# High-Throughput Low-Power Computer Vision Acceleration on FPGAs

Xiaoyin Ma *, Walid Najjar †, and Amit Roy-Chowdhury *
*Electrical Engr. †Computer Sci. & Engr.
University of California, Riverside


## ABSTRACT

The reliance on object or people detection is rapidly growing beyond surveillance to industrial and social applications. Most high accuracy person/ object detection algorithms, such as the Histogram of Oriented Gradients (HOG), achieve very low throughput (< 1 frame/sec) due to their computational complexity. An FPGA implementation of these algorithms does increase the throughput, however the floating-point implementation requires roughly 10x the area of a fixed-point and achieves a clock frequency 3x lower. We have evaluated the fixed-point implementation of HOG on 10,000 benchmark images with *known ground truth* while varying the data bit width. The results show that 13 bits fixed point data achieves as good or better detection accuracy than the reference OpenCV floating-point version.

*The FPGA implementation achieves a 68.7x higher throughput than a high-end CPU, 5.1x higher than a high-end GPU, and 7.8x higher than the same implementation using floating-point on the same FPGA. The energy expenditure, measured in Joules per frame, are 117x lower than a high-end CPU and 24x than a GPU.*


## OBJECTIVES

- Multichannel, real-time object/person detection
- High-precision and consistent detection
- High-throughput and low energy (mobile/untethered)

## HISTOGRAM ORIENTED GRADIENTS (HOG)

- Overview
  - One of the most successful and popular object detection algorithm
  - Developed for pedestrian detection [1]
  - High detection accuracy but slow to compute
  - Use hardware (FPGA) to accelerate computation
- Detection Algorithm
  - Orientation and Magnitudes computed from pixel gradients (dx and dy)
  - Magnitudes binned into cells (8*8 pixels) based on orientation, nine bins per cell (Fig. 1)
  - Grids of overlapping blocks form detection window (Fig.2)
  - Window size 96*48 (11*5 blocks) [2,3]
  - Use densely scanned window for increased detection performance
  - Concatenate all histograms in a window as HOG feature
  - SVM classification for detection

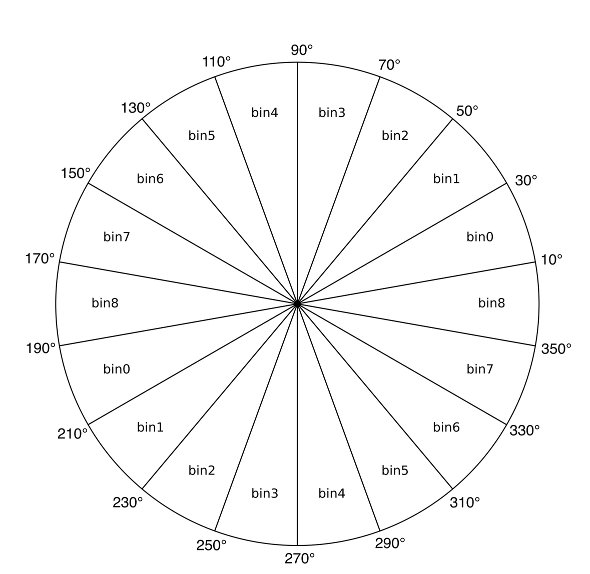



Fig.1 HOG cell binning

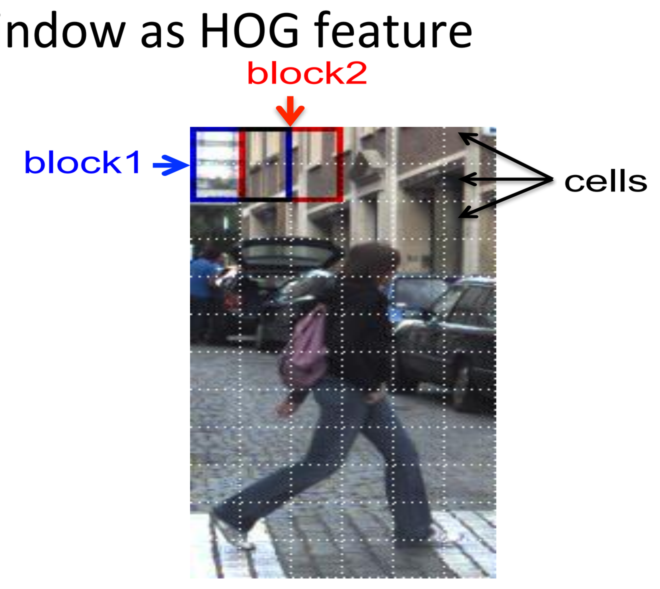



Fig.2 Illustration of HOG cells, blocks and window

## FIXED-POINT DATA

- Perform HOG detection in fixed-point from 27-bits to 11-bits
- ~10,000 images with ground truth from six pedestrian detection benchmark (Tab. 1) [2,4,5,6]
- Apply per-image evaluation methodology [5,8] and PASCAL method [7] for detection performance measurement
- Compare fixed-point detection with floating-point (precision and recall)
- Fix-13: better performance than floating-point (Fig. 3, higher is better)

| | Daimler | Caltech | TUD | BAHNHOF | JELMOLI | SUNNY | TOTAL |
|---|---|---|---|---|---|---|---|
| # image | 2,117 | 5,346 | 237 | 992 | 439 | 354 | 9,485 |
| # object | 2,603 | 8,310 | 661 | 4,459 | 1,834 | 1,783 | 19,650 |

$$\begin{cases} precision = \dfrac{TP}{TP+FP} \\ recall = \dfrac{TP}{TP+FN} \end{cases}$$

TP: true positives
FP: false positives
TN: false negatives

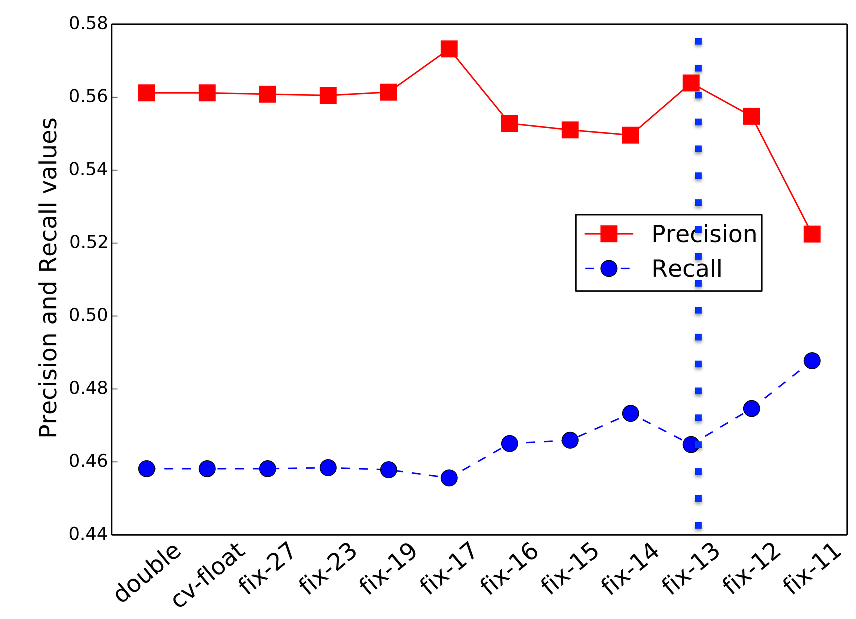
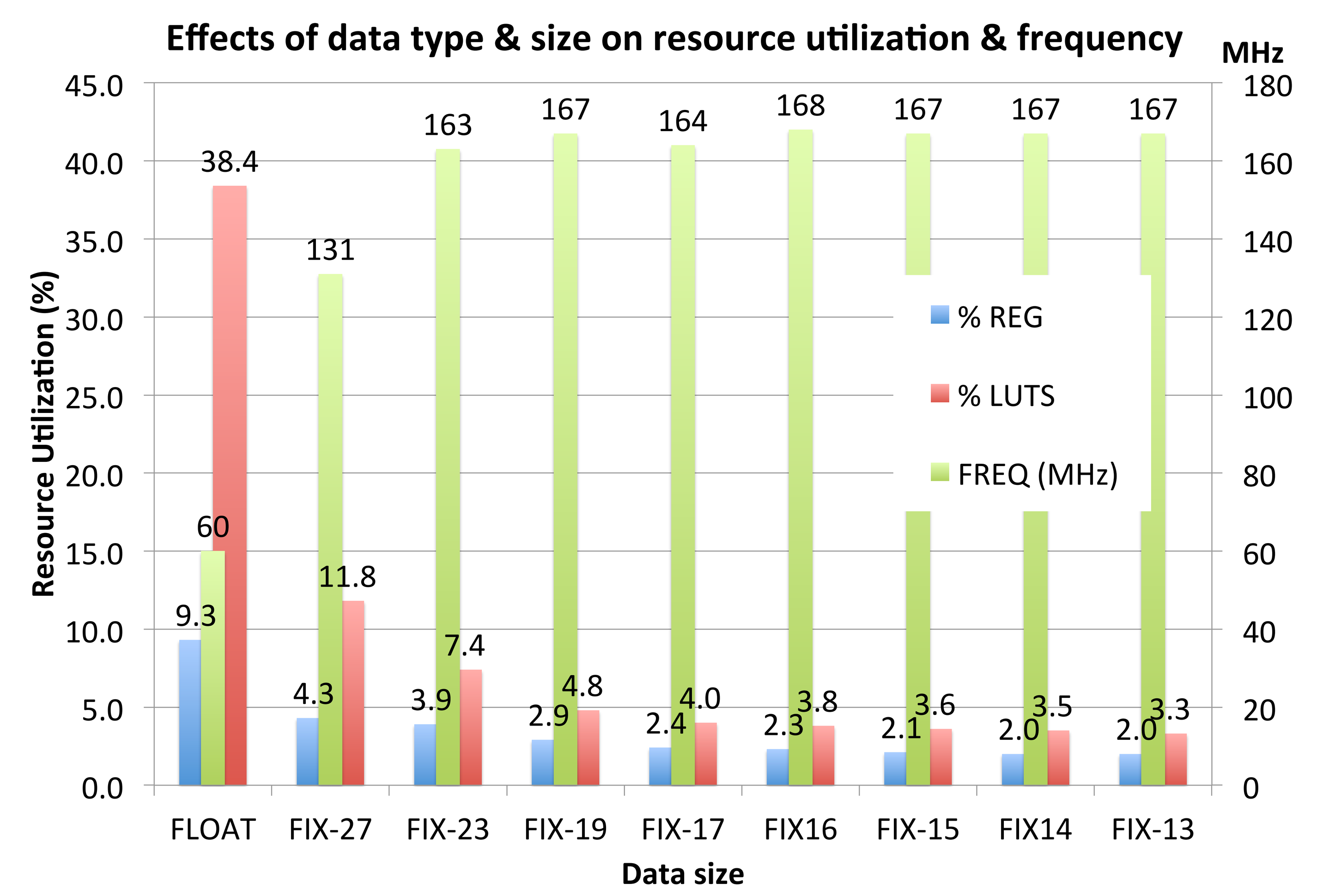



Fig. 3 Overall detection results.
Values averaged from all benchmarks




## EXPERIMENTAL EVALUATION

**FPGA implementation**:

- 13 bit fixed-point data, Convey HC-2ex, Xilinx Virtex-6 LX760 FPGAs, 16 memory channels of 64-bit per FPGA at 150 MHz.
- 15 HOG-Engines executing in parallel. One channel as shared output
- Single FPGA execution speed: **27.4 ms/frame**

**CPU:** Intel Xeon 5220 **GPU:** Nvidia K20

| | | CPU | GPU | FPGA fp | FPGA fix-13 |
|---|---|---|---|---|---|
| **Throughput** | frames/sec | 0.993 | 13.40 | 2.31 | 36.50 |
| | speed-up | 1.00 | 13.50 | 2.33 | 36.77 |
| **Energy** | Joules/frame | 80.60 | 16.80 | 4.02 | 0.69 |
| | ratio (%) | 100 | 20.84 | 4.99 | 0.855 |

## REFERENCES


[1] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in CVPR, 2005, pp. 886-893.
[2] M. Enzweiler and D. Gavrila, “Monocular pedestrian detection: Survey and experiments,” in PAMI, vol. 31, no. 12, pp. 2179-2195, 2009.
[3] “Opencv,” http://opencv.willowgarage.com.
[4] C. Wojek, S. Walk, and B. Schiele, “Multi-cue onboard pedestrian detection,” in CVPR, 2009, pp. 794–801.
[5] P. Dollar, C. Wojek, B. Schiele, and P. Perona, “Pedestrian detection: A benchmark,” in CVPR, 2009, pp. 304-311.
[6] A. Ess, B. Leibe, K. Schindler, and L. Van Gool, “A mobile vision system for robust multi-person tracking,” in CVPR, 2008, pp. 1–8.
[7] M. Everingham, A. Zisserman, C. K. Williams, L. Van Gool, et al., “The 2005 Pascal visual object classes challenge,” in Machine Learning Challenges. Springer, 2006, pp. 117–176.
[8] P. Dollar, C. Wojek, B. Schiele, and P. Perona, “Pedestrian detection: An evaluation of the state of the art,”, in PAMI, vol. 34, no. 4, pp. 743–761, 2012.