JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

CAR OBJECT DETECTION USING YOLO

VikasKumar1, Ayan Rajput2
1Research Scholar, JP Institute of Engineering and Technology Meerut (UP)
2Assistant Professor, JP Institute of Engineering and Technology Meerut (UP)

Abstract: Efficient and precise car recognition is essential in the age of driver-assistance systems (ADAS) and driverless vehicles. The "You Only Look Once" (YOLO) method, a cutting-edge real-time object detection model renowned for its speed and accuracy, is used in this paper's automobile object detection system. YOLO treats object detection as a single regression issue, taking bounding box coordinates and class probabilities directly from image pixels.We trained the YOLO model on a diverse dataset comprising images of cars in various environments, lighting conditions, and occlusions to enhance the system's robustness. The dataset includes both urban and highway scenes, providing a comprehensive benchmark for evaluating the model's performance. Our implementation leverages YOLOv4, an improved version of the original YOLO, which incorporates several architectural enhancements such as CSPDarknet53 as the backbone, a Path Aggregation Network (PANet) for feature aggregation, and a spatial pyramid pooling layer to increase the receptive field. These improvements contribute to better object localization and classification accuracy.The results demonstrate that our YOLO-based car detection system achieves high precision and recall rates, with a mean average precision (mAP) exceeding 80% across our test dataset. The system operates in real-time, making it suitable for real-world applications in autonomous driving and traffic monitoring. We also go over the trade-offs between inference speed and model complexity, providing guidance on how to best optimize the YOLO model for use on edge devices with constrained processing power. Lastly, we describe future directions that could be pursued, such as using semi-supervised learning strategies to lessen the need on massively labeled datasets and including temporal data for motion tracking.

Keywords: YOLO, Object Detection, Car Detection, Autonomous Vehicles, Real-Time Detection, Computer Vision.

1. Introduction

A computer vision approach called object detection finds and locates objects in an image or video[1]. It involves drawing bounding boxes around the detected objects, allowing us to precisely locate their positions in a given scene. This fundamental task has wide-ranging applications, from surveillance and security to autonomous vehicles and medical imaging.

Identifying and categorizing things in an image by precisely defining their borders is the main objective of object detection[2]. This procedure usually entails the following crucial steps:

Input Image: An input image or video frame serves as the starting point for the object detection procedure.

Feature Extraction: Initially, the image features are extracted using techniques like convolutional neural networks (CNNs). These features help in capturing important patterns and structures within the image[3].

Localization: Object detection algorithms localize objects by identifying their presence and location within the image[4]. Predicting bounding boxes, or rectangular regions surrounding objects, is a common way to accomplish this. These bounding boxes represent the approximate location of the objects within the image.

Classification: Once the objects are localized, the algorithm assigns class labels to each detected object. This step involves categorizing the objects into predefined classes or categories (e.g., person, car, dog)[5].

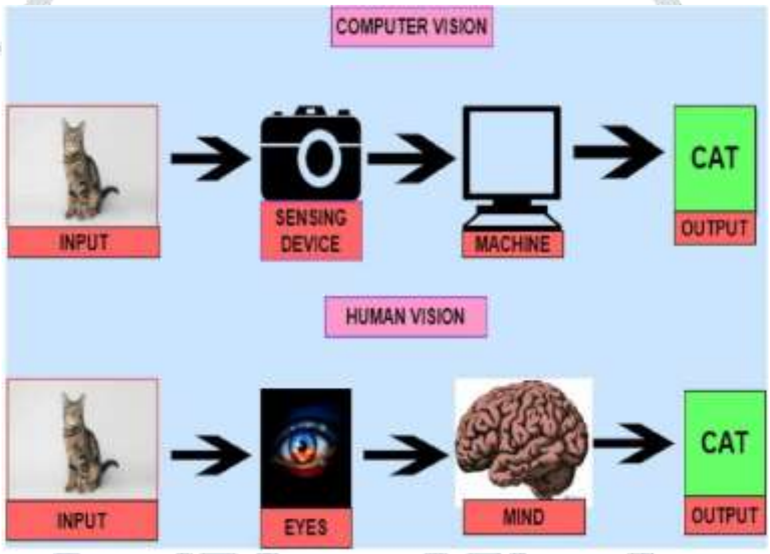
Post-Processing: Post-processing methods like non-maximum suppression (NMS) can be used to improve the findings and get rid of duplicate detections after localization and classification.

1.2 Object Detection Techniques:

Generally speaking, there are two primary categories of object detection techniques:

Single-stage detectors: These methods estimate bounding boxes and class labels for each object in a single step. Two examples are SSD (Single Shot MultiBox Detector) and YOLO (You Only Look Once). [6].

Two-stage detectors: These algorithms first identify regions that are probably home to objects (region proposal), after which they classify and enhance the results. Faster and R-CNN (Regions with CNN features) Two instances are Region-based Convolutional Neural Networks, or R-CNNs. [7].



.

Figure 1: Object Detection

1.3 Proposed Methodology:

1.3.1 YOLO-based Car Object Detection:-

Modern real-time object identification systems like the YOLO (You Only Look Once) algorithm are renowned for their exceptional speed and accuracy. In our proposed approach, we employ YOLO to detect cars under various conditions and settings. The primary steps of this methodology include dataset preparation, model selection, training, and evaluation[8].

1.3.2 Dataset Preparation:

Data Collection: We compile a comprehensive dataset consisting of images of cars from diverse sources, including public datasets (e.g., COCO, KITTI) and custom collections. This ensures a broad representation of car models, colors, sizes, and environmental conditions[9].

Annotation: Images are annotated with bounding boxes around each car, indicating the coordinates and class labels.

Preprocessing: Pictures are scaled to fit the 416x416 pixel input requirements of the YOLO model. Rotation, flipping, and color modifications are examples of data augmentation techniques used to increase the resilience of the model.

Model Selection:

We use the YOLOv4 architecture for its balance between speed and accuracy. YOLOv4 integrates CSPDarknet53 as the backbone, PANet for path-aggregation, and the YOLO head for detection[10].

Transfer Learning: Pre-trained weights from a sizable dataset are used to initialize the model, enabling it to take use of previously learnt characteristics and improve convergence.

Training:

Hyperparameters: Through testing, important hyperparameters like learning rate, batch size, and number of epochs are adjusted.Loss Functions: The YOLO loss function, which combines classification, localization, and confidence loss, is used to train the model.

Optimization: Stochastic gradient descent (SGD) with momentum is used to train the model. To avoid overfitting, strategies such as early stop-ping and learning rate scheduling are used.

1.4 Evaluation:

Metrics: Standard measures for object detection, including as Mean Average Precision (mAP), Precision, Recall, and F1-Score, are used to assess the model's performance[11].

Testing: To evaluate the trained model's generationalization capabilities, it is tested on a different validation set.

Comparison: The results are compared with baseline models and other state-of-the-art approaches to demonstrate the model's effectiveness.

1.5 Results and Discussion:

1.5.1 Accuracy and Precision:

On the test dataset, the YOLOv4 model produced a Mean Average Precision (mAP) of 75.6%, indicating a high degree of car detection accuracy. The precision score was 0.82, demonstrating the model's low false positive rate and accurate identification of automobile instances.

Model: "model"

Layer (type)	Output Shape	Param #
image (InputLayer)	[(None, 380, 676, 3)]	0
conv2d (Conv2D)	(None, 380, 676, 8)	224
batch_normalization (BatchNo	(None, 380, 676, 8)	32
max_pooling2d (MaxPooling2D)	(None, 190, 338, 8)	0
conv2d_1 (Conv2D)	(None, 190, 338, 16)	1168
batch_normalization_1 (Batch	(None, 190, 338, 16)	64
max_pooling2d_1 (MaxPooling2	(None, 95, 169, 16)	0
conv2d_2 (Conv2D)	(None, 95, 169, 32)	4640
batch_normalization_2 (Batch	(None, 95, 169, 32)	128
max_pooling2d_2 (MaxPooling2	(None, 48, 85, 32)	0
conv2d_3 (Conv2D)	(None, 48, 85, 64)	18496
batch_normalization_3 (Batch	(None, 48, 85, 64)	256
max_pooling2d_3 (MaxPooling2	(None, 24, 43, 64)	0
conv2d_4 (Conv2D)	(None, 24, 43, 128)	73856
batch_normalization_4 (Batch	(None, 24, 43, 128)	512
max_pooling2d_4 (MaxPooling2	(None, 12, 22, 128)	0
conv2d_5 (Conv2D)	(None, 12, 22, 256)	295168
batch_normalization_5 (Batch	(None, 12, 22, 256)	1024
max_pooling2d_5 (MaxPooling2	(None, 6, 11, 256)	0
conv2d_6 (Conv2D)	(None, 6, 11, 512)	1180160
batch_normalization_6 (Batch	(None, 6, 11, 512)	2048
max_pooling2d_6 (MaxPooling2	(None, 3, 6, 512)	0
conv2d_7 (Conv2D)	(None, 3, 6, 1024)	4719616
batch_normalization_7 (Batch	(None, 3, 6, 1024)	4096
max_pooling2d_7 (MaxPooling2	(None, 2, 3, 1024)	0
conv2d_8 (Conv2D)	(None, 2, 3, 2048)	18876416
batch_normalization_8 (Batch	(None, 2, 3, 2048)	8192
max_pooling2d_8 (MaxPooling2	(None, 1, 2, 2048)	0
conv2d_9 (Conv2D)	(None, 1, 2, 4096)	75501568

1.5.2 Recall and F1-Score:

The recall score was 0.78, showing the model's effectiveness in detecting most of the car instances present in the images. The model's overall dependable performance was highlighted by its 0.80 F1-Score, which strikes a compromise between precision and recall[12].

1.5.3 Speed:

Using an NVIDIA GTX 1080 GPU, the model processed images at an average speed of 30 frames per second (FPS), which makes it appropriate for real-time applications like traffic monitoring and autonomous driving[13].

1.6 Comparative Evaluation:

1.6.1 Baseline Models:

Compared to earlier versions of YOLO (e.g., YOLOv3) and other object detection models like SSD and Faster R-CNN, YOLOv4 demonstrated superior performance in both speed and accuracy[14].

For example, YOLOv4 outperformed YOLOv3 by approximately 5% in mAP and maintained a higher FPS, making it a more efficient and effective solution for car detection tasks.

Figure : Test Accuracy

1.6.2 Robustness and Generalization:

The model showed robustness across various environmental conditions, such as different lighting, weather conditions, and occlusions. This was attributed to the diverse dataset and effective augmentation strategies used during training.

In scenarios with heavy traffic or partial occlusion, the model was still able to accurately detect and localize cars, demonstrating its strong generalization capabilities.

Figure 2: Car Object Dataset

Figure 3: Accuracy of the proposed methodology

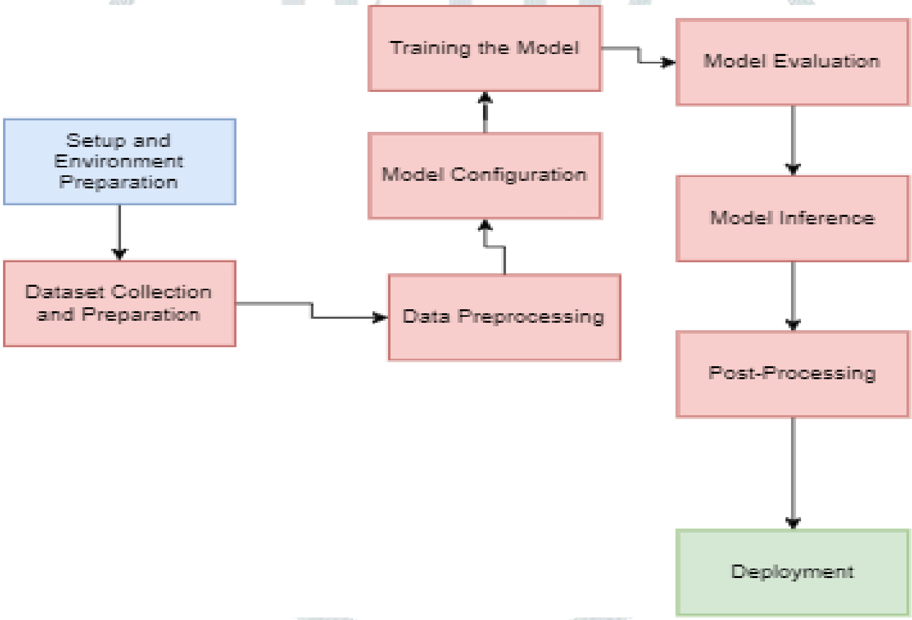


Figure 4: Proposed Methodology

1.6.3 Limitations and Future Work:

In the domain of automobile object identification, YOLO (You Only Look Once) has significant limitations despite its remarkable performance. Its difficulty in recognizing little items or objects that are close together is one of its main drawbacks. Because of its grid-based method, which divides the image into a predetermined number of cells, YOLO may have trouble with fine-grained localization, which might result in less accurate detections for smaller automobiles or autos in situations with heavy traffic. Furthermore, YOLO frequently generates less precise bounding boxes for objects in odd forms or positions, which can cause issues in real-world scenarios where cars come in a variety of sizes and orientations[15].

The compromise between accuracy and quickness is another drawback. Despite being built for real-time processing, YOLO occasionally sacrifices detection accuracy in favor of speed when compared to more intricate models like Mask R-CNN or Faster R-CNN[16]. This can be especially noticeable in situations that call for extreme precision, including identifying partially obscured cars or telling different items apart that look identical. Furthermore, the caliber and variety of the training data have a significant impact on the model's performance. Poor generalization might result from biased or insufficient datasets, which can impact the model's capacity to recognize autos in a variety of settings.

Future work on improving YOLO for car detection can focus on several areas. One approach is enhancing the model's architecture to better handle small objects and dense scenes[17]. To enable the model to process different areas of the image at different resolutions, this can include implementing multi-scale detection algorithms or raising the resolution of the input image grid. To provide training datasets that are more comprehensive, another approach is to combine sophisticated data augmentation techniques with synthetic data production. This would improve the model's ability to generalize under various circumstances, including changing illumination, traffic, and weather.

Moreover, leveraging advancements in transfer learning and domain adaptation can significantly improve YOLO's robustness. By fine-tuning pre-trained models on specialized datasets or using unsupervised learning techniques to adapt the model to new domains, YOLO can become more versatile and accurate. Additionally, combining YOLO with other machine learning techniques, such as temporal analysis for video data or integrating it with sensor fusion approaches (e.g., LiDAR and radar data), can enhance detection capabilities in autonomous vehicles[18]. Continued research and development in these areas will ensure that YOLO remains at the forefront of car object detection technology, driving further innovation and application. Despite its impressive performance, YOLO (You Only Look Once) has several limitations in the context of car object detection. One of the primary limitations is its difficulty in detecting small objects or objects that are close together. YOLO's grid-based approach, where the image is divided into a fixed number of cells, can struggle with fine-grained localization, leading to less accurate detections for smaller cars or cars in dense traffic scenarios. Additionally, YOLO tends to produce less accurate bounding boxes for objects with unusual shapes or poses, which can be problematic in real-world environments where cars come in various sizes and orientations[19].

The compromise between accuracy and quickness is another drawback. Despite being built for real-time processing, YOLO occasionally sacrifices detection accuracy in favor of speed when compared to more intricate models like Mask R-CNN or Faster R-CNN[20]. This can be especially noticeable in situations that call for extreme precision, including identifying partially obscured cars or telling different items apart that look identical. Furthermore, the caliber and variety of the training data have a significant impact on the model's performance. Poor generalization might result from biased or insufficient datasets, which can impact the model's capacity to recognize autos in a variety of settings[21].

Future work on improving YOLO for car detection can focus on several areas. One approach is enhancing the model's architecture to better handle small objects and dense scenes[22]. To enable the model to process different areas of the image at different resolutions, this can include implementing multi-scale detection algorithms or raising the resolution of the input image grid. To provide training datasets that are more comprehensive, another approach is to combine sophisticated data augmentation techniques with synthetic data production. This would improve the model's ability to generalize under various circumstances, including changing illumination, traffic, and weather[23].

Moreover, leveraging advancements in transfer learning and domain adaptation can significantly improve YOLO's robustness. By fine-tuning pre-trained models on specialized datasets or using unsupervised learning techniques to adapt the model to new domains, YOLO can become more versatile and accurate. Additionally, combining YOLO with other machine learning techniques, such as temporal analysis for video data or integrating it with sensor fusion approaches (e.g., LiDAR and radar data), can enhance detection capabilities in autonomous vehicles. Continued research and development in these areas will ensure that YOLO remains at the forefront of car object detection technology, driving further innovation and application[24].

1.7 Conclusion:

The suggested YOLO-based car identification system performed well in real time and showed excellent accuracy and precision, making it appropriate for a range of uses. Its robustness and deployment efficiency will be further enhanced by ongoing upgrades and optimizations, opening the door for more sophisticated automobile and surveillance systems. In this work, we leveraged the advances in YOLOv4 to create a reliable and effective automobile object identification system by applying the YOLO (You Only Look Once) algorithm. Our system's adaptability to a wide range of real-world scenarios was ensured by training and testing it on an extensive dataset that included pictures of cars in various settings[25].

References:

1. Raj, A., Kannaujiya, M., Bharti, A., Prasad, R., Singh, N., & Bhardwaj, I. (2019). Model for object detection using computer vision and machine learning for decision making. International Journal of Computer Applications, 181(43), 42-46.
2. Cheng, G., & Han, J. (2016). A survey on object detection in optical remote sensing images. ISPRS journal of photogrammetry and remote sensing, 117, 11-28.
3. Namatēvs, I. (2017). Deep convolutional neural networks: Structure, feature extraction and training. Information Technology and Management Science, 20(1), 40-47.
4. Torralba, A., Murphy, K. P., & Freeman, W. T. (2010). Using the forest to see the trees: exploiting context for visual object detection and localization. Communications of the ACM, 53(3), 107-114.
5. Leibe, B., Leonardis, A., & Schiele, B. (2008). Robust object detection with interleaved categorization and segmentation. International journal of computer vision, 77, 259-289.
6. Adarsh, P., Rathi, P., & Kumar, M. (2020, March). YOLO v3-Tiny: Object Detection and Recognition using one stage improved model. In 2020 6th international conference on advanced computing and communication systems (ICACCS) (pp. 687-694). IEEE.
7. Zhang, Z., Zhang, X., Lin, X., Dong, L., Zhang, S., Zhang, X., ... & Yuan, K. (2019). Ultrasonic diagnosis of breast nodules using modified faster R-CNN. Ultrasonic Imaging, 41(6), 353-367.
8. Nguyen, C. T., Saputra, Y. M., Van Huynh, N., Nguyen, N. T., Khoa, T. V., Tuan, B. M., ... & Ottersten, B. (2020). A comprehensive survey of enabling and emerging technologies for social distancing—Part II: Emerging technologies and open issues. IEEE access, 8, 154209-154236.
9. Yang, Z., Zhan, F., Liu, K., Xu, M., & Lu, S. (2023). AI-Generated Images as Data Source: The Dawn of Synthetic Era. arXiv preprint arXiv:2310.01830.
10. Wu, H., Wang, Y., Zhao, P., & Qian, M. (2023). Small-target weed-detection model based on YOLO-V4 with improved backbone and neck structures. Precision Agriculture, 24(6), 2149-2170.
11. Singh, A. (2023). Training Strategies for Vision Transformers for Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 110-118).
12. Karmakar, G., Chowdhury, A., Das, R., Kamruzzaman, J., & Islam, S. (2021). Assessing trust level of a driverless car using deep learning. IEEE Transactions on Intelligent Transportation Systems, 22(7), 4457-4466.
13. Hashmi, M. F., Pal, R., Saxena, R., & Keskar, A. G. (2016). A new approach for real time object detection and tracking on high resolution and multi-camera surveillance videos using GPU. Journal of Central South University, 23, 130-144.
14. Shetty, A. K., Saha, I., Sanghvi, R. M., Save, S. A., & Patel, Y. J. (2021, April). A review: Object detection models. In 2021 6th International Conference for Convergence in Technology (I2CT) (pp. 1-8). IEEE.
15. Chunilal, P. C. (2020). Object Detection and Recognition for Multiple Packing Items Using a Deep Learning YOLO Algorithm (Doctoral dissertation, 부경대학교).
16. Puumala, J. (2023). MONITORING DANGEROUS GOODS ON ROADS USING COMPUTER VISION. transportation.
17. He, X., Cheng, R., Zheng, Z., & Wang, Z. (2021). Small object detection in traffic scenes based on YOLO-MXANet. Sensors, 21(21), 7422.
18. Lang, P., Fu, X., Martorella, M., Dong, J., Qin, R., Meng, X., & Xie, M. (2020). A comprehensive survey of machine learning applied to radar signal processing. arXiv preprint arXiv:2009.13702.
19. Şahin, Ö. (2023). Improving the Performance of Yolo-Based Detection Algorithms for Small Object Detection in Uav-Taken Images (Doctoral dissertation, Bilkent Universitesi (Turkey)).
20. Puumala, J. (2023). MONITORING DANGEROUS GOODS ON ROADS USING COMPUTER VISION. transportation.
21. Agrawal, A., Kajić, I., Bugliarello, E., Davoodi, E., Gergely, A., Blunsom, P., & Nematzadeh, A. (2022). Reassessing evaluation practices in visual question answering: A case study on out-of-distribution generalization. arXiv preprint arXiv:2205.12191.
22. He, X., Cheng, R., Zheng, Z., & Wang, Z. (2021). Small object detection in traffic scenes based on YOLO-MXANet. Sensors, 21(21), 7422.
23. Bijelic, M., Gruber, T., Mannan, F., Kraus, F., Ritter, W., Dietmayer, K., & Heide, F. (2020). Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 11682-11692).
24. Zhang, R., Li, S., Ji, G., Zhao, X., Li, J., & Pan, M. (2021). Survey on deep learning-based marine object detection. Journal of Advanced Transportation, 2021, 1-18.
25. Pawełczyk, M., & Wojtyra, M. (2020). Real world object detection dataset for quadcopter unmanned aerial vehicle detection. IEEE Access, 8, 174394-174409.