Running 2-bit quantized CNN models on ARM CPUs

Davis Sawyer
Co-founder & CPO

Deeplite

May 4, 2023

Optimize AI for the Edge

Founded 2019

Edge Computing Challenges

High Computational Complexity

Millions of expensive floating-point operations for each input classification are needed.

Memory Footprint

Huge amounts of weights and activations with limited on-chip memory and bandwidth.

Power consumption

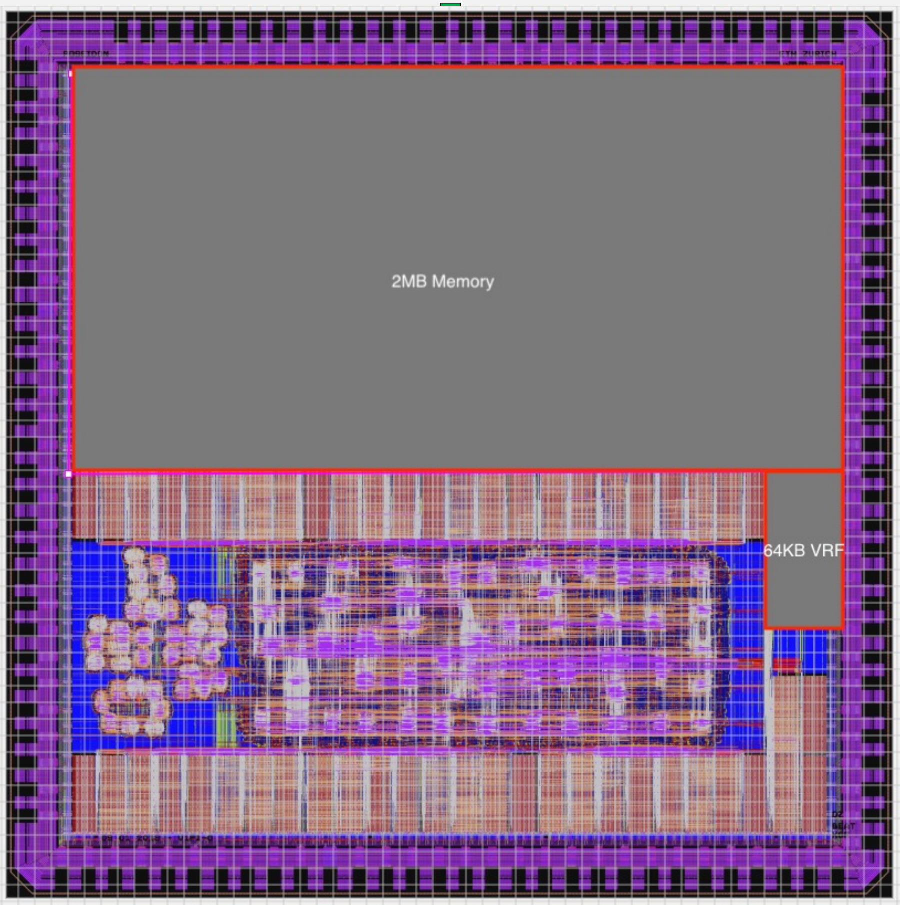
Deep learning requires significant power and can easily consume battery life

In-memory / near memory Computation

- **We must bring computation at the very edge (on-chip, in-memory)** Otherwise we lose a lot only on memory bottleneck

- **Size is the biggest challenge in this environment**

Edge of Computing



Industrial EdgeAI Challenges

- Big models/datasets
  - Expensive Training Step
- Very sensitive on accuracy drop depends on the use case
- Expecting real time execution on their edge HW
- Generic and automated approaches which work on all use cases

Deeplite Solution

5 Neutrino Annotator & Fine Tuner
- Create dataset from raw data
- Update model accuracy
- Data localization and drift

Fine-tune model to boost accuracy up to 20%

Continuously learn from Edge Device Data

1 Model Development

Deeplite Model Zoo
- Purpose-built Architectures
- Piccolo Detectors
- Hardware-optimized Models (GPU, NPU, CPU)
- Standard Model Zoo (50+ open-source CV models)
- Bring your own data (BYOD)

2 Training

Deeplite Neutrino Training
- Model Convertor
- Training Loops
- Model Profiler

Neutrino Annotator Training Data

Neutrino+ PyTorch

3 Model Optimization

Deeplite Neutrino Optimization
- Preserve Accuracy
- Compression
- Quantization
- Hardware-Aware
- Multiplier-Free Deep Neural Networks (for specialized AI Hardware)

4 On-device Inference

DeepliteRT
- Low-bit mixed-precision Runtime (Arm Cortex-A series)
- Full precision runtime (x86, Arm, GPU)

End-to-end Computer Vision Pipeline

Flexible platform with custom and open-source options

Sophisticated and production-tested model optimization

Neutrino Model Optimization

Pre-trained Model

Compression	Dramatically reduce model size for inference deployment on memory and resource-constrained devices. Neutrino's SOTA compression suite provides 2-30x compression without sacrificing accuracy on vision tasks.	**2-30x compression** — **User controls accuracy**
Hardware-Aware	Accelerate model inference and increase throughput on a range of target hardware. Automatically build models for the target hardware using model design space exploration, even **replacing unsupported operations** with supported ops. Speed up inference 2-5x on target Hardware.	**2-5x speed-up** — **User-defined Hardware target**
Quantization	The world's first ultra compact quantization for PyTorch Computer Vision models. Leverage Neutrino's low-bit mixed precision abilities (1,2,4,8 and 32-bit) for faster inference and optimized memory footprint on CPUs and MCUs.	**2-15x compression** — **User controls accuracy**
Multiplier-Free DNNs	Accelerate model inference by eliminating expensive multiplication operations using 1: Power-of-2 quantization or 2: Use Look-up tables to store low-bit weights/activations	**IP to deliver several order of magnitudes performance improvements on specialized hardware**

10x improvement
(more FPS, inferences/sec)

at lower cost
(less W, less MB, less $)

Our Low-bit Quantization Contribution

DeepliteNeutrino

Model Optimization

- QAT with 1 and 2-bit precision (weights and activations) for both detection and classification models.
- Supports any precisions (1,2,3, etc.)
- Mixed precision approach to minimize the accuracy drop of quantized models.

DeepliteRT

On-device Inference

- Custom ultra-low precision convolution operators to accelerate speed and memory throughput of quantized layers
- End-to-end framework to deploy and execute mixed precision ultra-low bit quantized models on Armv7 and Armv8 Cortex-A processors.

Issues with low bit quantization

- High accuracy loss for 2-bit quantization
- Mixed precision quantization
  - Different layers to have different bit precision to maximize accuracy
    - Sensitivity analysis
    - Minimize switching between different precision
    - trade-off between accuracy and speed
  - Automatic way with 2 bit, 8-bit and FP32 layers

Deeplite Neutrino Quantizer

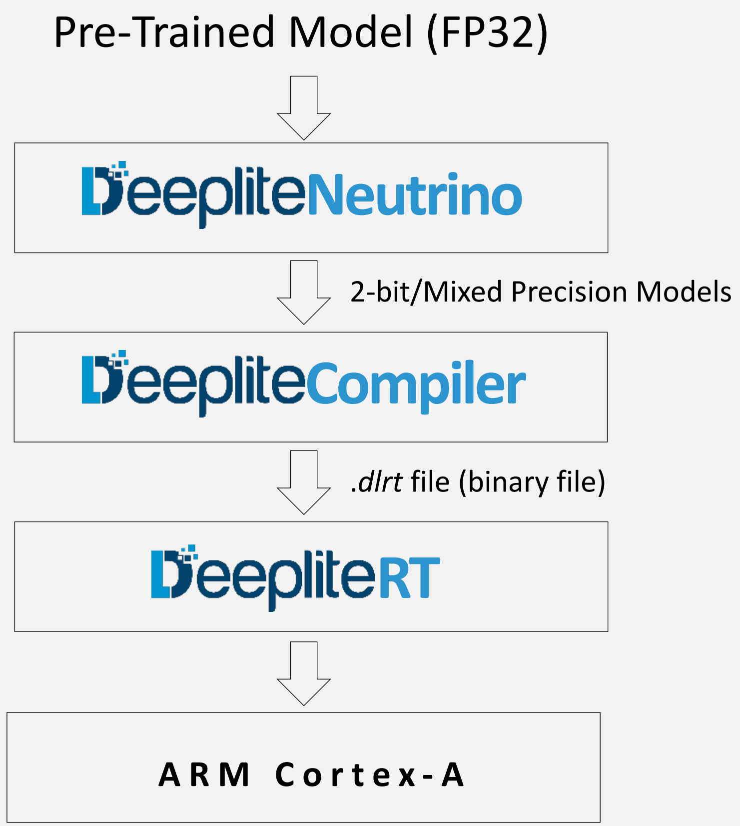
Sample Applications	Architecture	Deeplite Neutrino Quantization			precision	Accuracy Change	Dataset
		Original Size	Optimized Size	Improvement			
Image classification	ResNet18	42MB	2.9MB	**x15**	2w/2a	**0.00%** (Top1)	CIFAR100
	VGG19	76MB	5MB	**x15.3**	2w/2a	**<1.50%** (Top1)	CIFAR100
	ResNet18	42MB	2.9MB	**x15**	2w/1a	**<1.00%** (Top1)	VWW
	ResNet50	97MB	17MB	**x5.6**	2w/2a, 8w/8a[1]	**~1.50%** (Top1)	ImageNet
Object Detection	VGG16_SSD	90MB	5.6MB	**x16**	2w/2a	**<0.01** (mAP)	widerface
	VGG16_SSD	100MB	6.2MB	**x16**	2w/2a	**<0.02** (mAP)	VOC 2012
	Yolo5_6n	7MB	0.95MB	**x7**	2w/2a, FP32[1]	**<0.02** (mAP)	Custom (Person Detection)

[1]Mixed-Precision

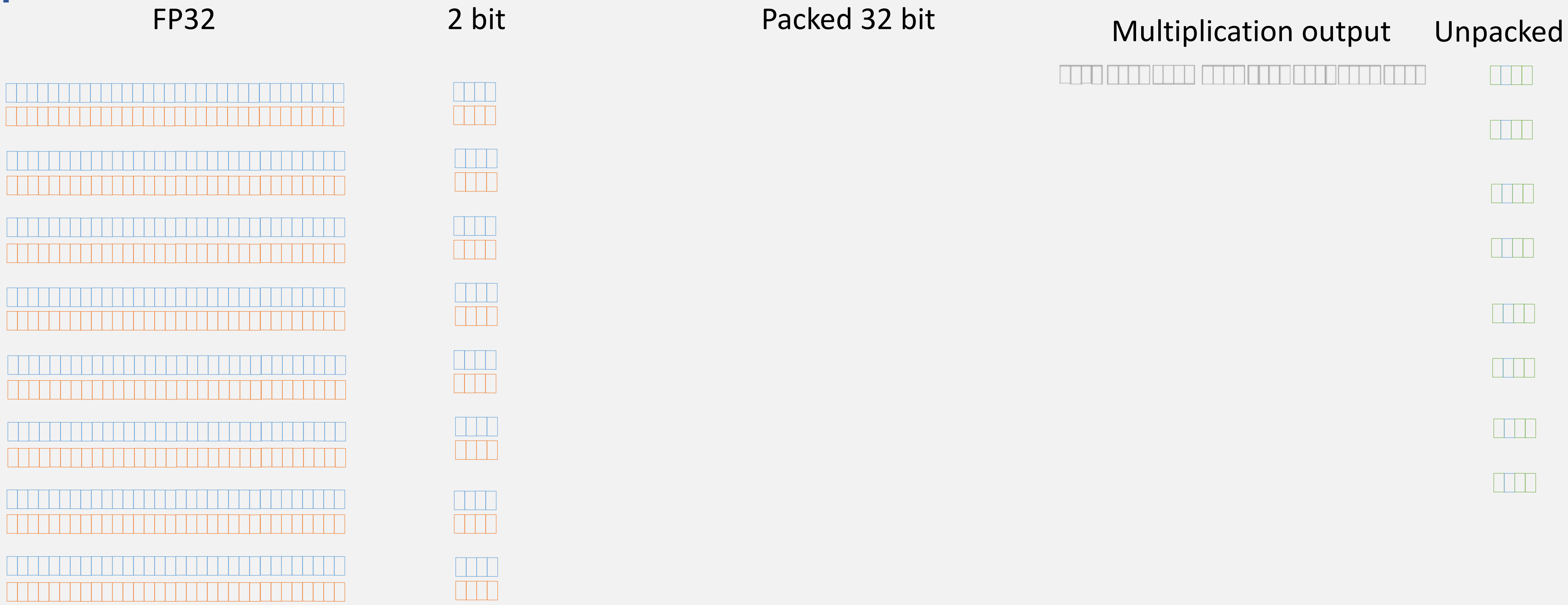
DeepliteRT

- Custom ultra-low precision convolution operators to accelerate speed and memory throughput of quantized layers

- End-to-end framework to deploy and execute mixed precision ultra-low bit quantized models on Armv7 and Armv8 Cortex-A processors.



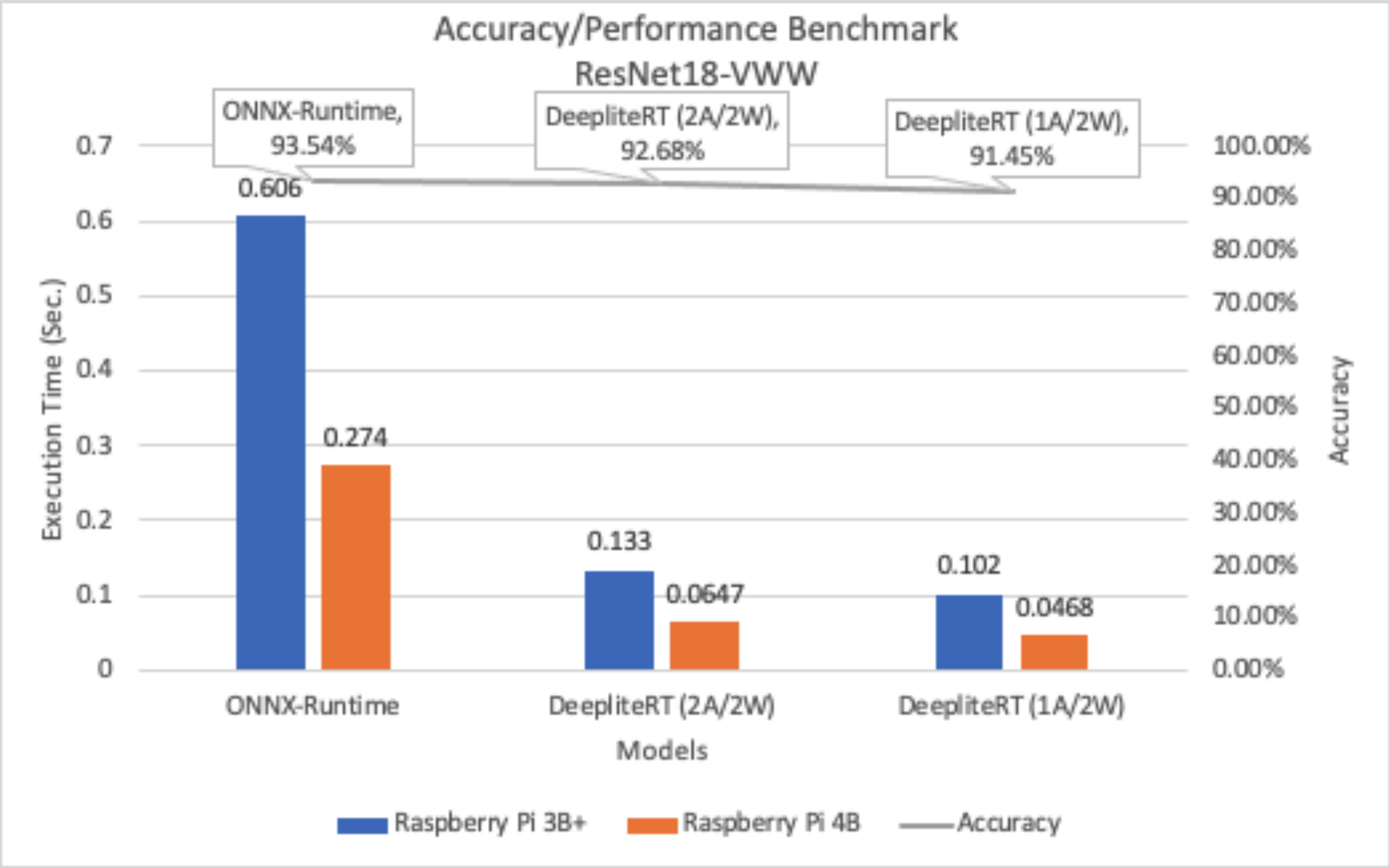
Low Bit Conv2D Implementation



DeepliteRT

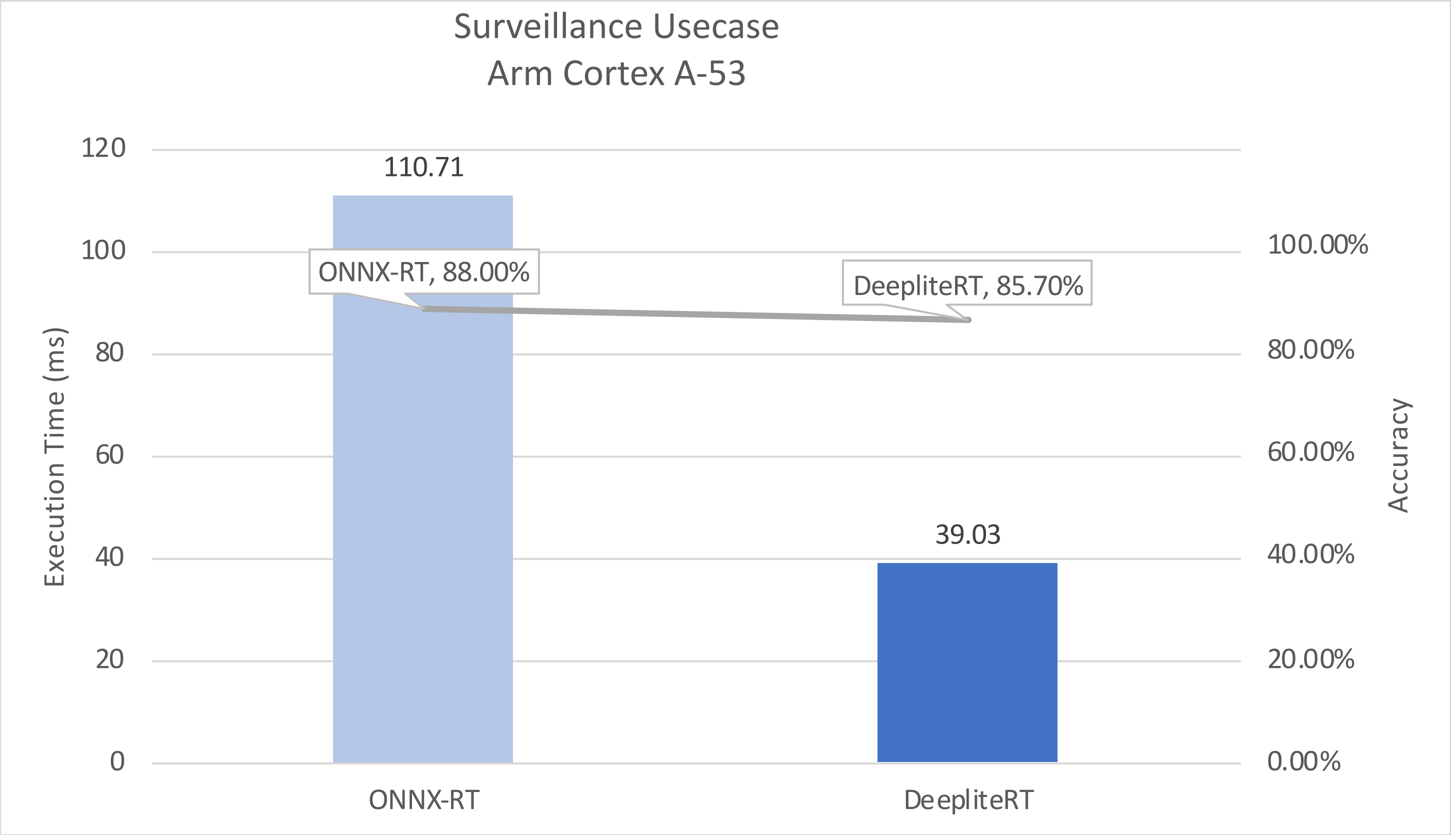
- Uses intrinsic from the Neon vectorized instruction set for both Armv7 and Armv8 architectures to target 32-bit and 64-bit Arm CPU devices.
- Efficient tiling and parallelization schemes used to improve upon the performance of the vectorized kernels.
- On the ResNet18 model running on the low-power Arm Cortex-A53 CPU in the Raspberry Pi 3B+, our overall implementation realizes speedups of up to 2.9x on 2-bit and 4.4x on 1-bit over an optimized floating-point baseline

ResNet18 on VWW



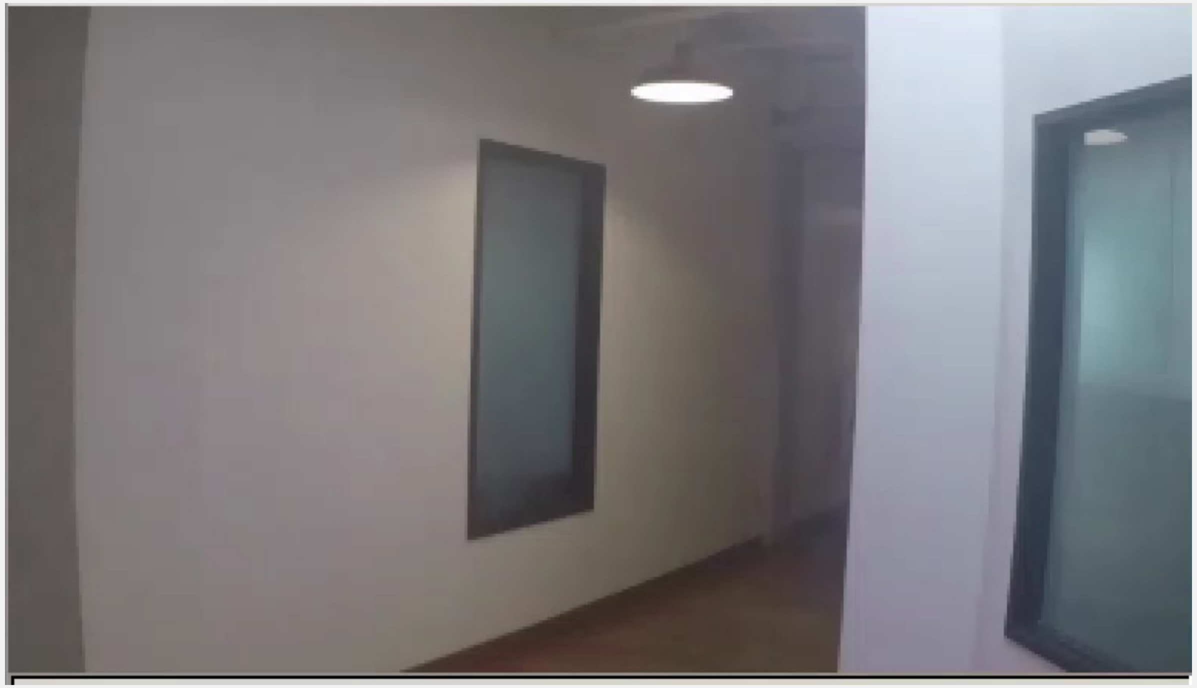
Accuracy/performance benchmark of DeepliteRT on ResNet18 model on VWW dataset (2A/2W- weights and activations quantized to 2 bits, 1A/2W- activations quantized to 1 bit and weights quantized to 2 bits)

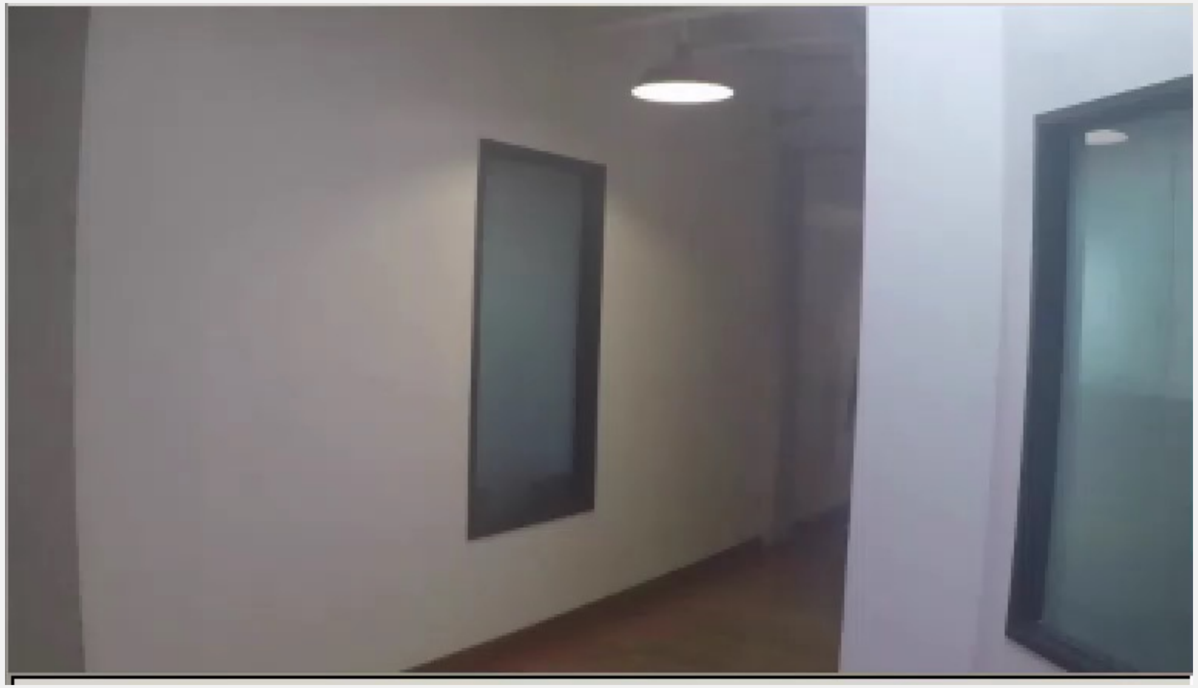
Industrial Use case #1



Object detection – Yolo5 based model

DeepliteRT - Detection Performance





ONNX RUNTIME

DeepliteRT

YOLOv5s COCO – Raspberry Pi 4B (4x Cortex-A72)

Combining All Techniques

MCU with 256KB on chip memory

x600

size reduction

12MB to 24KB
- KD
- Neutrino Compression Quantization

Use Case 1: Person Presence Detection

Baseline MobileNetv1 80% Top1 Accuracy

MLPerf submission is in progress…

Cortex-A53

100

FPS

- Neutrino Optimization & Low-bit Mixed-Precision Quantization

Use Case 2: Person Detection

Baseline Yolo5n Model

Conclusion

- 2-bit model running on commodity hardware is presented
- Mixed precision approach is used to minimize the accuracy loss
- Novel method is proposed to run the 2-bit model on Arm Cortex A devices
- Benchmark on classification and Object detection models are presented
- Future work
  - Extend to other hardware
  - Performance Improvement

Deeplite

Thank you

davis@deeplite.ai