MINIMALISTIC IMAGE SIGNAL PROCESSING FOR DEEP LEARNING APPLICATIONS

Ekdeep Singh Lubana[⋆†], *Robert P. Dick*[⋆]

[⋆]University of Michigan
Ann Arbor, USA

Vinayak Aggarwal[†], *Pyari Mohan Pradhan*[†]

[†]Indian Institute of Technology
Roorkee, India

ABSTRACT

In-sensor energy-efficient deep learning accelerators have the potential to enable the use of deep neural networks in embedded vision applications. However, their negative impact on accuracy has been severely underestimated. The inference pipeline used in prior in-sensor deep learning accelerators bypasses the image signal processor (ISP), thereby disrupting the conventional vision pipeline and undermining accuracy of machine learning algorithms trained on conventional, post-ISP datasets. For example, the detection accuracy of an off-the-shelf Faster RCNN algorithm in a vehicle detection scenario reduces by 60%. To make in-sensor accelerators practical, we describe energy-efficient operations that yield most of the benefits of an ISP and reduce covariate shift between the training (ISP processed images) and target (RAW images) distributions. For the vehicle detection problem, our approach improves accuracy by 25–60%. Relative to the conventional ISP pipeline, energy consumption and response time improve by 30% and 34%, respectively.

Index Terms— Deep learning accelerators, Image signal processor, RAW images, Covariate shift

1. INTRODUCTION

Deep learning, the current paradigm in machine learning algorithms, has achieved state-of-the-art performance in several application domains. However, the high energy, computation, and memory demands of deep neural networks (DNNs) limits their deployment in embedded applications. Vision applications provide fertile ground for achieving energy efficiency by allowing parallel execution of primitive analysis operations. These abilities are exploited by several in-sensor and near-sensor accelerators that use image sensor parallel readout capabilities to enable efficient convolution operations [1, 2, 3, 4].

The conventional (energy-intensive but accurate) imaging pipeline uses an image signal processor (ISP) to perform several non-linear transformation operations on an image before further analysis. However, existing in-sensor accelerators bypass the ISP, thereby disrupting the imaging pipeline and undermining accuracy (see Figure 1). In this paper, we consider the impact of this dataflow design choice and describe an alternative.

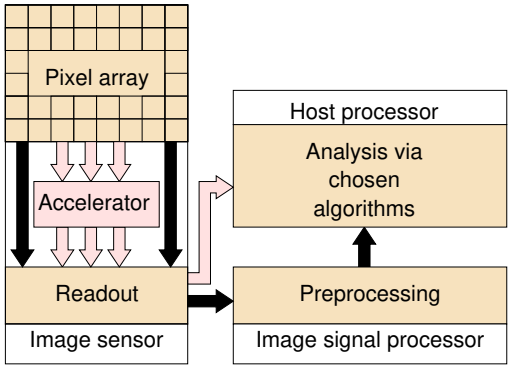
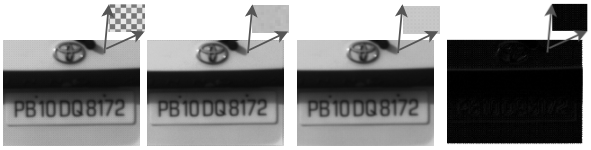


Fig. 1: A comparison of the conventional machine vision pipeline (black arrows) and in-sensor accelerators' pipeline (pink arrows). In-sensor accelerators bypass the ISP.

In particular, we observe that augmenting an image sensor with a pre-ISP deep learning accelerator introduces a deployment challenge. Since DNNs are typically trained on images already processed by the ISP, they learn distributions that differ from those of the RAW images produced by the image sensor. This issue of disparity in training (ISP processed images) and target (RAW images) data distributions[1] is typically known as covariate shift, and greatly reduces application accuracy for in-sensor accelerator pipelines; we observed 60% accuracy reduction for the cases we studied. A possible solution would be training networks on pre-ISP data (RAW images). However, that would either require developing new, large datasets of RAW images (an expensive task; see Section 4) or making do with limited training data, thus reducing accuracy. In contrast, our proposed approach allows in-sensor accelerators to directly use off-the-shelf DNNs trained on ISP-processed images or train with large existing data sets, without degrading accuracy.

This paper makes the following contributions. It describes a minimalistic image signal processing pipeline that improves the accuracy of in-sensor accelerators using DNNs trained on

[1]The probability distribution of the intensity of a pixel is defined as its data distribution.



(a) RAW. (b) Demosaic. (c) Bin. (d) Diff.

Fig. 2: (a) The RAW image has a tiled pattern. (b) Demosaicing gives a smooth image. (c) Binning renders a smooth image too. (d) The difference in demosaiced and binned image is minimal (for the given 400×400 image, MSE = 4.68).

conventional, ISP-processed datasets by performing gamma compression and pixel binning on the acquired image. These operations can be easily integrated within an image sensor, eliminating the need for an ISP, and making in-sensor accelerators feasible by performing the necessary local transformations. Using our approach, it is possible to maintain accuracy while eliminating the ISP, thereby improving detection accuracy by 25–60% and reducing system response time by 34% and system energy 30% in ISP retaining near-sensor accelerators [4].

Please note that our findings do not diminish the contributions of in-sensor accelerators: this paper finds flaws in the pipeline used in prior research and describes methods allowing in-sensor accelerators to reach their potential.

2. THE MACHINE VISION PIPELINE

The conventional machine vision pipeline involves the following three stages.

Sensing: The image sensor is responsible for sampling and uses photodiodes to convert incoming photons to proportional voltage levels. Images outputted by the image sensor are known as RAW images and are discontinuous as a result of using color filter arrays to capture the red, green, and blue intensities separately (see Figure 2(a)). Please note that our proposed methods can be extended to other color filter array patterns with minimal changes at the readout stage.

Preprocessing: Preprocessing takes place at the ISP. Conventionally, these operations aim at improving image aesthetics and are proprietary to manufacturers. However, the pipelines are generally similar and include several operations such as demosaicing, the conversion of the discontinuous, single-channel RAW image to a continuous, multi-channel (e.g., RGB) image. Finally, the RAW images are converted into a standard format such as JPEG or PNG.

Inference: A host processor executes application-specific algorithms for detection, classification, or other tasks.

In-sensor accelerators modify this pipeline by executing convolutional operations within the image sensor itself. The resulting features are then processed by the host processor (see Figure 1). This pipeline caries out feature extraction before ISP operations. As a result, the features are quite different from those of conventional, post-ISP training datasets. This results in covariate shift between the training (ISP processed images) and target (RAW images) data, significantly reducing application accuracy (see Section 5).

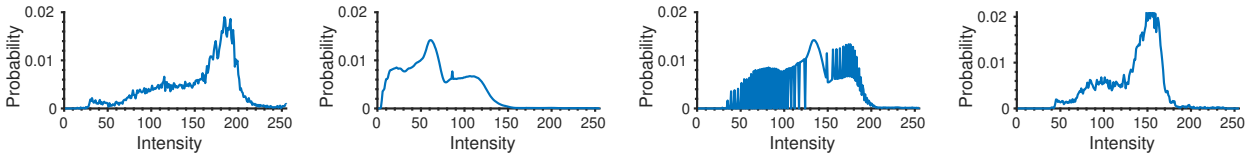
3. PRIOR WORK

Conventional machine vision systems are based on the pipeline described in Section 2, in which the ISP influences both accuracy and energy efficiency. Buckler et al. studied the impact of individual ISP operations on task efficiency [5]. They used a software tool to convert ISP processed images to RAW images, and found that demosaicing, denoising, and gamma compression are crucial for maintaining high application accuracy. However, since several of the operations involved in the ISP pipeline are non-linear and non-invertible, the RAW images can only be approximately reconstructed– for ex., noise needs to be explicitly added to the image signal. It is possible that their approximate reconstruction of RAW images might have led Buckler et al. to overestimate the impact of demosaicing and denoising operations on task efficiency. In actuality, we find that any smoothing algorithm that removes the tiling effects of the color filter array and has partial denoising effects is sufficient to produce high application accuracy. Buckler found this plausible in an email correspondence. Pixel binning has denoising effects and smooths the discontinuous RAW images to produce continuous images (see Figure 2), achieving continuity similar to that of demosaicing. Further, pixel binning can be performed within image sensors by using their parallel readout architecture, making it suitable for in-sensor deep learning accelerators. Therefore, we propose a two-step preprocessing pipeline that involves gamma compression and pixel binning only.

4. FINDING THE NECESSARY PIPELINE OPERATIONS

Since a RAW and ISP-processed image correspond to the same scene, they both capture the scene structure and geometry. Their major differences are due to local transformation operations such as gamma compression, which alter the data distributions of pixels. Since machine vision algorithms are trained on ISP-processed images, they learn representations based on the transformed data distributions. While an argument can be made for training networks on RAW images, there are several hindrances: (1) RAW image datasets are not publicly available (which this paper partially addresses through publication of such a dataset); (2) the RAW image format is not standardized, and conversion to and from other formats such as JPEG and PNG is proprietary; and (3) RAW images have large sizes, impeding ease of transfer.

In this section, we propose a minimalistic image signal processing pipeline that uses gamma compression and



(a) ISP-processed. (b) RAW. (c) Gamma comp. (d) Gamma comp. & binning.

Fig. 3: The intensity distributions for ISP-processed and RAW images are dramatically different. Use of gamma compression produces a non-linear transformation of (b), which, upon denoising using binning, gives an approximation of (a).

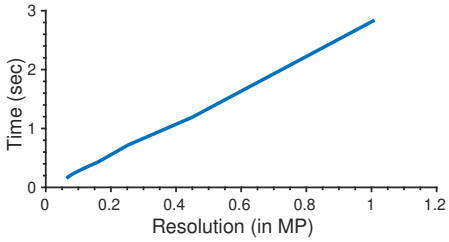


Fig. 4: Faster RCNN processing time increases linearly with resolution.

pixel binning to approximate the data distribution of ISP processed images. This reduces the amount of covariate shift between the ISP processed (training) and RAW images (target), thereby enabling the use of off-the-shelf DNNs in in-sensor and near-sensor accelerators. We analyze the effect of our proposed pipeline on pixel intensity distribution as an analog for the data distribution.

4.1. Gamma Compression

Human vision has a logarithmic response to incident illumination energy, in contrast to the linear response of image sensors. ISPs focus on rendering aesthetically appealing images. As a result, RAW image intensity distributions are starkly different from those of the post-ISP images (see Figure 3).

Gamma compression is a local, non-linear transformation that exponentiates individual pixels with an exponent less than 1. We observe that using gamma compression, a noisy approximation of the ISP-processed intensity distribution can be achieved (see Figure 3(c)). We'll be using the Adobe 1998 standard for gamma compression in our work [6].

$$P_{gamma} = (P_{norm})^{\gamma} \tag{1}$$

4.2. Pixel Binning

Pixel binning is a well known subsampling scheme that involves averaging followed by decimation. Since averaging removes gaussian noise, binning also reduces noise [7]. As shown in Figure 3(d), binning the gamma compressed image results in denoising and produces a smooth intensity distribution. Pixel binning also has another benefit: it reduces the image size by subsampling, thus enabling more efficient analysis. Since DNN-based object detection algorithms use small convolutional kernels, high-resolution images proportionately increase processing time and energy (see Figure 4). Even on a 6 GiB NVIDIA GTX 1060 GPU, images larger than 1,000×1,000 exceed available memory. Prior work on in-sensor accelerators has neglected the impact of image size because evaluations were on images too small for realistic object detection.

The binning operation for subsampling an $s \times s$ image using a $w \times w$ binning window can be formulated as follows:

$$P_{bin}(i,j) = \frac{1}{2w+1} \sum_{k=-w}^{+w} P(s \times i + k, s \times j + k). \tag{2}$$

4.3. Hardware Implementation

The operations described above are sufficient to approximate the data distributions of ISP-processed images (see Figure 3). Since our goal is to achieve high accuracy and enable use of off-the-shelf DNNs for in-sensor deep learning accelerators, we must incorporate gamma compression and pixel binning before image sensor readout. To this end, we propose the following modifications to image sensors.

Logarithmic pixels: Image sensors use linear photodiodes to proportionately convert illumination energy into voltage. In logarithmic pixels, the source follower transistor in the active pixel sensor functions in the sub-threshold region, resulting in logarithmic output voltage response with respect to incident illumination energy [8]. It is known that the logarithmic function can well approximate gamma compression [9]. Therefore, using logarithmic pixels, gamma compression can be integrated within the sensor itself.

Pixel binning: Binning can be performed via an analog averaging circuit at the readout stage. Specifically, the parallel readout pipeline ends in a chain of accumulating capacitors, which can be shorted for charge sharing and averaging. Pixel binning is supported by almost all modern image sensors. While it requires the use of additional circuitry for averaging, increasing sensor power consumption, the increase is negligible [10]. Further, since readout time decreases quadratically with the binning window length, net sensing energy is decreased.

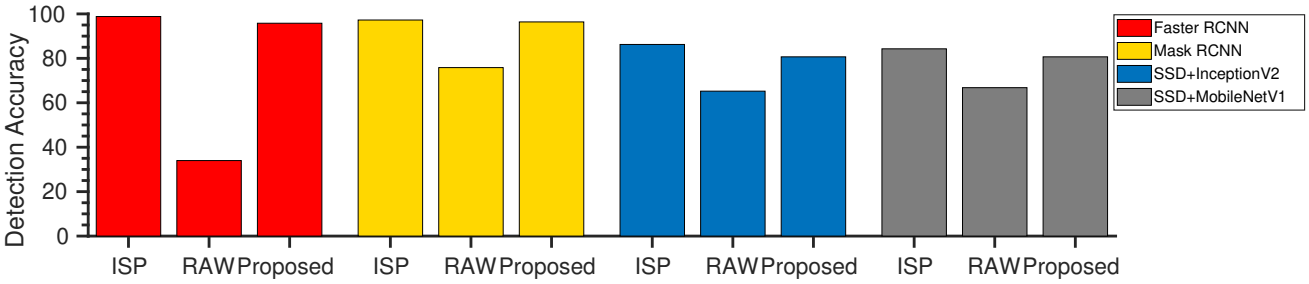


Fig. 5: Detection accuracy for conventional post-ISP images, RAW images, and proposed pipeline images.

(a) Proposed (b) Post-ISP (c) Ground truth

Fig. 6: Using the proposed pipeline, off-the-shelf DNNs can be used on RAW images.

Table 1: Mean average precision for different networks.

Pipeline	RAW	Proposed	ISP
Faster RCNN	0.07	1	0.87
Mask RCNN	0.66	0.85	0.84
SSD+InceptionV2	0.62	0.76	1
SSD+MobileNet	0.6	1	0.8

5. EVALUATION

We simulate our proposed sensing framework's effects by performing gamma compression and pixel binning on RAW images. As mentioned before, prior work on in-sensor accelerators shows evaluation results on ISP-processed images, thereby underestimating the impact of dataflow choice on accuracy. However, we use RAW images acquired from a commercial camera for our evaluation, as would be seen by an in-sensor accelerator. The dataset we gathered for testing consists of 225 RAW images, which have 1,215 vehicle instances in total. This dataset has been made publicly available[2]. These images have not been processed by the ISP and are essentially just the digital versions of the raw analog image.

Impact on accuracy: We use TensorFlow's model zoo [11], which includes several state-of-the-art object detection networks, as our benchmark for evaluating object detection accuracy under various considerations. Figure 5 shows the detection accuracy of the RAW images, images processed via the proposed pipeline, and corresponding ISP processed images in a vehicle-detection scenario. We define an object to have been detected when the determined bounding box has greater than 40% overlap with the ground truth bounding box (see Figure 6). The detection accuracy improves by 25–60% over RAW images, which in-sensor accelerators would otherwise encounter. Table 1 enumerates the mean average precision (mAP) values for the different networks and processing pipelines. Note that use of strong feature extractors in the Faster RCNN and MobileNet implementations results in a higher amount of false positives in ISP processed images, reducing net mAP values.

Impact on energy and response time: In order to evaluate the impact of the ISP on an embedded vision system's response time and energy, we use a Raspberry Pi 3 microcontroller to calculate the net time consumed by a conventional ISP pipeline. The Raspberry Pi allows easy control of the ISP and supports TensorFlow-lite, a low-level API for TensorFlow. To fit within available memory, a quantized version of SSD MobileNet V2 is used. The energy evaluation model used is as described by Lubana and Dick [12]. Our results show 34% reduction in system latency and 30% reduction in system energy relative to the conventional ISP pipeline for the vehicle-detection problem. These results imply that using our proposed pipeline, near-sensor accelerators that retain the ISP [4] can further reduce analysis latency and system energy consumption by bypassing the ISP altogether.

6. CONCLUSION

This paper explains why in-/near-sensor deep learning accelerators cannot use off-the-shelf DNNs without running a minimal but carefully designed preprocessing pipeline. The necessary operations, gamma compression and pixel binning, approximate the data generating distribution of the ISP processed images and can be easily incorporated within an image sensor by using logarithmic pixels and binned readouts. The proposed method enables the use of off-the-shelf DNNs in in-sensor and near-sensor accelerators, reduces their error rate by 25–60%, and decreases system energy consumption by 30% and analysis latency by 34%.

[2]https://github.com/EkdeepSLubana/raw_dataset

7. REFERENCES

[1] R. LiKamWa, Y. Hou, Y. Gao, M. Polansky, and L. Zhong, "RedEye: Analog ConvNet Image Sensor Architecture for Continuous Mobile Vision," in *Proc. Int. Symp. on Computer Architecture*, June 2016, pp. 255–266.

[2] K. Lee, S. Park, S. Park, J. Cho, and E. Yoon, "A 272.49 pJ/pixel CMOS Image Sensor with Embedded Object Detection and Bio-inspired 2D Optic Flow Generation for Nano-air-vehicle Navigation," in *Symp. on VLSI Circuits*, June 2017, pp. C294–C295.

[3] M. F. Amir, J. H. Ko, T. Na, D. Kim, and S. Mukhopadhyay, "3-D Stacked Image Sensor With Deep Neural Network Computation," *IEEE Sensors Journal*, vol. 18, pp. 4187–4199, May 2018.

[4] Z. Du, R. Fasthuber, T. Chen, P. Ienne, L. Li, T. Luo, X. Feng, Y. Chen, and O. Temam, "ShiDianNao: Shifting Vision Processing Closer to the Sensor," in *Proc. Int. Symp. on Computer Architecture*, June 2015, pp. 92–104.

[5] M. Buckler, S. Jayasuriya, and A. Sampson, "Reconfiguring the Imaging Pipeline for Computer Vision," in *Proc. Int. Conf. on Computer Vision*, Oct 2017, pp. 975–984.

[6] Adobe Systems Incorporated, "Inverting the color component transfer function," Tech. Rep., Adobe RGB (1998) Color Image Encoding, 2005.

[7] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing (3rd Edition)*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2006.

[8] H. Cheng, B. Choubey, and S. Collins, "A Wide Dynamic Range CMOS Image Sensor with an Adjustable Logarithmic Response," in *Proc. SPIE Sensors, Cameras, and Systems for Industrial/Scientific Applications*, 2008, vol. 6816, pp. 1 – 8.

[9] A. Omid-Zohoor, C. Young, D. Ta, and B. Murmann, "Toward Always-On Mobile Object Detection: Energy Versus Performance Tradeoffs for Embedded HOG Feature Extraction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 5, pp. 1102–1115, May 2018.

[10] D. Kim, M. Song, B. Choe, and S. Y. Kim, "A Multi-Resolution Mode CMOS Image Sensor with a Novel Two-Step Single-Slope ADC for Intelligent Surveillance Systems," *Sensors*, vol. 17, no. 7, 2017.

[11] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy, "Speed/Accuracy Trade-Offs for Modern Convolutional Object Detectors," in *Proc. Conf. on Computer Vision and Pattern Recognition*, 2017, pp. 3296–3297.

[12] E. S. Lubana and R. P. Dick, "Digital foveation: An energy-aware machine vision framework," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 37, no. 11, pp. 2371–2380, Nov 2018.