International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Automatic Vehicle Detection and Tracking Strategy Using Deep Learning Model (YOLO v2 & R-CNN)

Priyanka Ankireddy[1*], S. Gopalakrishnan[2], V. Lokeswara Reddy[3]

Submitted: 17/01/2024 **Revised**: 25/02/2024 **Accepted**: 03/03/2024

Abstract*:* Vehicle detection and tracking plays a crucial role in various smart mobility networkss, including traffic monitoring, security surveillance, and autonomous vehicles. Image processing techniques offer a powerful tool for achieving these tasks by analyzing visual information captured from cameras. Since Deep Learning (DL) is developing so quickly, the computer vision community has demanded that excellent, reliable, and efficient services be developed across a range of domains. The objective of vehicle detection and tracking is to automatically detect and track the movement of vehicles in real-time video sequences. The paper presents novel vehicle detection and tracking system using image processing techniques. The approach consists of the key stages: An assortment of image processing techniques was employed in the production of this data. Utilizing the most recent iteration of the YOLO model, YOLO v2, as well as the R-CNN model and Fast-RCN, have all been employed for detection purposes. Following the identification and tracking processes, the number of vehicles and their estimated speeds are calculated. Moreover, the proposed approach using Fast-RCNN has the best performance with precision of 98.94% and recall of 99.12% for the CDnet 2014 dataset respectively..

Keywords: *Vehicle detection, Deep learning, Quicker R-CNN, YOLOv2*

1. Introduction

Enhancing image processing quality through deep learning-based vehicle detection and tracking addresses the limitations of traditional image processing techniques in accurately identifying and tracking vehicles in various video sequences. Convolutional neural networks (CNNs) and other forms of deep learning have totally changed the game when it comes to processing photos. They make it possible to extract intricate patterns and information from images, which has led to significant improvements in object detection and tracking tasks. The current state of the art in image-based vehicle object recognition is fragmented between more conventional machine vision approaches and more advanced deep learning techniques. Machine vision techniques have traditionally relied on tracking a it's movements to differentiate it from a static background image. In this method, there are three main submethods: background subtraction [2], continuous video frame difference [3], and optical flow [4].

Traditional image processing techniques often struggle to handle the complexities of real-world video sequences, resulting in inaccuracies and inefficiencies in vehicle detection and tracking.Vehicles exhibit significant variations in appearance because of things like varied car models, blockage, and brightness. Traditional image processing techniques often fail to generalize to these variations, leading to missed detections and false positives. Complex and cluttered backgrounds make it difficult to distinguish vehicles from other objects using traditional image processing methods. Background subtraction techniques, while useful, can be sensitive to noise and may not be effective in all scenarios. Vehicles can partially or fully occlude each other, making it challenging to maintain consistent identification and track individual vehicles using traditional methods. Occlusion handling is often limited, leading to tracking failures.

Unless correctly examined to derive meaningful knowledge, the image is meaningless. HOG (Historogram of Oriented Gradient) [5, Haar [6, and LBP [7] are examples of hand-crafted features that are among the most effective ways for recognizing automobiles. However, these features do not offer a universal answer, and classifiers need to be adjusted to suit in different settings. For vehicle detection, a shallow neural network is also used, yet its performance has not produced the required quality. To deal with this mountain of data, a new approach is required that can reliably and rapidly process large amounts of data efficiently.

Deep learning, particularly CNNs, has emerged as a powerful tool for overcoming the limitations of traditional image processing in vehicle detection and tracking. CNNs are able to learn complex patterns and features from

[1]*Department of Information Technology, Hindustan Institute of Technology and Science, Chennai, Tamilnadu-603103, India. Email: priyasivakrishna99@gmail.com**
[2]*Department of Information Technology, Hindustan Institute of Technology and Science, Chennai, Tamilnadu-603103, India. Email:drsgk85@gmail.Com*
[3]*Department of Computer science and Engineering, KSRM College of Engineering, Kadapa, Andhra Pradesh-516003, India. Email: vlreddy@ksrmce.ac.in*

images, enabling them to effectively distinguish vehicles from background clutter and handle variations in vehicle appearance. Additionally, deep learning models can be optimized for real-time performance, making them suitable for practical applications. The use of deep learning techniques like convolutional neural networks (DL), recurrent convolutional neural networks (RCNNs), and deep neural networks (DNNs) enhances the speed, accuracy, and robustness of algorithms that identify and categorize automobiles in still photos or moving video.Deep learning-based image processing improves vehicle detection and tracking. To overcome traditional image processing approaches, deep learning models are developed and implemented to recognize and track automobiles in video sequences.

In this paper CNN-based two-step algorithms have significantly advanced vehicle detection capabilities, contributing to the development of intelligent transportation systems and enhancing safety and security in various applications. Yhe two-step approach allows for modularity and flexibility in choosing different techniques for region proposal generation and classification.

2. Related Works and Background

In [8], the scientists employed an association method and similarity measurement for vehicle tracking, and deep learning for visual vehicle detection. They applied the YOLOV3 detection method to the experimental data and video detection. The vehicle video detection experiment is conducted within the TensorFlow framework, utilizing the YOLOV3 convolutional neural network structure. The detection accuracy and error detection rate are used to assess the detection outcomes. Firstly, a driving record segment is recorded on the roads surrounding the school, and the enhanced information set for video detection is established. Following that, the weightless network of YOLOV3 is trained using the data collection. Lastly, a comparison is made of the test outcomes between the supplemented data set and the KITTI data set. Subsequently, the network is employed to contrast the detection outcomes of several detection strategies. After obtaining the object selection frame following video recognition, the second task is to construct the vehicle tracking model. The vehicle appearance characteristics are extracted using the classic hog feature extraction approach, and the motion similarity is then computed. The target trajectory management algorithm recovers or releases the target trajectory to produce the tracking effect once The total resemblance between the targets in the front and rear photos has been ascertained.

In [9], DeepTrackNet is a single end-to-end deep learning system for autonomous vehicle visibility, localization, and tracking. The DeepTrackNet architecture that has been suggested uses a deep regression network for vehicle tracking and a mobilenet SSD for vehicle detection and localization. Based on the conducted experimental study, the paper concludes that the suggested DeepTrackNet design has produced adequate outcomes for actual time vehicle tracking and identification on an Nvidia Jetson integrated computer framework utilizing a single- lens camera with an appropriate delay. On the VTB TB-100 dataset, deep regression tracker has the lowest overall average failure frames (4.388), while mobile net SSD has a median detection time of 84 (ms).

In [10], For car and pedestrian recognition, the authors assessed four contemporary deep-learning object-detection strategies separately and empirically: the quicker F-RCN, SSD, HoG, and YOLOv7. They also looked into the classification of circumstances of the weather by utilizing a real-time dataset (DAWN). The proposed method is superior to the technique that is currently thought to be the latest and greatest for vehicle identification and tracking in unfriendly and hazardous environments, as shown by the results of the tests.

In [11], Multiple target identification and categorization for clever vehicles was demonstrated. DL forms the basis of this approach. Complex traffic conditions and diverse demand make multi-target identification and classification in traffic road scenes difficult. Every detection and recognition task is finished by selecting an appropriate deep learning method. For road multi-target identification, YOLO v5 algorithm is used. Target monitoring of YOLOv5 detection findings is achieved using Deep SORT method. MTCNN is used for license plate recognition. Target identification and input video tracking are ensured by the implemented system. Presenting the experiment data shows that target detection and tracking are effective and accurate.

In [12], DeepSORT, an elementary online and instantaneous fashion multitarget surveillance system with an extensive relationship metric, was employed to investigate automobile tracking. A YOLOv5s_DSC vehicle identification algorithm based on the YOLOv5s method contributes precise and rapid vehicle identification data to the DeepSORT algorithm, as a result of the strategy's heavy reliance on finding targets. A 1% discrepancy exists between YOLOv5s_DSC and ideal mAP0.5 (mean average precision), recall rate, and accuracy rate. It compared to YOLOv5s, has 23.5% fewer parameters, 32.3% less computation, 20% less weight file size, and 18.8% more image processing speed. After adding DeepSORT, YOLOv5s_DSC + DeepSORT can process 25 FPS and is more occlusion-resistant.

3. Materials and Methods

Figure 1 shows the proposed vehicle detection and tracking system design. We begin by entering the camera data from the accident scene. The area of the road's surface is then removed and divided. The deep learning object detection algorithm detects highway traffic vehicles. Feature extraction is then applied to the identified vehicle box to finalize multi-object tracking and collect data on vehicle traffic.

(i) Region Proposal Generation: This stage aims to identify potential regions in the image that likely contain vehicles. These regions are called region proposals or candidate boxes. Various techniques can be employed for region proposal generation, such as selective search or region-based convolutional networks (R-CNNs).

(ii) Classification and Bounding Box Regression: For each region proposal, this stage classifies whether the region contains a vehicle or not. Additionally, it refines the bounding box coordinates to accurately localize the vehicle within the region. This stage typically involves a CNN-based classifier, like Quick R-CNN or Quicker R-CNN.

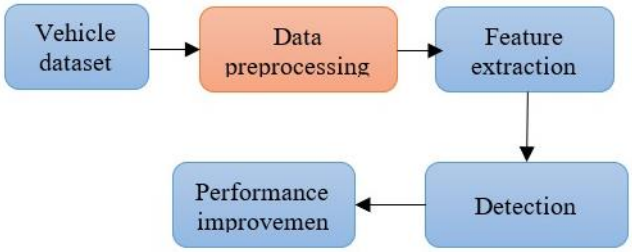

Fig. 1. The architecture design of vehicle detection and tracking

(iii) Dataset description

The CDnet 2014 dataset is a popular benchmark for evaluating change detection algorithms. It consists Contains 22 movies, with over 70,000 frames that have been annotated pixel-wise, encompassing five new categories that incorporate issues that are experienced in a variety of surveillance contexts. This collection contains videos shot in difficult weather, with low frame rates, at night, using PTZ capture, and during periods of air turbulence. The dataset includes a large number of videos with a wide variety of scenes and conditions.All the videos are annotated with pixel-wise ground truth labels, which allows for accurate evaluation of change detection algorithms.

(iv)Data Preprocessing:

Each frame of the video is read and converted to a usable format for processing and converting the image to grayscale might be beneficial. Apply filters to remove noise and improve image quality and remove the static background from the image to focus on moving objects.

(v)Detection

Employ an Use an object identification technique, such as Faster R-CNN, YOLO, or RNN, to find the cars in the frame. Draw bounding boxes around the detected vehicles in the frame.Extract features from the detected objects, like size, color, texture, etc

(vi) Regions with CNN Features, or R-CNN, is a family of object detection algorithms that utilize a deep convolutional neural network (CNN) for both region proposal generation and object classification and bounding box refinement. This is a two-step object detection technique that produces a list of possible object areas first, then classifies and fine-tunes each region's bounding box. The RPN effectively identifies potential vehicle locations in the image, reducing the search space and focusing on areas of interest.The CNN-based feature extraction and classification stages enable the algorithm to distinguish vehicles from other objects and refine their bounding boxes with precision.

(vii) YOLO (You Only Look Once): This series of object detection techniques predicts object classes and bounding boxes straight from an input image by using a deep convolutional neural network (CNN). In contrast to the two-stage R-CNN algorithm, YOLO is a one-stage algorithm, making it significantly faster and more efficient.YOLOv2 incorporated high-resolution feature maps from earlier layers, enabling more precise object localization, YOLOv2 replaces direct bounding box prediction with convolutional predictions over multiple anchor boxes, enhancing the flexibility of the model.

Fast R-CNN: Using CNN Features, Fast R-CNN is a cutting-edge object detection technique if that has demonstrated remarkable performance in vehicle detection tasks. Its two-stage architecture, combining region proposal generation and object classification, allows for accurate and efficient vehicle localization and classification.

Step 1: The CNN that makes up the Fast R-CNN is often pre-trained on the ImageNet classification task. CNN's final pooling layer is now a "ROI pooling" layer, with its last FC layer is thus a (K + 1) grouping soft max layer branch and a class-specific bounding box regression branch.

Step 2: By feeding the full image into the backbone CNN, we can extract the features from the final convolution layer. Given CNN's foundation feature maps are considerably less than the initially captured photo. This hinges on the speed of the underlying CNN, which in a VGG backing context is usually 16.

Step 3: A region proposal method like selective search generates object proposal windows. According to Regions

with CNNs, rectangular regions on an image can be used to indicate the existence of an item. These regions are called object suggestions.

Step 4: The ROI Pool layer then collected chunk of the spine feature map that is associated with it. The single pyramid level spatial pyramid pooling (SPP) layer is a specific case of the ROI pool layer. In essence, the layer creates sub-windows of size h/H by w/W from the features from the chosen proposal windows (which originate from the region proposal method) and pools the features in each of these sub-windows. All input sizes produce fixed-size output features of size (H x W). The result works with the network's first fully-connected layer because H and W were chosen. In the Quick R-CNN study, seven are the selected values for H and W. ROI pooling is done individually in each channel, just like conventional pooling.

Step 5: The ROI Pooling layer's N x 7 x 7 x 512 output features (where N is the number of proposals) are fed to the softmax, BB-regression, and FC layers. This part of the softmax classification tree gives you the chance values for each ROI that belongs to K categories and one "catch-all" background group. By utilizing the result of the BB regression branch, the bounding boxes of the region proposal method are improved.

4. Results and Discussions

The experiment is simulated using Matlab and Python. The discussion will center on quantitative evaluation of detection and tracking, drawing comparisons to the state-of-the-art technique. In order to judge how accurate the recognition is, two numbers called Precision and Recall are used. These numbers have been used before and are considered standard. If you divide the number of correctly identified car pixels by the total number of correctly identified object pixels, which includes both true positives and false positives, you get the precision.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Recall is the percentage of accurately recognized vehicle pixels in the dataset, including false negative pixels (FN).

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

To compare the background subtraction-based proposed method, experiment 1 retrains the network with 127 highway baseline scene frames from ChangeDetection.net dataset (CDnet 2014). Fig. 2 shows samples of these pictures.

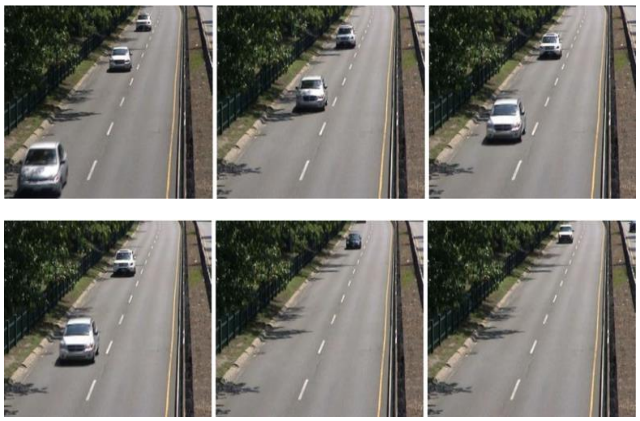

Fig. 2. Example CD-net2014 highway training photos for fine-tuning.

In experiment 2, train the network. The examples of pictures are shown in Figure 3. In particular, keep in mind that test picture frames differ from training picture frames. For the second experiment using the matlab images, we only utilized a small fraction of the frames from several sequences, including the tram stop, the street corner at night, and the intermittent pan. When testing, the remaining frames are supplemented by around 90% of these sequences.Most of these sequences are used for testing purposes, with the remaining 10% serving as mere frames.

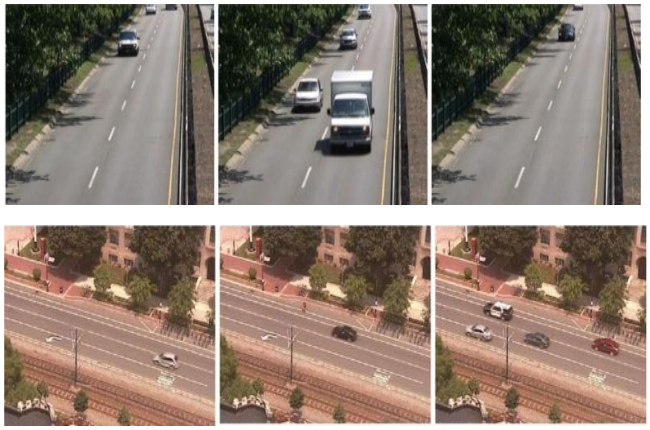

Fig. 3: The fine-tuning process made use of sample training images retrieved from matlab and sequences in the CD-net2014 dataset.

Table 1: Results comparing training time and computation time of three models

Model	Training time (sec)	Computation time (sec)
R-CNN	93.51	73.21
YOLOv2	91.24	63.54
Fast R-CNN	62.31	35.47

Figure 4 shows the Fast R-CNN having less training time of 92.31 sec and computation time of 35.47 sec compared to R-CNN and YOLOv2 models.

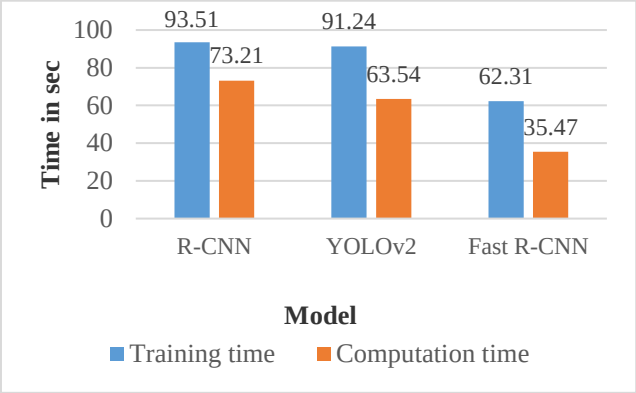

Fig. 4. Performance comparison of training and computation time for three models

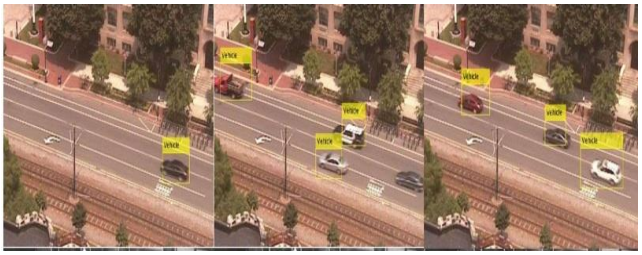

Fig. 5. Sample detection results using experiment 2. with a beautiful day, shifting tree views, and a camera angle.

Table 2: Results comparing Precision and Recall of three models

Model	Precision in %	Recall in %
R-CNN	96.88	97.54
YOLOv2	97.25	98.21
Fast R-CNN	98.94	99.12

Figure 6 shows the Fast R-CNN having highest precision of 96.88% and recall of 99.12% as compared to R-CNN and YOLOv2 models.

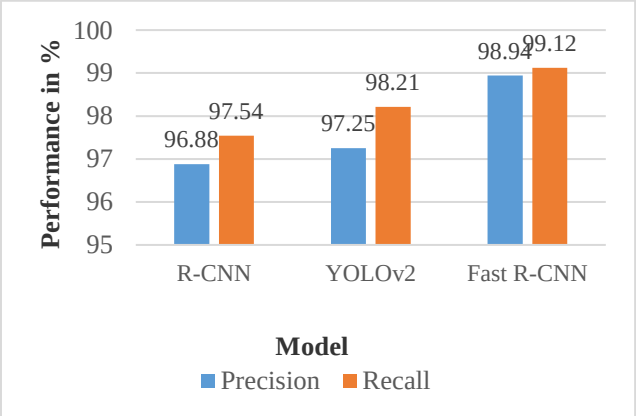

Fig. 6. Performance comparison of training and computation time for three models

5. Conclusion

Convolution Neural Networks (CNNs) and In this work, trackers were effectively employed to detect and monitor vehicles. The following classifier detectors were employed: R-CNN, YOLOv2, and Fast-RCNN. We found that the suggested rapid R-CNN outperformed both the R-CNN based methods of identification and background elimination. A variety of vehicle sceneries should be included in the training photos for accurate recognition and tracking, as shown in both experiments. Furthermore, it has been reported that the dataset yielded a 99.12% recall, a 98.94% precision, a training time of 62.31 seconds, and a calculation time of 35.47 seconds. This work attempts to provide a useful tool for intelligent transportation systems and other applications by improving tracking and identification of vehicles using image processing.

References

[1] 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.

[2] Radhakrishnan, M. (2013). Video object extraction by using background subtraction techniques for sports applications. Digital Image Processing, 5(9), 91–97.

[3] Qiu-Lin, L.I., & Jia-Feng, H.E. (2011). Vehicles detection based on three-frame-difference method and cross-entropy threshold method. Computer Engineering, 37(4), 172–174.

[4] Liu, Y., Yao, L., Shi, Q., Ding, J. (2014). Optical flow based urban road vehicle tracking, In 2013 Ninth International Conference on Computational Intelligence and Security. https://doi.org/10.1109/cis.2013.89: IEEE

[5] Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20– 25 June 2005; Volume 1, pp. 886–893.

[6] Mita, T.; Kaneko, T.; Hori, O. Joint haar-like features for face detection. In Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05), Beijing, China, 17–21 October 2005; Volume 2, pp. 1619–1626.

[7] Zhang, G.; Huang, X.; Li, S.Z.; Wang, Y.; Wu, X. Boosting local binary pattern (LBP)-based face recognition. In Proceedings of the Chinese Conference on Biometric Recognition, Guangzhou,

China, 13–14 December 2004; Springer: Berlin/Heidelberg, Germany, 2004; pp. 179–186.

[8] Zhang, Yaoming & Song, Xiaoli & Wang, Mengen & Guan, Tian & Liu, Jiawei & Wang, Zhaojian & Zhen, Yajing & Zhang, Dongsheng & Gu, Xiaoyi. (2020). Research on visual vehicle detection and tracking based on deep learning. IOP Conference Series: Materials Science and Engineering. 892. 012051. 10.1088/1757-899X/892/1/ 012051.

[9] Amara, Dinesh & Karthika, R. & Soman, K.. (2020). DeepTrackNet: Camera Based End to End Deep Learning Framework for Real Time Detection, Localization and Tracking for Autonomous Vehicles. 10.1007/978-3-030-30465-2_34.

[10] Zaman, Mostafa & Saha, Sujay & Zohrabi, Nasibeh & Abdelwahed, Sherif. (2023). Deep Learning Approaches for Vehicle and Pedestrian Detection in Adverse Weather. 10.1109/ITEC55900.2023. 10187020.

[11] Gao, Hongbo & Su, Huiping & He, Xi & liao, yanzhen & Wu, Yulin & Juping, Zhu & Zhang, Fei. (2023). Multi-target Detection and Classification for Intelligent Vehicle Based on Deep Learning. 10.1007/978-981-99-2789-0_29.

[12] Lin, Lixiong & He, Hongqin & Xu, Zhiping & Wu, Dongjie. (2023). Realtime Vehicle Tracking Method Based on YOLOv5 + DeepSORT. Computational Intelligence and Neuroscience. 2023. 10.1155/2023/7974201.