Bulletin of Networking, Computing, Systems, and Software – www.bncss.org, ISSN 2186-5140
Volume 15, Number 1, page 35, January 2026

Evaluating the Efficiency of Hailo-8 for YOLOv10-Based Object Detection in Edge Environments

Phyo Min Myat[*], Yasunori Osana[†]

Abstract

This work will examine the real-time performance of object detection based on the Hailo-8 AI accelerators, a low-power inference developed for edge computing. The system performance was tested in terms of system throughput, system latency, and CPU load performance metrics in an Intel Core i9-10900K system running Ubuntu 22.04. On a YOLOv10 model with 80 classes of objects in the COCO dataset, an average of 12.48 FPS with a total system latency of 57.39 ms per frame was recorded, with 56.40 ms spent in inference and 0.98 ms for CPU processing. This clearly indicates that Hailo-8 significantly outperforms CPU-only inference with low power consumption performance.

Keywords: Object Detection, Hailo-8, YOLOv10, Edge Computing, COCO dataset

1 Introduction

Growing IoT technology demands real-time object detection at the network edge environment. On the one hand, edge inference offers lower latency, improved privacy, and reduced bandwidth. On the other hand, power and processing capabilities are limited. The Hailo-8 accelerator [1] hailo-8 is a neural processing unit specifically designed for deep learning inference, delivering high performance while conserving power. This paper examines the capabilities of the Hailo-8 when executing YOLOv10, focusing on performance metrics such as system-level throughputs and latencies, as well as CPU activity.

2 Motivation And Related Work

Edge-based deep learning enables autonomous, privacy-preserving decision-making near data sources. Platforms such as Google Coral TPU, Intel Movidius, and NVIDIA Jetson Nano have demonstrated this capability. Studies using YOLO models show that CPU-only processing is too slow for real-time, while GPU acceleration improves speed but increases power consumption. The Hailo-8, a multi-core neural processor, offers efficient convolutional performance; however, few studies have evaluated it with modern models like YOLOv10 [3]. This research addresses that gap through an experimental performance analysis on an edge computing system.

[*]Graduate School of Science and Technology, Kumamoto University, Kumamoto, Japan. phyo@lut.hps.cs.kumamoto-u.ac.jp

[†]Research and Education Institute for Semiconductors and Informatics, Kumamoto University, Kumamoto, Japan. osana@kumamoto-u.ac.jp

3 Experimental Setup and Result Discussion

The experiment was conducted on an Intel Core i9-10900K host with a Hailo-8 M.2 AI accelerator running Ubuntu 22.04. Runtime control used the HailoRT C++ API [2], while OpenCV handled image preprocessing and Non-Maximum Suppression (NMS). The YOLOv10 model, trained on the COCO dataset with 80 classes, was tested on 393 video frames resized to 640x640 pixels, with confidence and NMS thresholds of 0.5 and 0.7. The system achieved 12.48 FPS with a total latency of 57.39 ms, including 56.40 ms on Hailo-8 and 0.98 ms on the CPU. Inference accounted for about 98% of total latency, and CPU-only inference (< 2 FPS) was significantly slower, achieving a 6-8x speedup with low energy consumption on Hailo-8. Minor CPU-side bottlenecks remain in multithreading, asynchronous streaming, and DMA-based data transfer.

4 Conclution and Future Work

This work has also validated that the Hailo-8 provides a suitable platform for real-time object detection, balancing accuracy, processing speed, and power consumption with YOLOv10. Inference in Hailo-8 is highly optimized, but host-side overheads are a performance bottleneck. Future research will focus on software advances, such as concurrent data streaming and handling, as well as power consumption assessments and integration with FPGA-SoC-based systems. Since our host-side software is written in C++, optimization at the lower level on FPGA-SoC-based systems is possible.

References

[1] Hailo Technologies, "Hailo-8 AI Processor for Edge Devices." 2024. [Online]. `https://hailo.ai/hailo-8/`

[2] Hailo Technologies, "HailoRT Software Suite Documentation." 2024. [Online]. `https://hailo.ai/developer-zone/software-downloads`

[3] Y. Zheng, J. Zhang, Z. Wang, and J. Gao, "YOLOv10: Real-Time End-to-End Object Detection." *arXiv preprint arXiv:2405.14458*, 2024. [Online]. `https://arxiv.org/abs/2405.14458`