Vol. 1, No. 1, September 2024

IJAIS International Journal of Artificial Intelligence and Science

Publsihed By Asosiasi Doktor Sistem Informasi Indonesia

Traffic Vehicle Detection Using Faster R-CNN: A Comparative Analysis of Backbone Architectures

Luqman Hakim[1], Aria Hendrawan[2], Rofiatul Khoiriyah[3]

[1,2,3]Department of Informatics Technology, Faculty of Information Technology and Communication, Semarang University, Semarang, Indonesia

Email: g211190073@student.usm.ac.id[1], ariahendrawan@usm.ac.id[2], g211190073@student.usm.ac.id[3]

Received:
August 3, 2024
Revised:
August 31, 2024
Accepted:
September 29, 2024
Published:
September 30, 2024

Corresponding Author:
Author Name*:
Aria Hendrawan
Email*:
ariahendrawan@usm.ac.id

DOI: 10.63158/IJAIS.v1.i1.5

Abstract. Object detection is a crucial task in computer vision, where advanced deep learning models have shown significant improvements over traditional methods. In this study, the Faster R-CNN algorithm is applied to a traffic dataset containing six vehicle categories: Bus, Car, Motorcycle, Pick Up Car, Truck, and Truck Box. The novelty of the research lies in the comparison of four backbone architectures ResNet50, ResNet50V2, MobileNetV3 Large, and MobileNetV3 Large 320 evaluated for their performance in vehicle detection at IoU thresholds of 0.5 and 0.75. The results reveal that ResNet50 provided the best overall performance, achieving mAP scores of 0.966 at IoU 0.5 and 0.887 at IoU 0.75, offering a balanced trade-off between precision and recall. ResNet50V2 and MobileNetV3 Large also performed well, with mAP scores of 0.945 and 0.870 for ResNet50V2, and 0.969 and 0.843 for MobileNetV3 Large, respectively. However, MobileNetV3 Large 320 showed the lowest detection performance, with mAP scores of 0.857 at IoU 0.5 and 0.551 at IoU 0.75. These findings provide useful insights into the suitability of different architectures for vehicle detection tasks, particularly in traffic surveillance applications.

Keywords: deep learning, Faster R-CNN, MobileNetV3, ResNet50, vehicle detection, computer vision

1. INTRODUCTION

The area of computer vision has extensively researched the subject of object detection. To satisfy the expanding demand for precise object detection models, many strategies

IJAIS International Journal of Artificial Intelligence and Science
Publsihed By Asosiasi Doktor Sistem Informasi Indonesia

are applied [1]. Bounding boxes are used in object detection to forecast each location of the object in addition to categorizing different types of objects [2]. Object detection has been used in a variety of industries, including driverless vehicles, surveillance, and more [3]. Deep neural networks are one of the advanced machine learning techniques needed to accomplish object detection. Object Detection has received the most study attention from 2021 until 2023, according to a bibliometrix [4] survey of 2000 articles, as shown in Figure 1.

Deep neural networks (DNNs) may learn the required representations purely from the provided raw data since they are made up of a large number of neurons and more than one hidden layer that are arranged in a highly nested network design. Deep learning is the term used to describe this practice [5]. The advancement of computer vision has accelerated with the advent of deep learning technology [1]. The convolutional neural network (CNN), a variation of DNN that enhances the capabilities of a multi-layer perceptron (MLP), has the advantage of utilizing the topological information of the input sample and being unaffected by post-processing transformations like resizing, translation, and others [6]. There are several algorithms that uses CNN including Faster-CNN [7], YOLO [8], and SSD [9].

In [7] Ren, i.e. proposed the Faster R-CNN (Faster Region-Convolutional Neural Network) object identification architecture in 2015. This architecture makes full use of the capabilities of Graphics Processing Units (GPUs) by using the Region Proposal Network (RPN) method to construct bounding boxes instead of more conventional techniques. By adding the RPN, Faster R-CNN accelerates and improves object identification by performing object categorization and region proposal generation in the same neural network architecture. By doing away with distinct region proposal techniques, this method improves performance and GPU resource utilization [10].

In this paper, we use two different backbones: ResNet50 and MobileNetV3 Large. ResNet50 is a deep convolutional neural network design that has one fully connected layer at the end and 49 convolutional layers throughout [11]. It is notable for its residual connections, which help to address the vanishing gradient problem and enable the training of incredibly deep networks. The MobileNetV3 architecture expands on the strengths of MobileNetV1 and MobileNetV2, its forerunners. One of the significant

innovations in MobileNetV3 is the utilization of a novel block known as the "Mobile Inverted Residual Bottleneck" (MBConv), which consists of a light depthwise separable convolution, a linear bottleneck, and another depthwise separable convolution [12]. The MBConv block minimizes the number of parameters and computing costs while enabling effective information flow. A highly optimized CNN architecture called MobileNetV3 Large was created specifically for effective image categorization on embedded and mobile devices [12]. After we train the model using Faster R-CNN and four different backbones, we deployed the machine learning model using streamlit library.

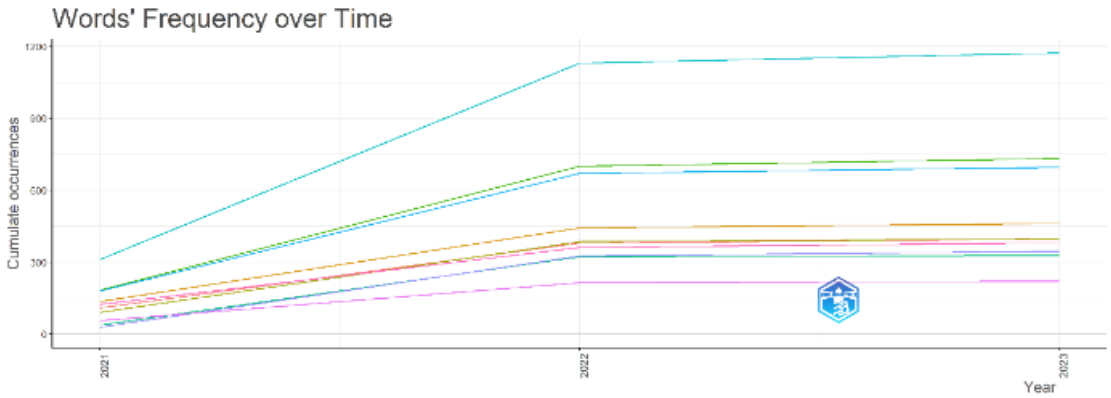

Figure 1. Bibliometrix survey on the last three years

In this study, we use Faster RCNN to detect six different classes of vehicles using ResNet50 and MobileNetV3 Large as the backbone. Hence, we hope that this research will lead to the development of a highly accurate machine learning model to aid in the detection of vehicles in traffic.

2. METHODS

2.1 Dataset

The dataset utilized in this case is an Indonesian traffic dataset in a Pascal VOC XML format. This collection contains 1104 pictures, each with a resolution of 480 by 480 pixels. The object on the image in this dataset have been classified with six vehicle classes: "Bus, Car, Motorcycle, Pick Up Car, Truck," and "Truck Box" Figure 2. The total 1104 pieces of data consist of 80% training data, 15% validation data, and 5% testing data [13].

IJAIS International Journal of Artificial Intelligence and Science

Publsihed By Asosiasi Doktor Sistem Informasi Indonesia

Figure 2. six class object in dataset "Bus", "Car", "Motorcycle", "Pick Up Car", "Truck", "Truck Box"

2.2 Model Development

In this research, we used a Faster R-CNN implemented in PyTorch. We employed a Faster R-CNN built in PyTorch in this study. Backbones of two sorts were used: ResNet-50 and MobileNetV3. ResNet-50 was utilized in two ways: resnet50 and resnet50_v2. The pre-trained models resnet50 and resnet50 _v2 utilize ResNet-50 as the backbone network and the Feature Pyramid Network (FPN) to enhance identification of tiny objects and boost detection precision. The difference between the two models is that the second is an upgraded version of the prior model, including changes to the FPN architecture to increase detection accuracy [14]. As for the MobileNetV3, we used a mobilenet_V3_large_fpn and mobilenet_V3_large_320_fpn model. Those two is a pre-trained models that uses MobileNetV3 Large as a backbone network and FPN to improve the detection precision. The difference between those two version is that the fasterrcnn_mobilenet_v3_large_320_fpn is more lightweight, because it is focused and specifically used for an input size of 320 x 320 pixels.

All backbones were trained under the same conditions for 40 epochs. Two T4 Tesla GPUs provided by Kaggle were utilized for the training phase. For every backbone utilized, the training step required in the range of three to five hours. A higher IoU implies a bigger overlap between the predicted bounding box and the ground truth bounding box, which signals better localization accuracy. The IoU (Intersection of Union) was assessed by the overlap predicted bounding box and ground truth bounding box. As a result, more precise

IJAIS International Journal of Artificial Intelligence and Science
Publsihed By Asosiasi Doktor Sistem Informasi Indonesia

detections are made since this model predictions closely match the positions of the real objects. The formula for IoU is as shown in Equation 1.

$$J(A,B)=\frac{|A\cap B|}{|AUB|} \tag{1}$$

Where A is predicted bounding box and B is the ground truth bounding box. $|A \cap B|$ means area of the overlapping bounding boxes, and $|A \cup B|$ means the cardinality or size of the union of predicted bounding box and ground truth bounding box.

The evaluation was done by calculating the mAP, AP, Precision, and Recall value. mAP (mean Average Precision) is a commonly used metric for evaluating the performance of the trained object detection models, including Faster RCNN. The formula for the Precision and Recall is as shown in Equation 2 and 3.

$$Precision=\frac{TP}{TP+FP} \tag{2}$$

$$Recall=\frac{TP}{TP+FN} \tag{3}$$

Where TP is True Positive, FP is False Positive and FN is False Negative. Precision is ratio of true positive detections to the total number of detections, and recall is the ratio of true positive detections to the total number of ground truth objects. As for the mAP, we need to calculate the AP (Average Precision) first use Equation 4.

$$AP=\int_{r=0}^{1} p(r)dr \tag{4}$$

Where $p(r)$ is precision at a specific recall level r, r is recall, and the integral symbol represents the mathematical integration over a range of values, while in this case the integration is performed from $r = 0$ to $r = 1$, which means we are integrating the precision values over the entire recall range from 0 to 1. As for the mAP formula is as shown in Equation 5.

$$mAP=\frac{1}{N}\times\sum_{i}^{N} APi \tag{5}$$

Where N is the number of object categories, and AP_i is the Average Presicion for category i

IJAIS International Journal of Artificial Intelligence and Science
Publsihed By Asosiasi Doktor Sistem Informasi Indonesia

3. RESULTS AND DISCUSSION

3.1 Performance Evaluation

The performance of the Faster R-CNN model using four different backbone architectures—ResNet50, ResNet50V2, MobileNetV3 Large, and MobileNetV3 Large 320—was evaluated on the Pascal VOC dataset. The results are presented in Table 1 (IoU = 0.5) and Table 2 (IoU = 0.75), and the prediction results using these backbones are illustrated in Figure 3.

Table 1. Evaluation results in IoU=0.5

Backbone	Precision	Recall	mAP
ResNet50	0.918	0.989	0.966
ResNet50V2	0.927	0.970	0.945
MobileNetV3 - Large	0.926	0.983	0.969
MobileNetV3 - Large 320	0.881	0.869	0.857

Table 1 shows the results for IoU = 0.5. Here, ResNet50 demonstrated the best overall performance with a precision of 0.918, recall of 0.989, and mAP of 0.966, balancing precision and recall effectively. ResNet50V2 followed closely with a precision of 0.927 and recall of 0.970, achieving a mAP of 0.945.

MobileNetV3 Large achieved the highest mAP of 0.969 with a precision of 0.926 and recall of 0.983, proving to be a competitive backbone. However, MobileNetV3 Large 320 exhibited the lowest performance, with a precision of 0.881, recall of 0.869, and mAP of 0.857.

Table 2. Evaluation results in IoU=0.75

Backbone	Precision	Recall	mAP
ResNet50	0.876	0.936	0.887
ResNet50V2	0.889	0.920	0.870
MobileNetV3 - Large	0.860	0.884	0.843
MobileNetV3 - Large 320	0.682	0.615	0.551

Table 2 displays the results for IoU = 0.75, where all models showed reduced performance due to the stricter confidence threshold. ResNet50 maintained its lead with a precision

of 0.876, recall of 0.936, and mAP of 0.887, performing well under stricter detection conditions.

ResNet50V2 achieved a precision of 0.889, recall of 0.920, and a mAP of 0.870, remaining a solid alternative. MobileNetV3 Large showed a slight drop with a precision of 0.860, recall of 0.884, and a mAP of 0.843, still proving useful for object detection. MobileNetV3 Large 320 had the lowest performance, with a precision of 0.682, recall of 0.615, and a mAP of 0.551, indicating it is less suited for high-confidence object detection tasks.

Figure 3. Prediction Results Using Four Different Backbones

The prediction results from the Faster R-CNN model with the four different backbones are visualized in Figure 3. This figure illustrates the qualitative differences in detection performance for each backbone across a variety of images from the dataset. The figure supports the quantitative findings in Tables 1 and 2, showing that ResNet50 and MobileNetV3 Large provide the most reliable detection results, while MobileNetV3 Large 320 exhibits weaker detection accuracy.

Across both IoU thresholds, ResNet50 proved to be the most balanced backbone, providing high performance under both loose and strict detection conditions.

IJAIS International Journal of Artificial Intelligence and Science
Publsihed By Asosiasi Doktor Sistem Informasi Indonesia

MobileNetV3 Large showed promise, achieving the highest mAP at IoU = 0.5, but its performance dropped more significantly at IoU = 0.75. ResNet50V2 maintained consistent performance throughout, and MobileNetV3 Large 320 struggled under both IoU thresholds, making it less suitable for higher-confidence detection tasks.

After completing the evaluations, the models were deployed using the Streamlit library, enabling both image and video inference. Two modes were provided: "real-time" and "not real-time" detection. In "real-time" mode, the file is processed immediately upon upload, while in "not real-time" mode, the file is processed in the background, and a downloadable output is provided once detection is complete. The model operates on a CPU, so processing times may vary. The deployed model can be accessed at https://fasterrcnnvdapp.streamlit.app.

3.2 Discussion

The results of this study offer valuable insights into the effectiveness of different backbone architectures when applied to the Faster R-CNN model for vehicle detection tasks on the Pascal VOC dataset. The evaluation at two Intersection over Union (IoU) thresholds—0.5 and 0.75—highlights the strengths and weaknesses of each backbone, enabling a clear understanding of their suitability for different object detection scenarios.

The performance of ResNet50 stands out across both IoU thresholds, achieving an mAP of 0.966 at IoU = 0.5 and 0.887 at IoU = 0.75. These results show that ResNet50 offers the most balanced trade-off between precision and recall, making it a reliable choice for vehicle detection. Its high precision at both IoU levels indicates strong accuracy in detecting vehicle objects with minimal false positives, while its recall ensures a low rate of missed detections. The consistent results suggest that ResNet50 is well-suited for real-world applications where both precision and recall are crucial.

On the other hand, ResNet50V2 delivered slightly lower performance compared to ResNet50, with a mAP of 0.945 at IoU = 0.5 and 0.870 at IoU = 0.75. While it maintained competitive precision and recall scores, it was marginally outperformed by its predecessor, ResNet50. However, ResNet50V2 still demonstrated strong detection

abilities, making it a viable alternative, particularly in scenarios where slight variations in performance are acceptable. The slight dip in performance might be attributed to architectural differences that prioritize certain computational efficiencies over detection accuracy.

MobileNetV3 Large proved to be a competitive contender, outperforming all other backbones in terms of mAP at IoU = 0.5, with a score of 0.969. Its precision and recall values were close to those of ResNet50, making it a strong choice for object detection, particularly when a lightweight model is preferred. However, at IoU = 0.75, MobileNetV3 Large experienced a more noticeable drop in performance, with an mAP of 0.843. This decline suggests that while MobileNetV3 Large excels under less stringent conditions, it struggles to maintain the same level of performance when higher confidence is required in detection tasks. The backbone's lightweight architecture likely contributes to this performance drop, as it may prioritize speed and efficiency over accuracy at higher thresholds.

In contrast, MobileNetV3 Large 320 showed the weakest performance of all the tested backbones. With a mAP of 0.857 at IoU = 0.5 and a significant drop to 0.551 at IoU = 0.75, this backbone exhibited clear limitations, particularly under stricter detection conditions. The lower precision and recall values further emphasize that this model is less reliable for high-confidence object detection. The architectural limitations of the MobileNetV3 Large 320, which is optimized for resource-constrained environments, likely contribute to its reduced performance, especially in tasks that require a high degree of accuracy. These findings suggest that while MobileNetV3 Large 320 may be useful in scenarios where computational resources are severely limited, it may not be the best choice for tasks requiring robust detection performance.

The results across the two IoU thresholds reveal that performance tends to degrade as the IoU threshold increases from 0.5 to 0.75, which is expected in object detection tasks. This is particularly evident with MobileNetV3 Large 320, where the performance drop was the most significant. The stricter IoU threshold requires the predicted bounding boxes to be closer to the ground truth boxes, which naturally leads to a decrease in the number of positive detections. This trend is common in object detection tasks but is more

pronounced in lightweight architectures like MobileNetV3 Large 320, which are designed with fewer parameters and less computational complexity.

On the other hand, ResNet50 and ResNet50V2 displayed more gradual declines in performance as the IoU threshold increased, suggesting that these backbones are more resilient to stricter detection conditions. This resilience makes them more suitable for applications where precision is critical, such as autonomous driving or traffic surveillance, where misdetections could have significant consequences.

The results illustrate the inherent trade-offs between precision, recall, and mAP across the different backbones. ResNet50 consistently delivered a strong balance between precision and recall, making it ideal for tasks that demand both high accuracy and low false positive rates. MobileNetV3 Large, while excelling at IoU = 0.5, saw a sharper drop in recall at IoU = 0.75, indicating that while it may be competitive in less strict scenarios, it may not be the best choice for tasks that require high-confidence detections. ResNet50V2 similarly provided solid performance, though it did not outperform the original ResNet50, making it a viable but slightly less effective option. Lastly, MobileNetV3 Large 320 fell short in both precision and recall, indicating that its design sacrifices detection accuracy for efficiency, making it less suitable for scenarios where performance is a priority.

From a practical standpoint, the choice of backbone architecture depends heavily on the application and the available computational resources. ResNet50 is recommended for applications that require a robust and reliable model capable of performing well even under strict IoU thresholds. This makes it particularly well-suited for use cases such as autonomous driving or traffic monitoring, where accuracy and reliability are paramount. On the other hand, MobileNetV3 Large offers a more lightweight alternative that still delivers strong performance at lower IoU thresholds. It could be useful in scenarios where computational resources are limited, such as mobile or embedded systems, but its drop in performance at higher IoU thresholds should be considered if high-confidence detection is essential.

Lastly, MobileNetV3 Large 320, while the least accurate, may still be a viable option in environments where computational resources are extremely constrained, and the detection task does not require high precision or recall.

ResNet50 emerges as the most versatile and robust backbone for vehicle detection tasks, consistently offering the best balance between precision and recall across different IoU thresholds. MobileNetV3 Large also shows promise, particularly in less stringent detection scenarios, while ResNet50V2 provides a solid alternative to ResNet50. However, MobileNetV3 Large 320 is less suited for tasks requiring high-confidence detection, and its use should be limited to low-resource environments where performance trade-offs are acceptable.

4. CONCLUSION

This study evaluated four backbone architectures—ResNet50, ResNet50V2, MobileNetV3 Large, and MobileNetV3 Large 320—using Faster R-CNN for vehicle detection. ResNet50 proved to be the most reliable, achieving the best balance between precision and recall across IoU thresholds, making it an ideal choice for high-accuracy vehicle detection in traffic monitoring systems. MobileNetV3 Large demonstrated competitive performance, particularly at lower IoU thresholds, which could be useful in applications where computational resources are limited, such as mobile or embedded systems. However, MobileNetV3 Large 320 struggled under more demanding conditions, making it less suitable for tasks requiring high precision. The results of this study have direct implications for traffic surveillance systems. ResNet50 could be deployed in real-time traffic monitoring to accurately detect vehicles in high-density traffic, offering a robust solution for applications such as automated traffic control, congestion monitoring, or accident detection. MobileNetV3 Large could be useful in resource-constrained environments, such as smart cameras or edge devices, where efficiency is key.

ACKNOWLEDGMENT

The Authors would like to express our gratitude to the Study Program of Informatics Engineering at Faculty of Information Technology and Communication in Semarang University for their unwavering support in this research.

IJAIS International Journal of Artificial Intelligence and Science
Publsihed By Asosiasi Doktor Sistem Informasi Indonesia

REFERENCES

[1] R. Padilla, S. L. Netto, and E. A. B. Da Silva, "A Survey on Performance Metrics for Object-Detection Algorithms," in *Int. Conf. Syst. Signals Image Process.*, vol. 2020-July, pp. 237–242, Jul. 2020, doi: 10.1109/IWSSIP48289.2020.9145130.

[2] X. Wu, D. Sahoo, and S. C. H. Hoi, "Recent Advances in Deep Learning for Object Detection," *ACM Comput. Surv.* (incomplete information, add more details).

[3] H. Zhang and X. Hong, "Recent progresses on object detection: a brief review," *Multimed. Tools Appl.*, vol. 78, pp. 27809–27847, 2019, doi: 10.1007/s11042-019-07898-2.

[4] M. Aria and C. Cuccurullo, "bibliometrix: An R-tool for comprehensive science mapping analysis," *J. Informetr.*, vol. 11, no. 4, pp. 959–975, Nov. 2017, doi: 10.1016/J.JOI.2017.08.007.

[5] C. Janiesch, P. Zschech, and K. Heinrich, "Machine learning and deep learning," *Electron. Mark.*, vol. 31, no. 2, pp. 685–695, Apr. 2021, doi: 10.1007/s12525-021-00475-2.

[6] S. Albahli, M. Nawaz, A. Javed, and A. Irtaza, "An improved Faster R-CNN model for handwritten character recognition," *Arab. J. Sci. Eng.*, vol. 46, no. 3, pp. 8509–8523, 2021, doi: 10.1007/s13369-021-05471-4.

[7] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.

[8] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-December, pp. 779–788, Jun. 2016, doi: 10.1109/CVPR.2016.91.

[9] W. Liu et al., "SSD: Single shot multibox detector," in *Lect. Notes Comput. Sci.*, vol. 9905, pp. 21–37, 2016, doi: 10.1007/978-3-319-46448-0_2.

[10] W. Yang, Z. Li, C. Wang, and J. Li, "A multi-task Faster R-CNN method for 3D vehicle detection based on a single image," *Appl. Soft Comput.*, vol. 95, p. 106533, Oct. 2020, doi: 10.1016/J.ASOC.2020.106533.

[11] S. Sinha and N. Gupta, "Computer-aided diagnosis of malaria through transfer learning using the ResNet50 backbone," *J. Med. Syst.*, vol. 36, no. 4, 2023.

[12] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, and Q. V. Le, "Searching for MobileNetV3," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, pp. 1314–1324, 2019.

[13] A. Hendrawan, R. Gernowo, O. D. Nurhayati, B. Warsito, and A. Wibowo, "Improvement Object Detection Algorithm Based on YoloV5 with BottleneckCSP," in *Proc. IEEE Int. Conf. Commun. Netw. Satellite (COMNETSAT)*, pp. 79–83, 2022, doi: 10.1109/COMNETSAT56033.2022.9994461.

[14] S. Yang, D. Jiao, T. Wang, and Y. He, "Tire Speckle Interference Bubble Defect Detection Based on Improved Faster RCNN-FPN," *Sensors*, vol. 22, no. 10, p. 3907, May 2022, doi: 10.3390/S22103907.