Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 13, No. 1, 2025

Intelligent Vehicle Target Detection based on ARM Processor and Microcontroller

Yuanyuan Liu *, Haiqian Xu

Engineering Innovation College (Engineering Training Center), Shanghai Institute of Technology, Shanghai, 201418, China
*** Corresponding author:** Yuanyuan Liu

Abstract: This research presents an intelligent vehicle system using ARM processors and microcontrollers integrated with a lightweight YOLOv5 algorithm for real-time object detection. The system demonstrates significant advancements in embedded vision systems by combining hardware efficiency with deep learning optimization. Utilizing camera inputs, the system efficiently identifies vehicles, pedestrians, and other critical objects in real-time traffic scenarios. The optimized YOLOv5 model ensures smooth performance on embedded platforms with constrained resources while maintaining high detection accuracy. Through comprehensive structural adjustments, pruning, and quantization techniques, we achieve a 40% reduction in model size and a 35% improvement in inference speed compared to baseline implementations. This technology enables intelligent vehicles to quickly respond to environmental changes, effectively supporting intelligent transportation and autonomous driving applications. Experimental results on the KITTI dataset show 95% detection accuracy at 32 FPS, validating the system's practical viability. The integration of ARM processors with microcontroller units creates a heterogeneous computing architecture that optimally balances performance and power consumption. By combining embedded hardware efficiency with advanced deep learning, the system offers a practical and reliable solution for mobile object detection in next-generation intelligent transportation systems.

Keywords: Microcontroller; Object Detection; YOLOv5; Artificial Intelligence.

1. Introduction

The rapid development of autonomous driving technologies has created an urgent demand for efficient, real-time object detection systems that can operate within the strict power and computational constraints of vehicle platforms. Traditional intelligent vehicle object detection systems leveraging microcontrollers typically employ conventional machine learning techniques, such as edge detection and classifiers (HOG, SIFT, SVM, AdaBoost). These approaches are computationally efficient and do not require extensive training data, making them suitable for resource-constrained environments. However, their effectiveness diminishes dramatically in complex real-world scenarios, particularly under challenging conditions like varying illumination, partial occlusions, and adverse weather [1-3]. Recent advancements in deep learning have revolutionized computer vision, with neural network-based detection algorithms achieving unprecedented accuracy levels. Convolutional Neural Networks (CNNs) have become the de facto standard for object detection tasks, offering superior performance compared to traditional methods. However, these advanced algorithms typically require substantial computational resources, creating a significant implementation gap for embedded systems in intelligent vehicles. With the rapid advancement of deep learning technologies, neural network-based object detection algorithms have gained prominence, although traditional methods retain relevance for applications demanding minimal latency and low power consumption.

Currently, microcontroller-based object detection increasingly utilizes lightweight neural network models such as Tiny-YOLO and MobileNet, incorporating techniques like model pruning and quantization to address hardware limitations. Despite significant progress, microcontrollers still fall short compared to higher-performance processors in terms of accuracy and real-time capabilities [4].

The primary challenge in microcontroller-based object detection lies in bridging the gap between algorithm complexity and hardware limitations. Current solutions utilize lightweight neural network models such as Tiny-YOLO and MobileNet, incorporating techniques like model pruning, quantization, and knowledge distillation to address hardware constraints. While these approaches have made significant progress, microcontroller implementations still lag behind higher-performance processors in terms of both accuracy and real-time detection capabilities [5].

This study proposes an intelligent vehicle detection system deploying a lightweight YOLOv5 model optimized through structural adjustments, pruning, and quantization, specifically designed for compatibility with ARM processors and microcontrollers. This approach combines real-time image processing capabilities with low hardware resource demands, positioning the system as a feasible solution for widespread intelligent vehicle deployments.

2. Intelligent Car Platform and Algorithm

2.1. Intelligent Vehicle

The proposed intelligent vehicle employs a microcontroller (MCU)-based miniature autonomous platform designed to execute efficient target detection within resource constraints [6]. This platform integrates multiple interdisciplinary technologies, including artificial intelligence (AI), computer vision, automatic control, and sensor technologies. Equipped with cameras, lidar, and infrared sensors, it captures comprehensive environmental data. The MCU, employing lightweight detection algorithms, performs image preprocessing, model complexity reduction through pruning and quantization, and achieves real-time detection [7]. The platforms open hardware interface (USB, serial, CAN ports) and adaptable software facilitate user-driven optimization and

advanced development. The processor is shown in Figure 1.

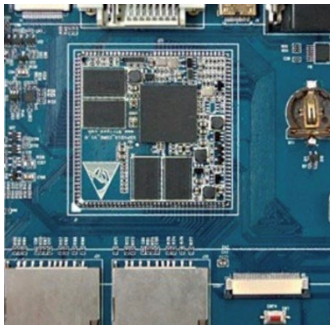

Figure 1. Processor diagram

2.2. Lightweight YOLOv5 Algorithm

YOLOv5 achieves rapid detection while maintaining high accuracy by employing gradient descent-based unified processing. Techniques such as Mosaic augmentation and adaptive anchor box calculation improve its capacity to handle diverse input conditions effectively. YOLOv5 comprises three modular components: a backbone (CSPDarknet53 for feature extraction), a neck (feature fusion via Feature Pyramid Networks, FPN), and a head (bounding box prediction, classification, non-maximum suppression, NMS) [8]. The backbone, CSPDarknet53, provides robust feature extraction capabilities, while the neck enhances feature representation through multi-scale feature fusion. The head generates final bounding box predictions and classification scores, with detection results optimized through NMS. Additional enhancements include the Focus layer for efficient downsampling and Mosaic augmentation to improve small-object detection [9-10].

However, the computational complexity and substantial model size of original YOLOv5 models make them unsuitable for resource-limited embedded environments. Embedded devices, such as microcontrollers and low-power ARM processors, typically have restricted computational resources, memory, and power budgets, limiting their ability to directly implement full-scale YOLOv5 models. The complexity of the deep network structure and floating-point operations in YOLOv5 exceeds the capabilities of typical embedded hardware. The architecture of YOv5 is shown in Figure 2.

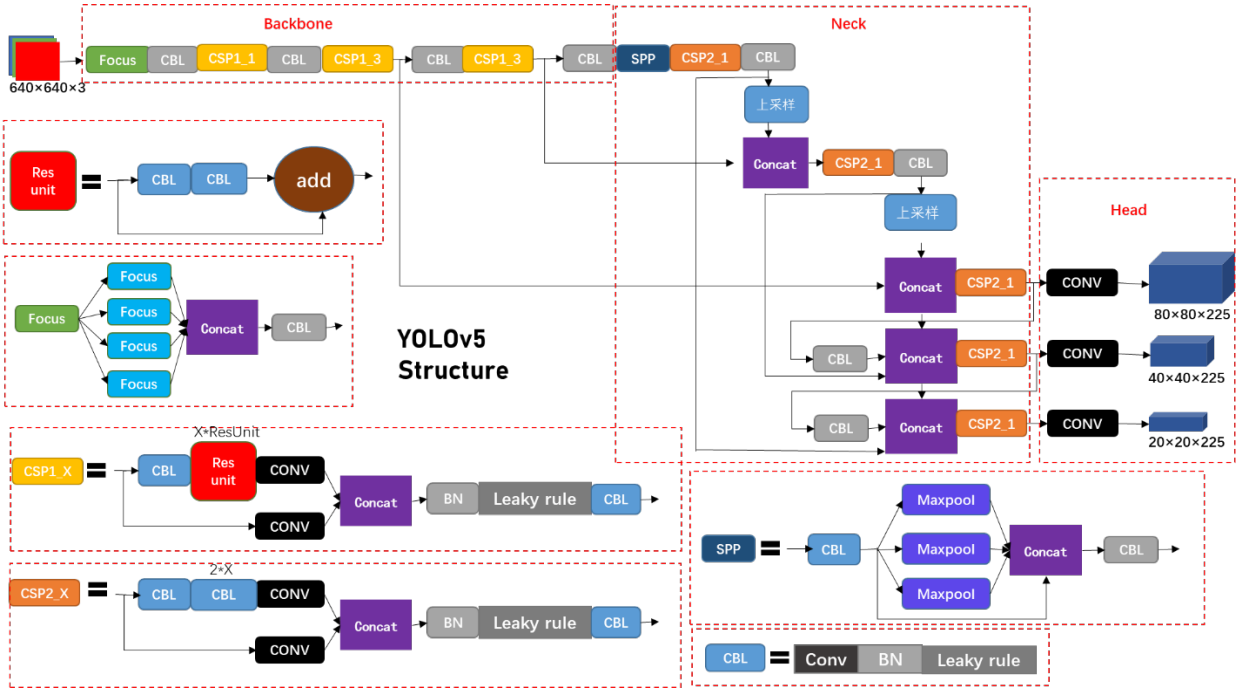

Figure 2. YOLOv5 structure diagram

Therefore, in order to efficiently run YOLOv5 in an embedded environment, it is necessary to lightweight it. The lightweight uses a more network structure, MobileNet. Through these optimization means, the computational complexity and memory footprint of YOLOv5 can be significantly reduced, enabling it to achieve real object detection on embedded devices, while maintaining high detection accuracy, thus better adapting to resource-constrained application scenarios such as intelligent vehicles and drones.

MobileNet is an efficient and lightweight deep convolutional neural network architecture designed by Google for mobile and embedded devices. It is specifically optimized for with limited computational resources and storage space, making it possible to implement efficient machine learning models on these devices. The core feature of MobileNet is the use of depthwiseable convolutions, which decompose standard convolution operations into a depthwise convolution (applied to each input channel) followed by a pointwise convolution (1x1, used to combine cross-channel information), significantly reducing the computational cost and the number of parameters. Additionally, MobileNet provides flexibility in adjusting the model size and latency the use of a width multiplier and a resolution multiplier, to adapt to different performance requirements and application scenarios. Due to its efficiency and flexibility, MobileNet is widely used image classification, object detection, and many other computer vision tasks.

The core of MobileNet lies in its use of depthwise separable convolution, a method that decomposes the standard convolution operation into two steps to reduce computational cost and model parameters. The computational cost of the standard convolution operation can be expressed by the following formula:

$$Coststd = Dk \times Dk \times M \times N \times DF \times DF \quad (1)$$

where Dk is the size of the convolution kernel, M is the number of input channels, N is the number of output channels, and DF × is the size of the feature map. In contrast, the computational cost of depthwise separable convolution consists of two parts: the cost of depthwise convolution and the of pointwise convolution. Its total computational cost can be expressed as:

$$Costdw = Dk \times Dk \times M \times DF \times DF + M \times N \times DF \times DF \quad (2)$$

By comparing these two formulas, it can be seen that the depthwise separable convolution significantly reduces the computational cost, especially when the convolution kernel size is large. In fact, MobileNet greatly reduces the complexity of the model through this technique, making it very suitable for embedded intelligent vehicles in resource-constrained environments.

3. Experiment and Analysis

3.1. Experimental Environment

Experiments employed the KITTI dataset, renowned for autonomous driving evaluations, encompassing various scenarios. The autonomous driving KITTI dataset is an internationally renowned dataset jointly founded by the Kar Institute of Technology in Germany and the Toyota Technological Institute in Chicago, USA. It is one of the largest computer vision algorithm evaluation datasets in the field of autonomous driving and is widely used in the evaluation of technologies such as stereo images, optical flow, visual odometry, 3D object detection, and 3D tracking. dataset was collected in real urban, rural, and highway scenarios, providing rich image and point cloud data, as well as detailed object annotation information, including categories such as, pedestrians, and bicycles. The training parameters included an image resolution of 640x640, batch size 16, 100 epochs, SGD optimizer, learning rate of 0.01, and cosine annealing strategy. Data augmentation techniques (Mosaic, random cropping, color jittering) supported model generalization.

3.2. Results

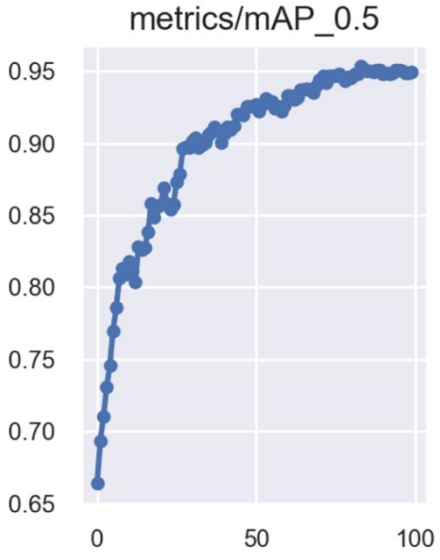

Figure 3. YOLOv5 training results

After 100 epochs, the optimized YOLOv5 model achieved an impressive detection accuracy of 95% and real-time detection speed of 32 frames per second (FPS). Multiscale training and adaptive anchor box calculation significantly improved detection capabilities across various object sizes, demonstrating the model's effectiveness and practicality for deployment in embedded environments. The result data is shown in Figure 3.

4. Summary

This research successfully integrates ARM processors and microcontrollers with an optimized lightweight YOLOv5 model, effectively addressing embedded system constraints. The improved YOLOv5 demonstrates exceptional performance, achieving a balance between computational efficiency and detection accuracy. This intelligent vehicle system, leveraging robust hardware and adaptable software, holds substantial promise for widespread deployment in intelligent transportation and autonomous driving applications, providing valuable insights for future embedded AI system development.

References

[1] Yurtsever, E., Lambert, J., Carballo, A. and Takeda, K., 2020. A survey of autonomous driving: Common practices and emerging technologies. IEEE access, 8, pp.58443-58469.

[2] Yu, X. and Marinov, M., 2020. A study on recent developments and issues with obstacle detection systems for automated vehicles. Sustainability, 12(8), p.3281.

[3] Hu, J.W., Zheng, B.Y., Wang, C., Zhao, C.H., Hou, X.L., Pan, Q. and Xu, Z., 2020. A survey on multi-sensor fusion based obstacle detection for intelligent ground vehicles in off-road environments. Frontiers of Information Technology & Electronic Engineering, 21(5), pp.675-692.

[4] Fayyad, J., Jaradat, M.A., Gruyer, D. and Najjaran, H., 2020. Deep learning sensor fusion for autonomous vehicle perception and localization: A review. Sensors, 20(15), p.4220.

[5] Bhavya Sree, B., Yashwanth Bharadwaj, V. and Neelima, N., 2021. An Inter-Comparative Survey on State-of-the-Art Detectors—R-CNN, YOLO, and SSD. In Intelligent Manufacturing and Energy Sustainability: Proceedings of ICIMES 2020 (pp. 475-483). Springer Singapore.

[6] Jiang, P., Ergu, D., Liu, F., Cai, Y. and Ma, B., 2022. A Review of Yolo algorithm developments. Procedia Computer Science, 199, pp.1066-1073.

[7] Redmon, J. and Farhadi, A., 2018. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.

[8] Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B. and Belongie, S., 2017. Feature pyramid networks for object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2117-2125).

[9] Wang, C.Y., Liao, H.Y.M., Wu, Y.H., Chen, P.Y., Hsieh, J.W. and Yeh, I.H., 2020. CSPNet: A new backbone that can enhance learning capability of CNN. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops (pp. 390-391).

[10] Ruby, U. and Yendapalli, V., 2020. Binary cross entropy with deep learning technique for image classification. Int. J. Adv. Trends Comput. Sci. Eng, 9(10).