Optimizing Lightweight Object Detection Models for Autonomous Driving: A Comparative Study of Model Compression, Real-Time Performance, and Transfer Learning for Resource-Constrained Devices

Miss. Mane Dhanshree Ravso

Department of Computer Science and Engineering, Ashokarao Mane Group of Instituations Vatar tarf Vadgaon

DOI: https://doi.org/10.51584/IJRIAS.2025.10040021

Received: 11 April 2025; Accepted: 15 April 2025; Published: 01 May 2025

ABSTRACT

This paper discusses strategies for optimizing lightweight object detection models in autonomous driving, with regard to performance improvements on resource-constrained embedded platforms, such as the Nvidia Jetson and Raspberry Pi. The key optimization techniques involved are channel pruning and quantization, reducing model size and computational complexity that improve inference speed and efficiency. It also discusses the improvement of real-time detection speed, including lightweight architectures and pipeline optimizations to meet the stringent frame rate requirements of autonomous vehicles. It also discusses methods for improving detection accuracy in complex environments, such as urban streets and adverse weather conditions. The paper emphasizes the importance of balancing efficiency, accuracy, and speed to ensure the feasibility and safety of object detection in autonomous driving systems.

Keywords: Lightweight Object Detection, Embedded Platforms, Channel Pruning, Real-time Detection Speed

INTRODUCTION

Autonomous driving is one of the transformative technologies with the promise of revolutionizing transport. Object detection is an important part of such systems, which means detecting and identifying objects including vehicles, pedestrians, and traffic signs in real time. The importance of object detection lies in its role as a foundation for decisions in autonomous vehicles. Environmental perception ensures that vehicles can predict obstacles and respond in a timely manner to dynamic traffic conditions [1], [2].

To ensure dependable object detection, autonomous systems are based on embedded platforms such as Nvidia Jetson and Raspberry Pi, optimized for resource-constrained environments [3]. Such systems are vital for real-time decision-making as they have to balance the demand for computation with energy efficiency. Embedded systems are becoming increasingly critical as they allow for low-latency computation, which is necessary for real-world scenarios. For example, the scalable computing power of the Nvidia Jetson platform can be designed for image recognition and object detection, thus making the feasibility of autonomous navigation possible even in resource-constrained environments [4]. However, they have to trade off between speed, accuracy, and power consumption to make them work properly.

To this end, [5] recommended algorithms to help optimize object detection frameworks for the purpose of higher accuracy and inference speed through introducing feature fusion and pruning methods. Such enhancements ensure that models maintain real-time capabilities while operating within the constraints of embedded hardware [5]. Figure 1 below illustrates the optimization process within autonomous driving systems. It shows how object detection integrates with embedded platforms like Nvidia Jetson and Raspberry Pi, which are crucial for real-time processing. The diagram highlights key challenges such as computational limitations and the need for rapid decision-making. It also outlines several optimization techniques—channel pruning, quantization, feature fusion, and adaptive image scaling—that improve performance while addressing these constraints. The motivation behind optimizing the 'object detection' task is that, despite its very challenging application to run fast and accurately on resource-constrained embedded platforms, models such as YOLOv4 have reported various ways of striking this balance by using pruning and attention mechanisms [2]. Another approach that is increasingly popular applies adaptive image scaling to reduce latency without sacrificing the accuracy of detection [6]. Optimizing object detection models thus ensures that autonomous driving is safe, efficient, and scalable.

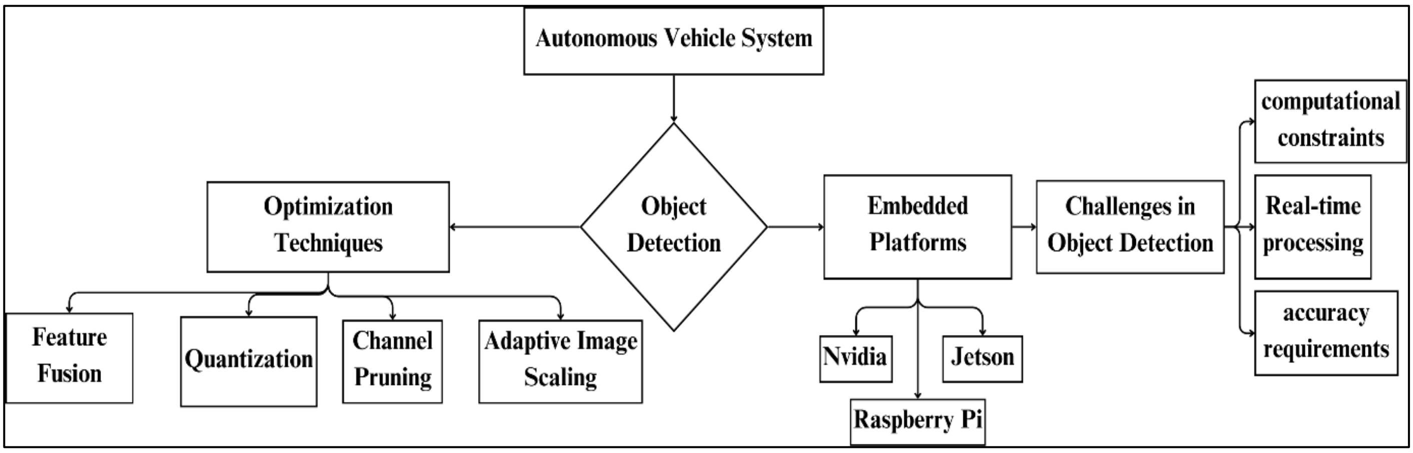

Figure 1 : Object Detection in Autonomous Driving Systems

This paper mainly discusses the optimization techniques of lightweight object detection models for autonomous driving. In this context, it deals with channel pruning, quantization, real-time speed enhancement, and accuracy improvement in complex environments. Evaluation of existing models provides insights into their performance on resource-constrained platforms like Nvidia Jetson and Raspberry Pi.

The scope of the paper involves model optimization, real-time speed enhancement, accuracy in complex scenarios, and comparative evaluation of lightweight models; it concludes with future research directions in autonomous driving object detection.

Model Optimization Techniques for Lightweight Object Detection

To address the deployment of object detection models on embedding devices, limited by computational storage, researchers had proposed algorithms known as channel pruning and quantization. These make more efficient uses of resources without very much loss to performance.

Channel pruning is a technique to selectively remove less important channels or filters from the convolutional layers of the neural network for both model size and computational complexity reduction. Recently, localization-aware pruning was proposed by [7] for focus on the preservation of channels important for both classification and localization tasks in object detection [7]. Similar to the above examples[8] used soft-gated modules that identified unimportant channels during training and trimmed them off, thus having fewer parameters and quicker inferences without any compromise on accuracy. [9] further progressed with this idea by generating knowledge distillation-based pruning with integrity of detections even after using smaller models. These pruning strategies are particularly useful for edge devices where memory and computational power are constrained.

Quantization reduces model parameters by converting floating-point weights into lower-precision formats, such as fixed-point or integer representations. [10] proved that QAT is superior to PTQ for preserving accuracy, especially for more challenging datasets. Another method is GradQuant, which is a clipping-free quantization method introduced by [11], that prevents accuracy loss due to outlier activations, thereby improving model performance. Quantization is a method used to reduce the storage requirements and accelerate the inference speed, thus it is very suitable for deployment on low-power devices.

Some researchers further push the optimization to the extreme by trying a combination of both pruning and quantization under one framework. [12] suggested the integration of both of these techniques using reinforcement learning, ensuring an optimal model compression and avoiding divergence during sequential pruning and quantization. QATFP-YOLO proposed by [13] combined QAT with filter pruning exhibiting outstanding performance even on non-GPU devices, as noted earlier.

Methods that take advantage of channel pruning and quantization combined improve remarkably the feasibility of deployment of object detection models on resource-limited embedded platforms.

Enhancing Real-Time Detection Speed

To overcome the limitations of real-time object detection for autonomous vehicles, several algorithms have been proposed and developed to counter the trade-offs between speed, accuracy, and resource constraints. In autonomous driving, operating at high frame-per-second rates is crucial; it ensures timely, accurate decisions on its environment and avoids any chance collisions. Object detection must run in real time in such dynamic environments. [2] placed importance on efficient detection in different traffic conditions and introduced low-latency models to support real-time decision-making. Next, [14] showed how MobileYOLO improves the FPS without loss of accuracy in detection.

To address these real-time constraints, a variety of optimization techniques have been developed. Some architectures, such as YOLO-tiny, MobileNetV2, and EfficientDet, are optimized for real-time applications. [15] also proposed ALODAD, an anchor-free lightweight detector that jointly relies on GhostNet for the decrease in the computational cost while high accuracy and real-time performance are achieved. [16], similarly proposed a YOLOGP efficient extension of YOLOv5 by taking the upsampling and compression techniques into consideration so as to support faster inference.

Real-time performance also relies on hardware acceleration, aside from architectural optimizations. Optimizing the efficiency of embedded devices with GPU or TPU support can significantly improve computational efficiency. [17] employed optical flow estimation to enhance the computational efficiency with queue stability in NVIDIA platforms and is thus very suitable for real-time object detection. Pipelining optimizes for further real-time performance. Methods, including early exit strategies and adaptive inference, allow the system to balance computational cost with accuracy. [18] used frame prioritization in edge computing, which permitted improvement of FPS on Jetson TX2 without impairing the detection capabilities.

The challenge persists in balancing speed with accuracy in autonomous vehicle driving systems. [5] proposed the YOLOv4-5D, a framework that achieves high FPS with minimal memory usage while preserving detection precision to effectively address performance vs. accuracy trade-offs. These optimization strategies collectively ensure that real-time object detection in autonomous driving systems can function effectively even on resource-constrained embedded platforms.

Improving Detection Accuracy in Complex Autonomous Driving Scenarios

Several authors have suggested novel algorithms and approaches to strike the efficiency-complexity and accuracy-efficiency balance in relation to object detection in complex autonomous driving scenarios. Detected objects in complex driving scenarios, including such challenging settings as urban streets, nighttime, and adverse weather conditions, include pedestrians, bicycles, and other vehicles. [19] stressed that the accuracy and robustness of object detectors are degraded by harsh environments, thus necessitating multi-modal approaches that combine LiDAR and camera data to improve detection. [20] added that real-time performance is especially difficult to achieve at the cost of high accuracy on resource-constrained embedded devices.

In light of such scenarios, a plethora of solutions has been suggested to overcome the challenge. The demand for light-weight models to gain precision is constantly on the rise in such adverse scenarios. Recently, Zhou et al. introduced a deformational convolution along with spatial attention into Faster-RCNN, increasing small object detection accuracy in the urban driving scene [21]. Analogously, [5] presented YOLOv4-5D which adds the layer of multi-scale detection to further enhance accuracy at the same detection speed. Transfer learning has also proven valuable for adapting lightweight models to specific environmental conditions. [22] evaluated the benefits of adding temporal data and domain-specific datasets: it significantly improved detection accuracy in adverse conditions such as nighttime or rainy weather.

Ensemble methods have also been exploited to improve performance on complex detection tasks. Many combinations of YOLO and Faster-RCNN have been made; the results were found to have an improvement in both segmentation and classification accuracies. [23] noted a 7 percent accuracy improvement in detection when the ensemble method was used. This was highly effective on occluded objects in dense urban environments, according to [23]. This way, based on a combination of these aspects, the researchers are moving to the realization of object detection systems that accurately and efficiently deal with complex and dynamic driving scenarios.

Discussion and Insights: Optimizing Lightweight Object Detection for Autonomous Vehicles

The present paper discusses optimization strategies for lightweight object detection models that can be deployed in autonomous driving to improve efficiency and performance on embedded platforms, such as the Nvidia Jetson and Raspberry Pi. Three areas of focus have been identified for this discussion: model optimization, enhancement of real-time speed, and improvement of accuracy in complex environments.

Table 1 and table 2 summarized the key points from above discussion which shows channel pruning and quantization are two of the most important optimizing techniques that decrease the size of the model and complexity in computation. According to [7], [8], [9], channel pruning removes the filters which are less important in order to preserve the essential feature and not reduce the detection accuracy. This, coupled with quantization [10], [11], makes the models more efficient, allowing them to infer more quickly and to consume less memory, a prerequisite for real-time processing on embedded devices.

Table 1: Summary of Key Papers Referenced

Paper	Technique	Key Findings
[7]	Channel pruning, localization-aware pruning	Improved accuracy by retaining essential features for both classification and localization.
[8]	Soft-gated pruning, pruning during training	Reduced parameters and faster inference with no accuracy loss.
[9]	Knowledge distillation-based pruning	Maintained detection accuracy with smaller models.
[10]	Quantization-aware training (QAT)	Enhanced accuracy retention during quantization.
[11]	GradQuant, clipping-free quantization	Prevented accuracy loss by avoiding clipping of outlier activations.
[12]	Combining pruning and quantization using reinforcement learning	Achieved optimal model compression without accuracy degradation.
[13]	QAT and filter pruning integration	Improved performance on non-GPU devices.
[15]	ALODAD, lightweight anchor-free detector with GhostNet	Reduced computational cost while maintaining accuracy and real-time performance.
[14]	MobileYOLO, FPS optimization for real-time performance	Enhanced FPS rates while preserving accuracy.
[16]	YOLOGP, efficient upsampling and compression for YOLOv5	Achieved faster inference with minimal memory usage.
[18]	Frame prioritization and early exit strategies for FPS	Improved FPS on Jetson TX2 while maintaining detection capabilities.

Table 2: Summary of Key Papers Referenced

Paper	Technique	Key Findings
[23]	Low-latency models for real-time detection in traffic	Efficient detection in dynamic traffic scenarios.
[19]	Multi-modal fusion (LiDAR + camera) for harsh environments	Improved robustness and accuracy in complex scenarios.
[22]	Transfer learning with temporal and domain-specific datasets	Improved detection in adverse conditions like nighttime.
[23]	Ensemble methods combining YOLO and Faster-RCNN	Enhanced accuracy in detecting occluded objects in urban areas.
[6]	Challenges in real-time detection in adverse conditions	Real-time detection, complex environments
[21]	Enhancing Faster-RCNN for small-object detection	Deformable convolution, spatial attention mechanisms
[22]	Transfer learning for domain-specific detection	Transfer learning, domain adaptation
[23]	YOLO and Faster-RCNN ensemble for improved detection accuracy	Ensemble methods, occlusion detection

Real-time detection speeds should be further improved. Balanced among being either fast or accurate, Light architecture is YOLO-tiny, MobileNetV2, and EfficientDet as presented by [14], [15]. Hardware acceleration and pipeline optimization strategies, early exit strategies [18] and adaptive inference allow the systems to achieve necessary FPS rates while staying performant for the application of autonomous driving. Besides, some methods such as YOLOGP by [16] even illustrate that sufficient upsampling can improve detection speed.

To develop the detection accuracy in complex scenes, such as urban streets and adverse weather conditions, the methods of multi-modal data integration by [19] and transfer learning by [22] have exhibited great potential. This ensures that object models are ready for dynamic, challenging environments. Overall, optimizing lightweight object detection models ensures the feasibility and safety of autonomous systems, providing a foundation for future research and advancements in real-time detection technologies.

CONCLUSION AND FUTURE DIRECTIONS

This paper discusses different optimization techniques for lightweight object detection models in autonomous driving, including methods such as channel pruning, quantization, and real-time speed enhancements. Improving computational efficiency without sacrificing accuracy, these techniques contribute significantly to the deployment of object detection models on resource-constrained embedded platforms like Nvidia Jetson and Raspberry Pi. Besides, strategies to improve the accuracy of detection in complex scenarios, for instance, adverse weather and urban environments are also discussed. This encompasses workable solutions in real life. Future work shall be fine tuning these methods and performance testing them under some real-world scenarios. For instance, more advanced pruning techniques, such as dynamic pruning and clipping-free quantization (Deng et al., 2023), can further optimize the efficiency of the model. Optimizing architectures and incorporating new AI accelerators will enhance the speed of real-time detection, thus improving system responsiveness. Further exploration of multi-modal fusion and transfer learning can also enhance detection accuracy in challenging environments. Evaluating innovations against the existing models in diverse scenarios will help develop more robust object detection systems in autonomous vehicles.

REFERENCES

1. F. Hawlader, F. Robinet, and R. Frank, “Vehicle-to-infrastructure communication for real-time object detection in autonomous driving,” in 2023 18th Wireless On-Demand Network Systems and Services Conference (WONS), 2023, pp. 40–46.

2. A. Hannan Khan, S. T. R. Rizvi, and A. Dengel, “Real-time Traffic Object Detection for Autonomous Driving,” arXiv e-prints, p. arXiv–2402, 2024.
3. J. Choi, D. Chun, H.-J. Lee, and H. Kim, “Uncertainty-based object detector for autonomous driving embedded platforms,” in 2020 2nd IEEE international conference on artificial intelligence circuits and systems (AICAS), 2020, pp. 16–20.
4. J. Yang, C. Wang, H. Wang, and Q. Li, “A RGB-D based real-time multiple object detection and ranging system for autonomous driving,” IEEE Sens J, vol. 20, no. 20, pp. 11959–11966, 2020.
5. Y. Cai et al., “YOLOv4-5D: An effective and efficient object detector for autonomous driving,” IEEE Trans Instrum Meas, vol. 70, pp. 1–13, 2021.
6. S. Heo, S. Jeong, and H. Kim, “Rtscale: Sensitivity-aware adaptive image scaling for real-time object detection,” in 34th Euromicro Conference on Real-Time Systems, 2022, p. 2.
7. Z. Xie, L. Zhu, L. Zhao, B. Tao, L. Liu, and W. Tao, “Localization-aware channel pruning for object detection,” Neurocomputing, vol. 403, pp. 400–408, 2020.
8. Z. Miao, Y. Zhang, W. Li, and R. Chen, “Acceleration of Infrared Target Detection via Efficient Channel Pruning,” in Journal of Physics: Conference Series, 2022, p. 12025.
9. X. Zhao and H. Zheng, “A Lightweight Detection Model for Typical Objects in Streetscape Dataset,” in 2021 China Automation Congress (CAC), 2021, pp. 548–553.
10. K. Choi, S. M. Wi, H. G. Jung, and J. K. Suhr, “Simplification of deep neural network-based object detector for real-time edge computing,” Sensors, vol. 23, no. 7, p. 3777, 2023.
11. C. Deng, Z. Deng, Y. Han, D. Jing, and H. Zhang, “GradQuant: Low-loss Quantization for Remote Sensing Object Detection,” IEEE Geoscience and Remote Sensing Letters, 2023.
12. W. Tang, X. Wei, and B. Li, “Automated model compression by jointly applied pruning and quantization,” arXiv preprint arXiv:2011.06231, 2020.
13. G. Idama, Y. Guo, and W. Yu, “QATFP-YOLO: Optimizing Object Detection on Non-GPU Devices with YOLO Using Quantization-Aware Training and Filter Pruning,” in 2024 33rd International Conference on Computer Communications and Networks (ICCCN), 2024, pp. 1–6.
14. K. Wang, T. Zhou, X. Li, and F. Ren, “Performance and challenges of 3D object detection methods in complex scenes for autonomous driving,” IEEE Transactions on Intelligent Vehicles, vol. 8, no. 2, pp. 1699–1716, 2022.
15. T. Liang, H. Bao, W. Pan, and F. Pan, “ALODAD: An anchor-free lightweight object detector for autonomous driving,” IEEE Access, vol. 10, pp. 40701–40714, 2022.
16. J. Wang and W. Lin, “YOLOGP: A YOLOv5-Based Lightweight Network for Efficient Vehicle Detection in Autonomous Driving Scenarios,” in Proceedings of the 2nd International Conference on Signal Processing, Computer Networks and Communications, 2023, pp. 98–105.
17. W. J. Yun, S. Park, J. Kim, and D. Mohaisen, “Self-configurable stabilized real-time detection learning for autonomous driving applications,” IEEE Transactions on Intelligent Transportation Systems, vol. 24, no. 1, pp. 885–890, 2022.
18. S.-Y. Jhong, Y.-Y. Chen, C.-H. Hsia, S.-C. Lin, K.-H. Hsu, and C.-F. Lai, “Nighttime object detection system with lightweight deep network for internet of vehicles,” J Real Time Image Process, vol. 18, no. 4, pp. 1141–1155, 2021.
19. S. Wang and M. Chen, “A lidar multi-object detection algorithm for autonomous driving,” Applied Sciences, vol. 13, no. 23, p. 12747, 2023.
20. Y. Kim, H. Hwang, and J. Shin, “Robust object detection under harsh autonomous-driving environments,” IET Image Process, vol. 16, no. 4, pp. 958–971, 2022.
21. Y. Zhou, S. Wen, D. Wang, J. Mu, and I. Richard, “Object detection in autonomous driving scenarios based on an improved faster-RCNN,” Applied Sciences, vol. 11, no. 24, p. 11630, 2021.
22. L. Huang, Y. Zeng, S. Wang, R. Wen, and X. Huang, “Temporal Based Multi-Sensor Fusion for 3D Perception in Automated Driving System,” IEEE Access, 2024.
23. S. A. Khan, H. J. Lee, and H. Lim, “Enhancing object detection in self-driving cars using a hybrid approach,” Electronics (Basel), vol. 12, no. 13, p. 2768, 2023.