy and effectiveness of traditional vehicle identification techniques are frequently compromised, especially in difficult situations like congested traffic areas or low light levels. Neural network-based techniques, on the other hand, have completely changed this sector by providing improved capabilities for reliable and fast vehicle recognition and counting.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
( Peer-Reviewed, Open Access, Fully Refereed International Journal )
Volume:06/Issue:04/April-2024 Impact Factor- 7.868 www.irjmets.com

## II. METHODOLOGY

A breakthrough in object identification, the YOLO (You Only Look Once) algorithm provides a simplified method for real-time detection jobs. In order to accomplish this efficiency, YOLO divides the input picture into a grid and, in a single pass, predicts the bounding boxes and class probabilities for items inside each grid cell. Because multi-stage processing is no longer necessary thanks to its single-stage design, YOLO can achieve impressively quick inference times without sacrificing accuracy. Bounding box predictions provide accurate localization and confidence estimates for identified objects by include crucial parameters including width, height, and a confidence score. To further enhance the detections, YOLO employs non-maximum suppression, guaranteeing that the output includes only the most certain and non-overlapping bounding boxes.

The YOLO algorithm has made great progress in object recognition by combining accuracy and efficiency. Because of its single-pass processing and grid-based methodology, it is especially well suited for situations where quick object identification and categorization in pictures and video streams is necessary. YOLO has been widely adopted in several sectors because to its simplicity and efficacy. Accurate and rapid object detection is crucial for improving automation, safety, and security. The YOLO algorithm is positioned to stay at the forefront of object detection technology as researchers continue to improve and optimize it. This will lead to advancements in industries including industrial automation, smart surveillance, and autonomous navigation.

### 1.Vehicle detection

First, a wide range of traffic scene datasets with labels derived from ground truth are gathered, covering a variety of environmental circumstances. The dataset is then preprocessed to improve its quality and standardize it, which helps the model perform better during training. A detection model that balances accuracy, speed, and computing efficiency is essential.

Examples of such models are SSD (Single Shot MultiBox Detector), Faster R-CNN (Region-based Convolutional Neural Network), and YOLO (You Only Look Once). The model is trained using optimization techniques such as stochastic gradient descent once the dataset has been separated into training, validation, and test sets. Its parameters are then refined based on annotated labels. Using the validation set, hyperparameter tweaking adjusts the model's parameters to enhance performance metrics. Evaluation measures evaluate the model's performance, including mean average precision, recall, and accuracy.

Post-processing methods, such as non-maximum suppression, are used to remove duplicates and improve identified vehicle bounding boxes. Lastly, performance analysis verifies the model's dependability and efficacy in realistic deployment settings by closely examining its precision, processing speed, and resilience across a range of real-world circumstances. This thorough process guarantees the creation of a reliable vehicle recognition system that can recognize cars in a variety of traffic situations.

### 2.Vehicle counting

Vehicle counting is a methodical process that aims to precisely count the number of cars in a specific traffic situation. First, a large-scale collection of pictures or videos of various traffic scenarios is gathered and annotated such that each car is given a unique label. To improve the efficiency of further processing stages, the dataset is then preprocessed to standardize its format, resolution, and quality. Vehicle identification and localization inside each frame or picture of the dataset is accomplished by using vehicle detection algorithms such as YOLO, SSD, or Faster R-CNN. After a vehicle is spotted, a tracking algorithm links the detected cars in a series of frames to follow their movements over time.

An algorithm for counting vehicles is created using the tracked vehicles to determine how many vehicles are present in predetermined areas of interest. In order to verify accuracy and dependability, the counting method is validated by contrasting the counted automobiles with ground-truth annotations. Based on validation findings, the algorithm parameters are modified to improve performance; this includes modifying threshold values and honing tracking strategies.

In order to assure accurate and dependable vehicle counting under a range of traffic situations, the optimized counting algorithm is then implemented in real-world traffic monitoring systems and continuously evaluated for performance and improvement. This thorough approach makes it possible to create reliable vehicle counting systems that can precisely count vehicles in traffic situations in real time.

IRJMETS

e-ISSN: 2582-5208

International Research Journal of Modernization in Engineering Technology and Science

( Peer-Reviewed, Open Access, Fully Refereed International Journal )

Volume:06/Issue:04/April-2024 Impact Factor- 7.868 www.irjmets.com

## III. MODELING AND ANALYSIS

**Gathering of Data:**

- Compile a large collection of pictures or videos that show different car situations.
- Make sure there is variety in the kinds of vehicles, the surroundings, and the perspectives.
- Put bounding boxes around each vehicle instance and name them appropriately to annotate the dataset

**Preparing Data:**

- Resize pictures to a standard scale that the neural network can accept as input.
- Adjust pixel values to a standard scale.
- Add more features to the dataset, such flipping, rotating, and brightness adjustments, to boost diversity.

**Model Selection:**

- Select the right neural network architecture for counting and detecting vehicles.
- Think about architectures such as SSD (Single Shot MultiBox Detector), YOLO (You Only Look Once), or Faster R-CNN.
- Strike a balance between computational resources, speed, and precision.

**Model Training**:

- Make training, validation, and test sets out of the dataset.
- Utilizing the training data, train the chosen model.
- If pre-trained models are available, use methods like transfer learning to achieve faster convergence and better generalization.
- Adjust the hyperparameters of the model (learning rate, batch size, etc.) in light of its performance on the validation set.
- Track training progress with indicators like accuracy and loss.

**Evaluation:**

- Track training progress with indicators like accuracy and loss.
- For object detection tasks, measure metrics include mean average precision (mAP), recall, precision, and F1-score.
- Examine the model's performance in relation to various car classes and environmental factors.

. **Reprocessing:**

- Apply post-processing techniques to count the number of cars and fine-tune the bounding boxes of the identified vehicles.
- Utilize non-maximum suppression (NMS) in order to eliminate redundant detections.
- Use tracking methods to ensure that counts in video data remain constant across several frames.

**Deployment:**

- Include the trained model in the intended system or application.
- Verify that the input sources (such as picture frames and video streams) are compatible.
- If relevant, optimize the model for real-time performance.
- Verify the deployed model's functionality by testing it in actual situations.

**Monitoring and Maintenance:**

- Maintain a close eye on the correctness and performance of the deployed model.
- Get user input and make adjustments based on observations made in the actual world.
- To keep the model accurate and relevant, update it with fresh data on a regular basis.

## IV. RESULTS AND DISCUSSION

We are giving this night time video as input for vehicle detection which we need to detect as shown in the below figure.

**Fig.1** Result for step1

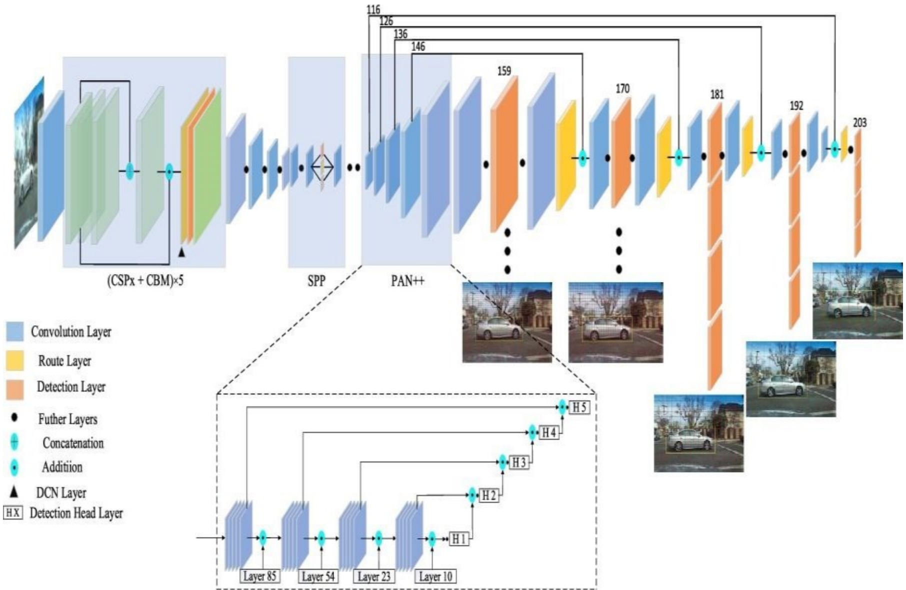


**Fig. 2** Result of step 2

**Table.1** Sample training data

| | image_id | xmin | ymin | xmax | ymax | x_center | y_center | w |
|---|---|---|---|---|---|---|---|---|
| 0 | vid_4_1000 | 281.259045 | 187.035071 | 327.727931 | 223.225547 | 0.450434 | 0.539817 | 0.068741 |
| 1 | vid_4_10000 | 15.163531 | 187.035071 | 120.329957 | 236.430180 | 0.100217 | 0.557191 | 0.155572 |
| 2 | vid_4_10040 | 239.192475 | 176.764800 | 361.968162 | 236.430180 | 0.444645 | 0.543678 | 0.181621 |
| 3 | vid_4_10020 | 496.483358 | 172.363256 | 630.020261 | 231.539575 | 0.833213 | 0.531451 | 0.197540 |
| 4 | vid_4_10060 | 16.630970 | 186.546010 | 132.558611 | 238.386422 | 0.110347 | 0.559122 | 0.171491 |

We can implement this project in many other fields like in detecting the number of vehicles at the traffic light point and show stop timer according to traffic density. These are the outputs of the project after detecting the vehicles during night time.

IRJMETS
e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
( Peer-Reviewed, Open Access, Fully Refereed International Journal )
Volume:06/Issue:04/April-2024 Impact Factor- 7.868 www.irjmets.com


**Fig. 3** Detecting of the vehicles

## V. CONCLUSION

In conclusion, employing the YOLOv8 neural network architecture for vehicle recognition and counting offers a viable way to improve traffic monitoring and management systems. By utilizing YOLOv8, which combines the benefits of efficiency, accuracy, and real-time processing, the system shows strong performance in identifying and counting cars in a range of traffic circumstances.

The system can successfully manage dynamic traffic conditions because of the excellent object identification accuracy and efficient real-time image processing of the YOLOv8 architecture. The system accomplishes efficient processing and accurate vehicle localization by utilizing YOLOv8's unified architecture for object identification, which helps to produce accurate counting results.

## VI. REFERENCES

[1] Kamkar, S.; Safabakhsh, R. Vehicle detection, counting and classification in various conditions. IET Intell. Transp. Syst. 2016, 10, 406–413. [Google Scholar] [CrossRef]

[2] Jianjun Hu , Yuqi Sun and Songsong Xiong (2021). " Research on the Cascade Vehicle

[3] Detection Method Based on CNN."

[4] Luiz G. Galvao, Maysam Abbod, Tatiana Kalganova, Vasile Palade and Md Nazmul Huda ,

[5] (2021). "A Real-Time Object Detector for Autonomous Vehicles Based on YOLOv4."

[6] Yang, H.; Qu, S. Real-time vehicle detection and counting in complex traffic scenes using background subtraction model with low-rank decomposition. IET Intell. Transp. Syst. 2017, 12, 75–85. [Google Scholar] [CrossRef]