Hardware Acceleration Design for Object Detector in Semiconductor Design Hackathon

°Yuxuan Pan[1], Atsutake Kosuge[1], Hideharu Amano[1] (1.Univ. Tokyo)

E-mail: pan@g.ecc.u-tokyo.ac.jp

Object detection is essential across various fields, yet achieving real-time performance on hardware-accelerated platforms remains challenging. During the AI and Semiconductor Design Hackathon 2024 at the University of Tokyo, we explored FPGA-based acceleration for object detector inference. By accepting minor accuracy tradeoffs, our approach achieved 5x to 39x speed improvements, enabling real-time processing.

We implemented a YOLOv3 [1] object detection system on the Xilinx KV260 FPGA development board, utilizing the Vitis-AI framework and DPU (Deep Learning Processor Unit) for hardware acceleration. The workflow included training the YOLOv3 mode, converting and quantizing the model, and deploying it to the FPGA board's programmable logic (PL). To enhance efficiency, we adopted the slighter YOLOv3-tiny [2] architecture, reduced input image size, and utilized multiple DPU instances via multi-threading. Inference programs were developed in C++ to run on the processing system (PS).

Performance was evaluated using static images and video inputs, measuring FPS (Frames Per Second) and mAP (mean Average Precision). Table 1 summarizes five configurations balancing mAP and FPS. The baseline YOLOv3 achieved 45% mAP but low FPS, unsuitable for real-time applications. In contrast, optimized configurations demonstrated significant improvements. Configuration #1 achieved 47% mAP at 9 FPS, while configuration #5 prioritized speed, reaching 101 FPS with lower mAP. Intermediate setups offered versatile tradeoffs.

Efforts to fine-tune the model to recover accuracy post-resizing and quantization, along with experiments on newer architectures like YOLOv10-n [3], were initiated but not fully completed due to time constraints. These directions hold the potential for further enhancing the balance between accuracy and efficiency in future work.

REFERENCES

[1] A. Farhadi et al., Proc. CVPR., vol. 1804, pp. 1-6, 2018.
[2] P. Adarshe et al., Proc. ICACCS, pp. 687-694, 2020.
[3] A. Wang et al., *arXiv preprint arXiv: 2405.14458*, 2024.

Table 1 Experimental results.

Index	**Baseline**	**Proposed**				
		#1	**#2**	**#3**	**#4**	**#5**
Model	YOLOv3	YOLOv3	YOLOv3	YOLOv3-tiny	YOLOv3-tiny	YOLOv3-tiny
Input Size	608×608	608×608	320×320	416×416	224×320	224×224
mAP	45%	47%	38%	24%	14%	12%
FPS	2.6	9	18	50	82	101