Object Detection with Embedded Machine Learning

X. Yang; S. Chaudhuri; L. Likforman-Sulem; L. Naviner
LTCI, Télécom Paris, Institut Polytechnique de Paris
xuecan.yang@telecom-paris.fr

Introduction

In this poster I will propose an accelerator of Convolutional Neural Netwotk(CNN) that mixes layers with different precisions. We can choose and combine between 16-bit floating point and binary values for features maps and weights. Taking the well known CNN YOLOv2 as an example we present exploration studies to determine the optimum precision for each layer. As well we present the hardware design in Xilinx PYNQ-Z1 board which is capable of implementing this selective binarization CNN.

Object Detection for Embedded Systems

Fig. 1: Machine learning for embedded systems

The Tiny-YOLO Network[1]

$$FLOP_{network} = 3.48G$$

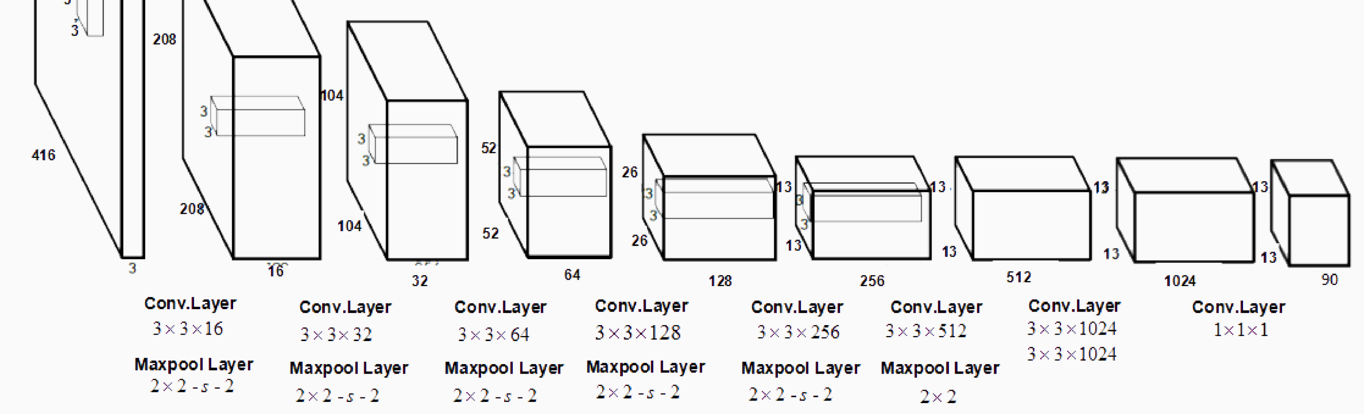


Fig. 2: Tiny-YOLO network

Challenges:

- Capacity of computing (multiplicator etc.)
- Band-width for loading the data

Quantization and Binarization

Quantization:

$$scale = \mathbf{min}(abs(\frac{2^{n-1}-1}{x_{max}}), abs(\frac{-2^{n-1}}{x_{min}}))$$

$$X_q = [scale \cdot X]$$

Dequantization:

$$X = scale^{-1} \cdot X_q$$

Binarization:[2]

$$\text{Filter}^B = sign(\text{Filter}) \qquad \text{Conv} \approx \text{Input} \otimes (\alpha^B \cdot \text{Filter}^B)$$

$$\alpha^B = \frac{1}{n}\text{Filter}_{l_1} \qquad = \alpha^B \cdot (\text{Input} \otimes \text{Filter}^B)$$

Comparison

Input ⊗ Weight	Operations	Band-width	Computation	IOU(224x224)
$\mathbb{R}_{32bits} \otimes \mathbb{R}_{32bits}$	x +	1x	Very Slow	46.28%
$\mathbb{I}_{8bits} \otimes \mathbb{I}_{8bits}$	x +	4x	Slow	37.64%
$\mathbb{R}_{16bits} \otimes \mathbb{B}$	+ -	15.68x	Fast	35.00%
$\mathbb{B} \otimes \mathbb{B}$	XNOR Bit-count	27.19x	Very Fast	7.00%

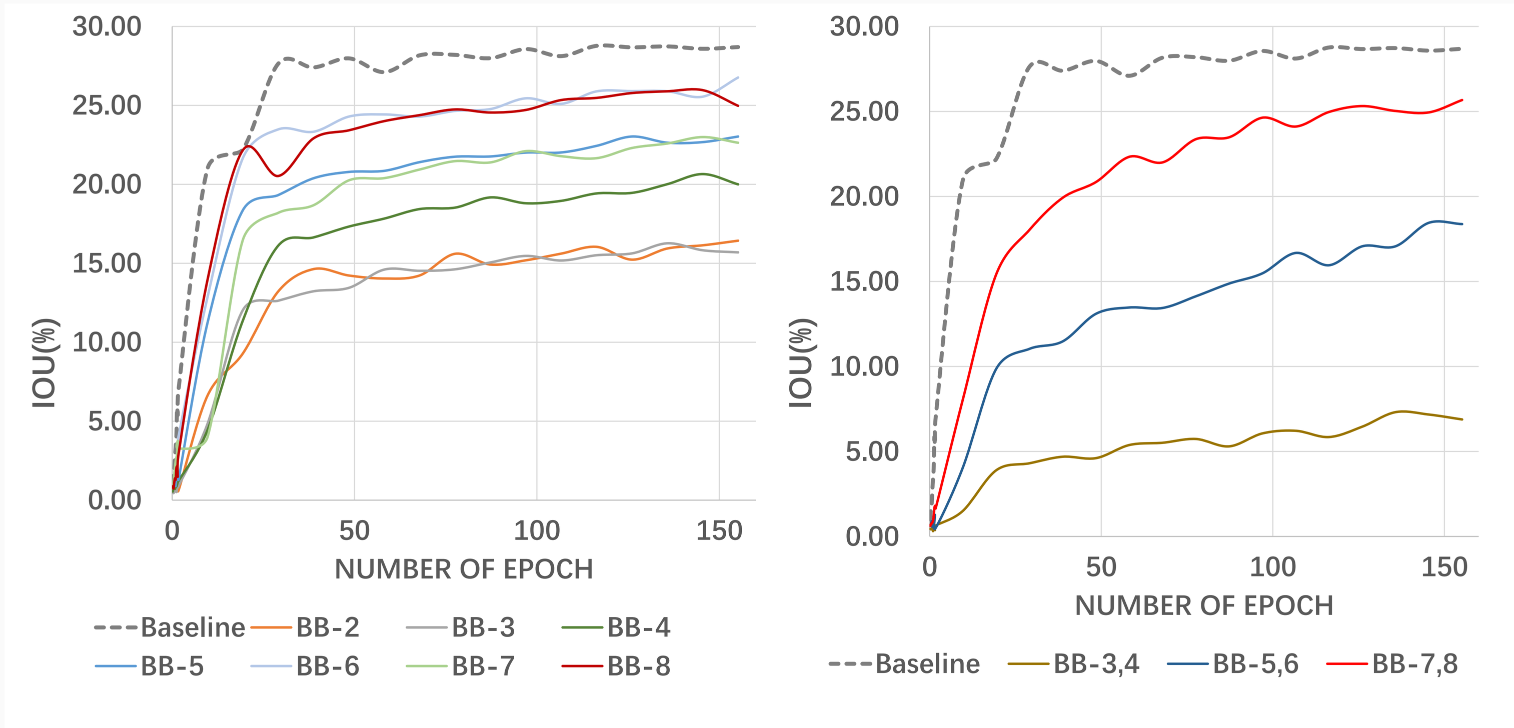
Conclusion We train the tiny YOLO CNN with a drone object detection data-set (DAC-SDC). By using selective binarization method, it is possible to achieve 1.68x improvement in performance incurring a tolerable 8.99% loss in precision measured by IOU.

Our proposed: Selective Binarization

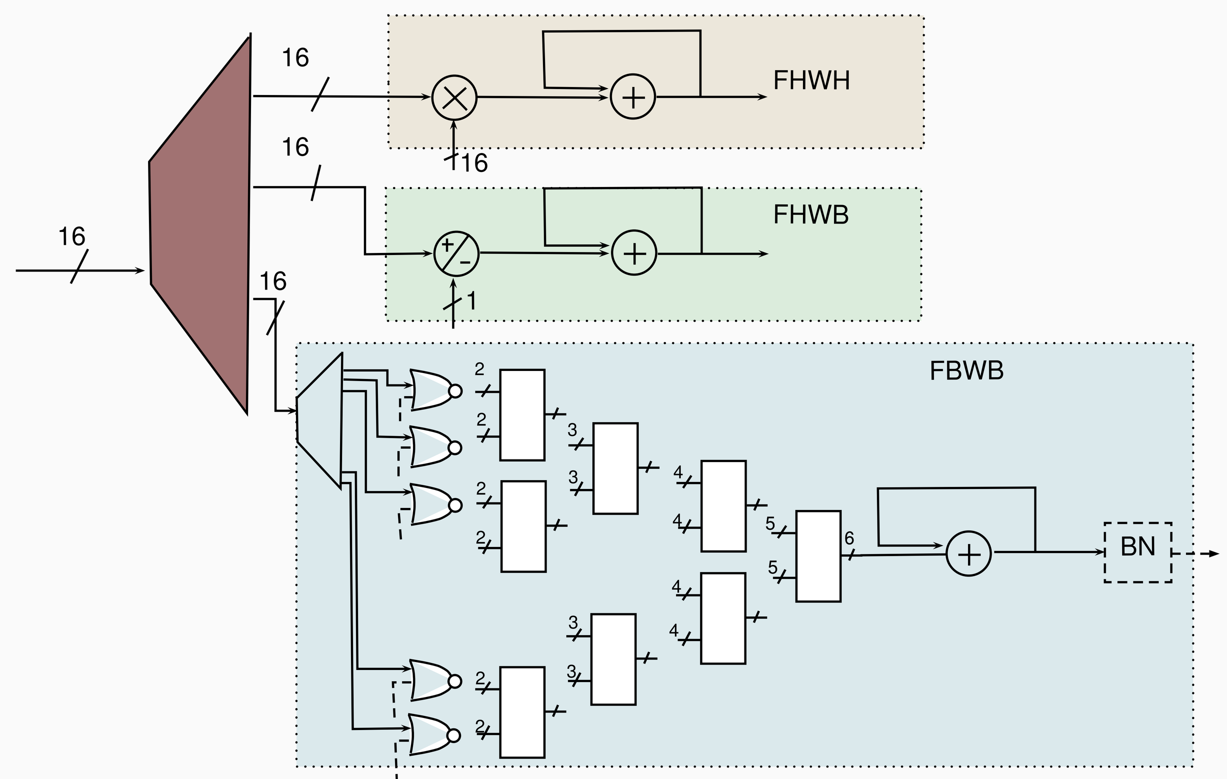
Conv Layer	1	2	3	4	5	6	7	8	9	FLOP(G)	IOU(%)
Baseline	HB	HB	HB	HB	HB	HB	HB	HB	HB	3.48	28.69
BB-1	BB	HB	HB	HB	HB	HB	HB	HB	HB	3.40	X
BB-2	HB	BB	HB	HB	HB	HB	HB	HB	HB	3.28	16.43
BB-3	HB	HB	BB	HB	HB	HB	HB	HB	HB	3.28	15.69
BB-4	HB	HB	HB	BB	HB	HB	HB	HB	HB	3.28	19.99
BB-5	HB	HB	HB	HB	BB	HB	HB	HB	HB	3.28	23.03
BB-6	HB	HB	HB	HB	HB	BB	HB	HB	HB	3.28	26.77
BB-7	HB	HB	HB	HB	HB	HB	BB	HB	HB	2.68	22.63
BB-8	HB	HB	HB	HB	HB	HB	HB	BB	HB	1.88	24.97
BB-1,2	BB	BB	HB	HB	HB	HB	HB	HB	HB	3.21	X
BB-3,4	HB	HB	BB	BB	HB	HB	HB	HB	HB	3.08	6.89
BB-5,6	HB	HB	HB	HB	BB	BB	HB	HB	HB	3.08	18.38
BB-7,8	HB	HB	HB	HB	HB	HB	BB	BB	HB	1.09	25.67

- HB: Layer where the feature maps is half floating point, and binary weights only.
- BB: Layer where both weights and feature maps are binary.
- BB-x : Architecture that x^{th} layer is BB, others are HB.
- BB-x,y: Architecture that x^{th} and y^{th} layer are BB, others are HB.

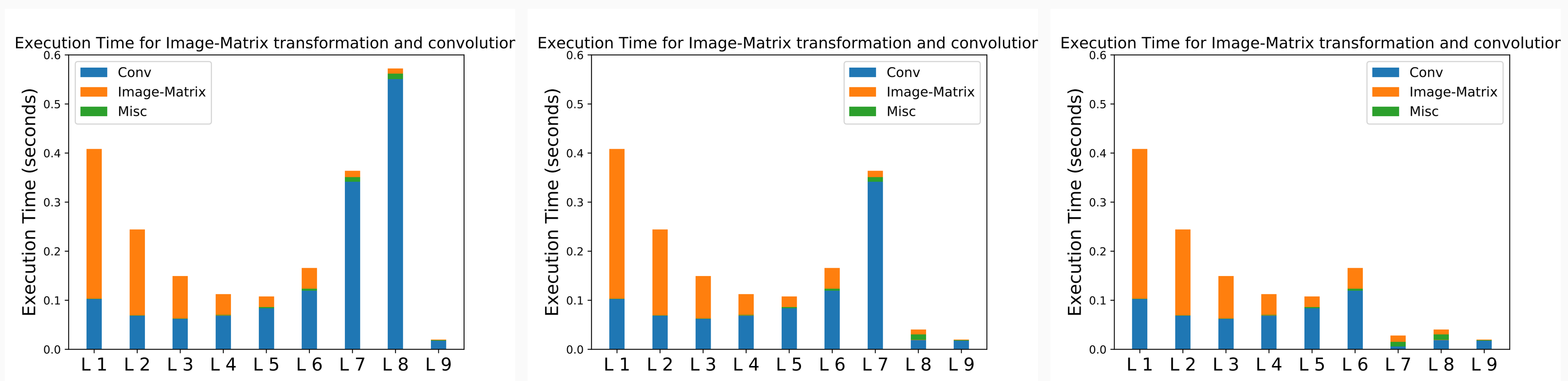
IOU along the training epochs



Compute Unit Implementation



Experiments



Baseline. IOU:28.69%

BB-8. IOU:24.97%

BB-7,8. IOU:25.67%

References

[1] Redmon, J., & Farhadi, A. (2016). YOLO9000: better, faster, stronger. arXiv preprint, 1612.

[2] Rastegari, M., Ordonez, V., Redmon, J., & Farhadi, A. (2016, October). Xnor-net: Imagenet classification using binary convolutional neural networks. In European Conference on Computer Vision (pp. 525-542). Springer, Cham.

[3] Xilinx PYNZ-Z1 *https://www.xilinx.com/products/boards-and-kits/1-hydd4z.html*