LOW-PRECISION FLOATING-POINT FOR EFFICIENT ON-BOARD DEEP NEURAL NETWORK PROCESSING

Cédric GERNIGON

Univ Rennes, INRIA

CO-AUTHORS

Prof. Olivier SENTIEYS (Univ Rennes, INRIA)

Dr. Silviu-Ioan FILIP (Univ Rennes, INRIA)

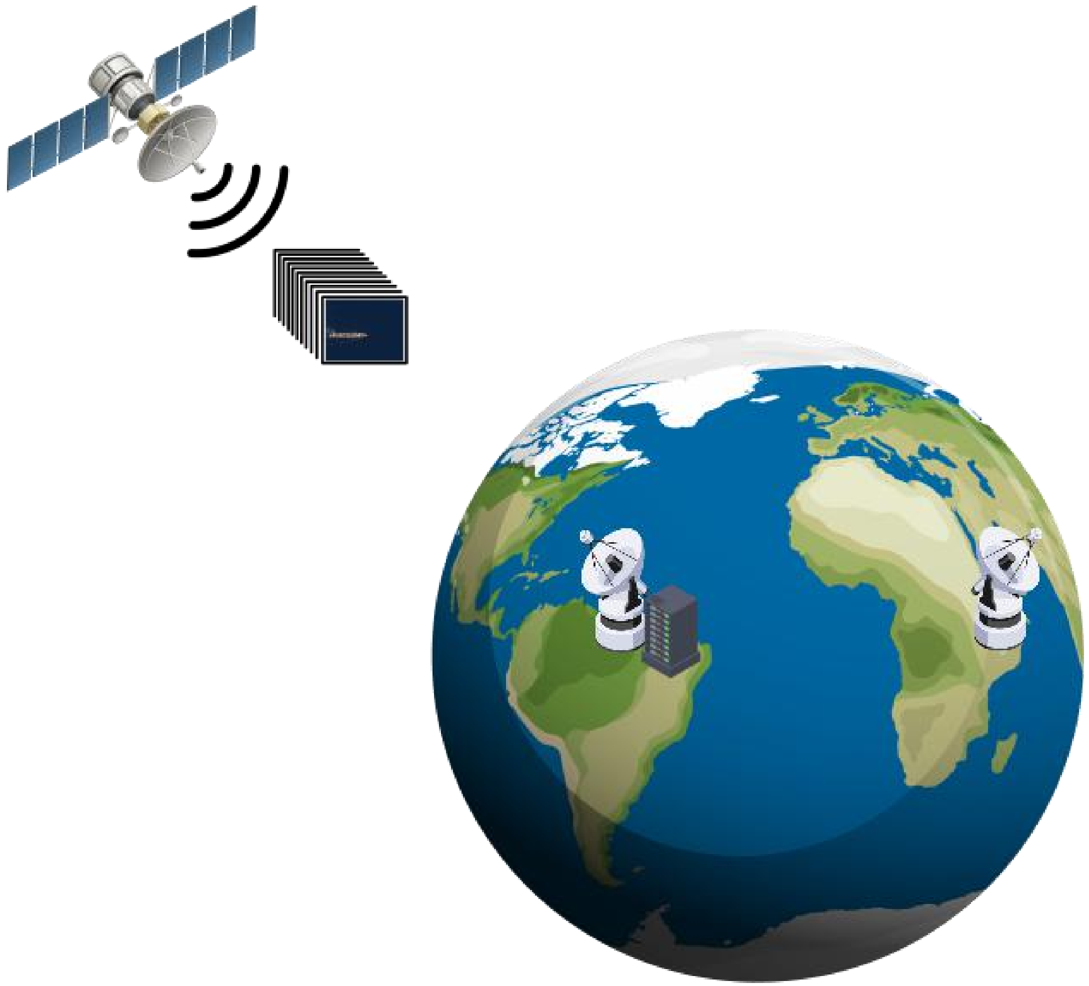
Clément COGGIOLA (CNES)

Mickaël BRUNO (CNES)

EDHPC 2023 2 - 6 October, 2023 Juan-les-Pins, France

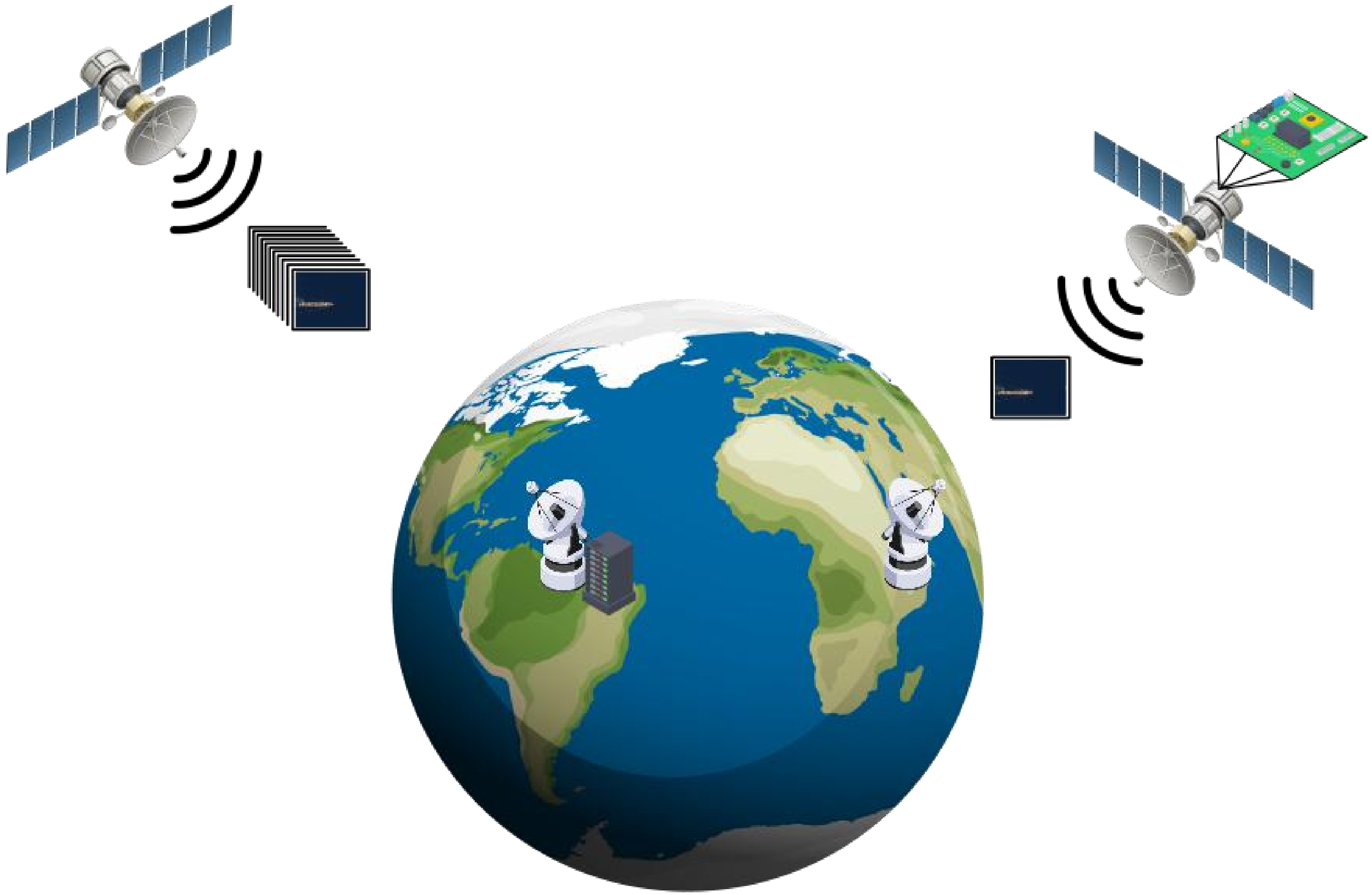
Context

- Earth Observation (EO) systems are limited by downlink communications



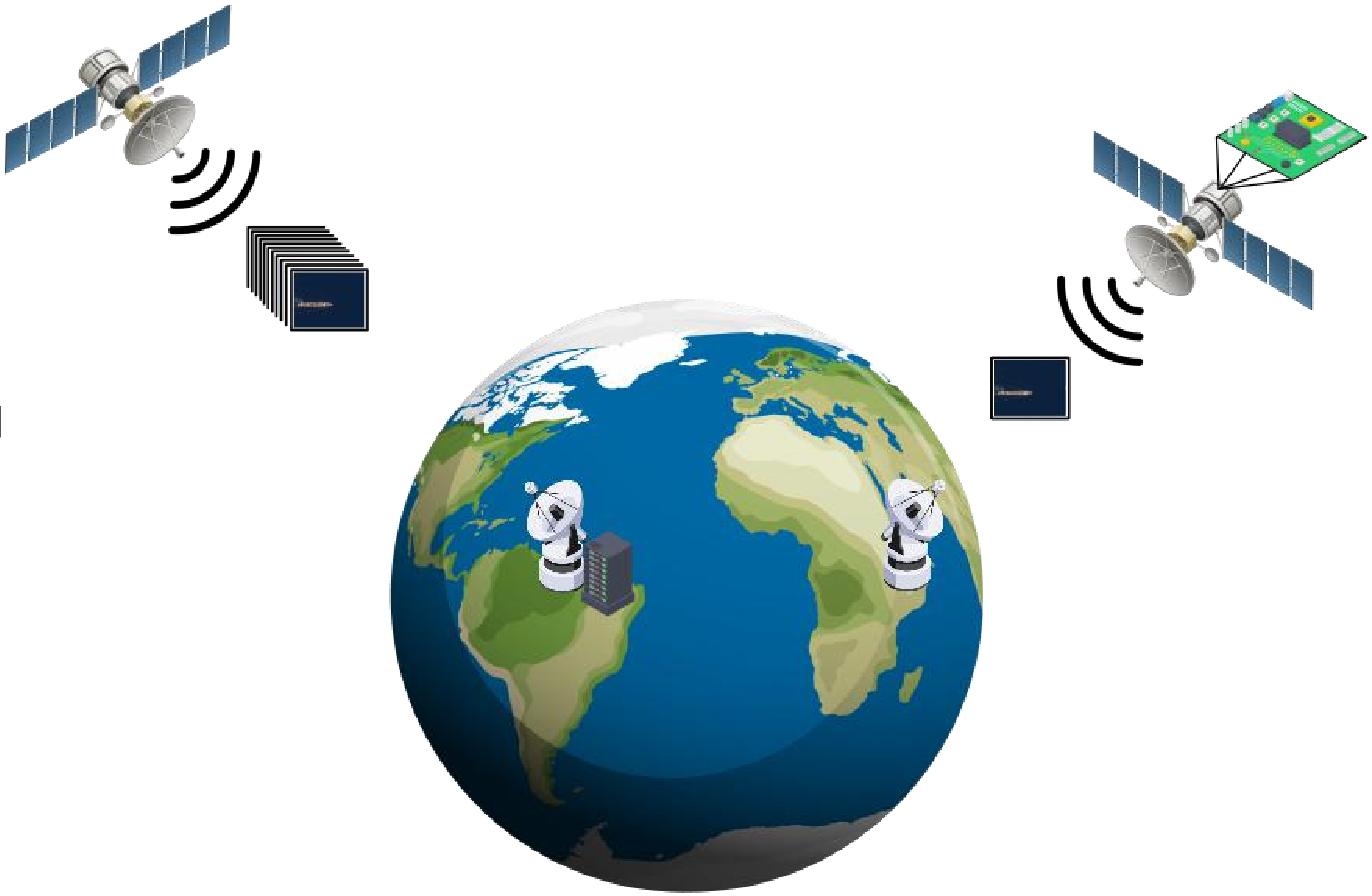
Context

- Earth Observation (EO) systems are limited by downlink communications
- An emerging solution is to transmit only relevant data through on-board processing



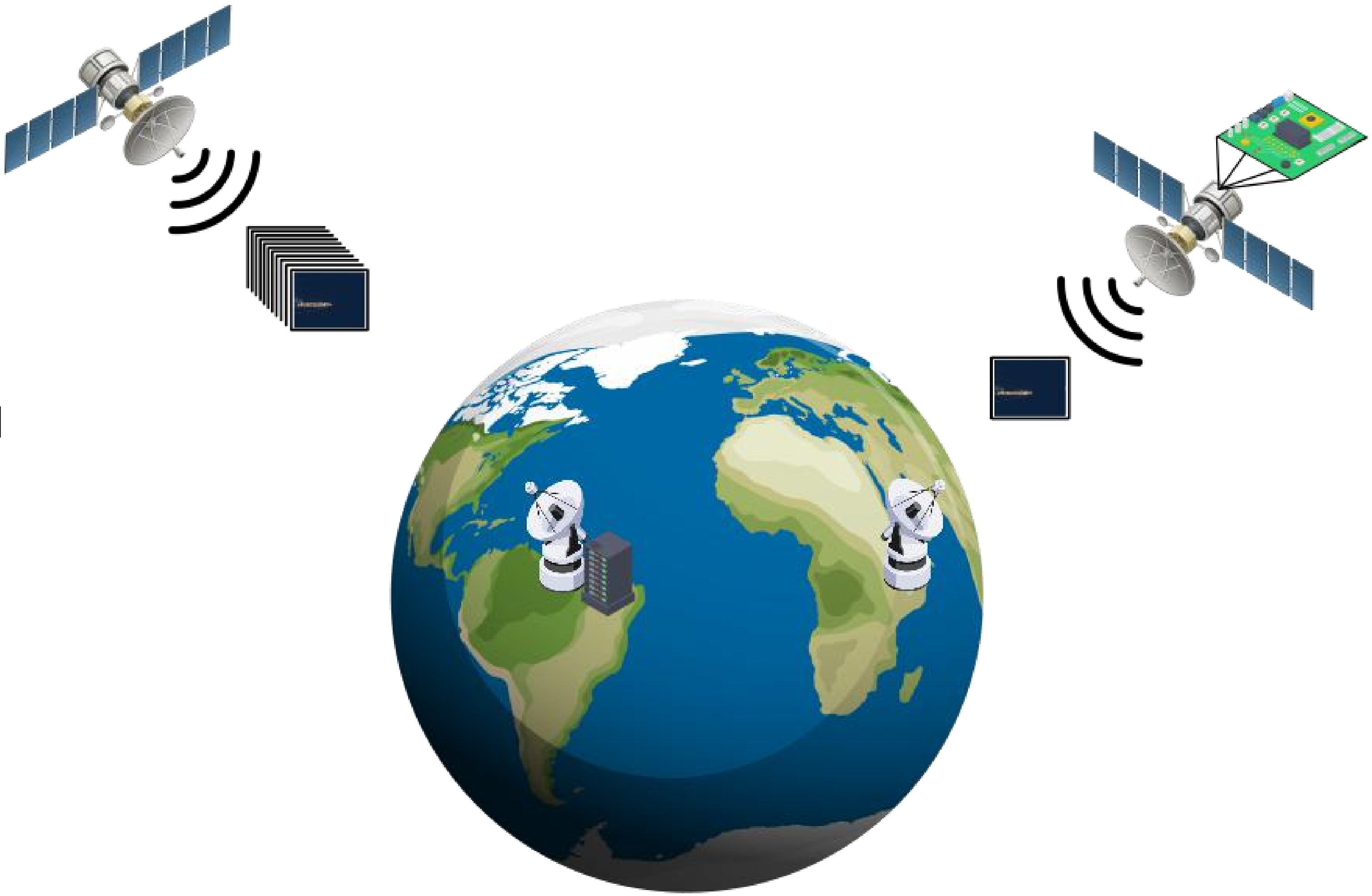
Context

- Earth Observation (EO) systems are limited by downlink communications
- An emerging solution is to transmit only relevant data through on-board processing
- The success of Deep Learning (DL) in space applications makes it a good candidate for on-board processing



Context

- Earth Observation (EO) systems are limited by downlink communications
- An emerging solution is to transmit only relevant data through on-board processing
- The success of Deep Learning (DL) in space applications makes it a good candidate for on-board processing
- Embedded DL is constrained by:
  - Hardware limitations
  - Power supply
  - Computing capacity



DNN compression

DNN compression methods:

- Pruning
- Weight sharing
- Efficient model architecture
- **Quantization**

DNN compression

DNN compression methods:

- Pruning
- Weight sharing
- Efficient model architecture
- **Quantization**

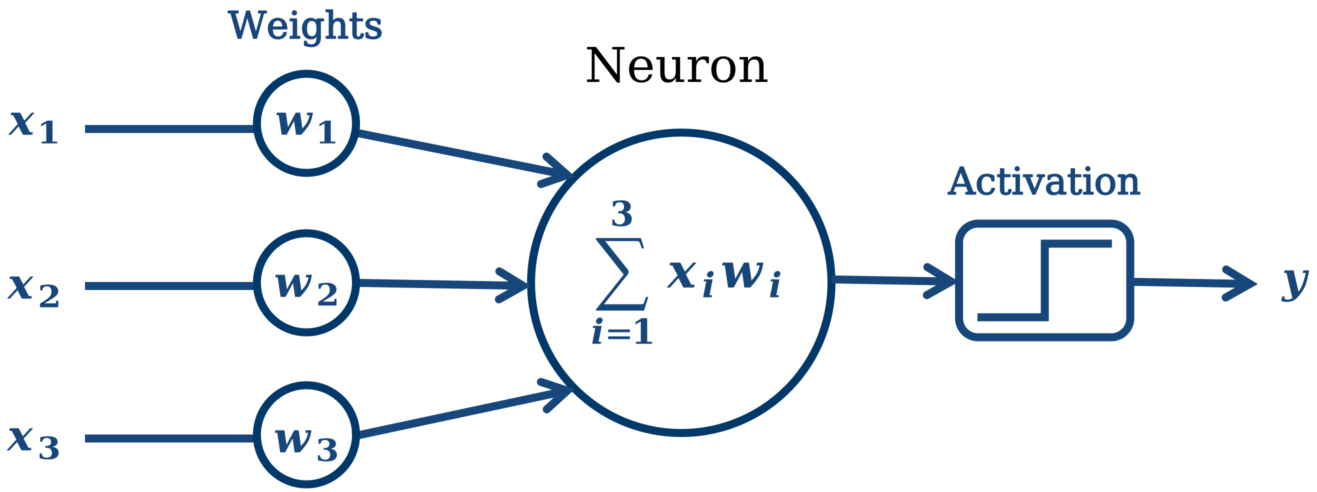
Pros and Cons

Pros	Cons
Memory usage	Less accurate
Power consumption	
Latency	

DNN compression

DNN compression methods:

- Pruning
- Weight sharing
- Efficient model architecture
- **Quantization**



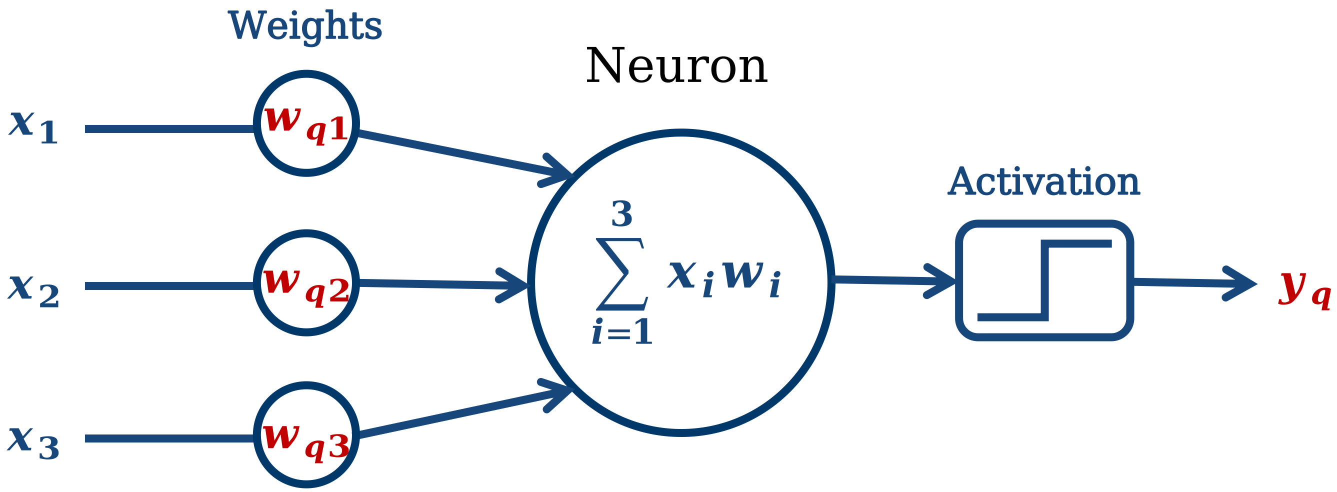
Pros and Cons

Pros	Cons
Memory usage	Less accurate
Power consumption	
Latency	

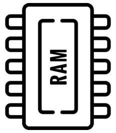
DNN compression

DNN compression methods:

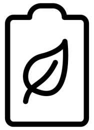
- Pruning
- Weight sharing
- Efficient model architecture
- **Quantization**



Pros and Cons



Memory usage

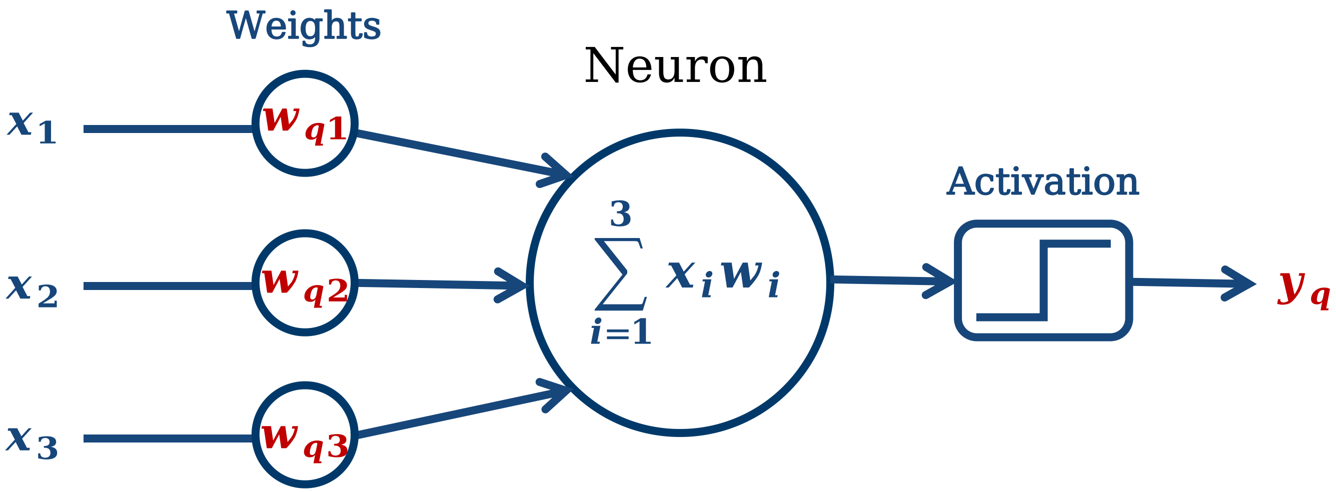

Power consumption

Latency

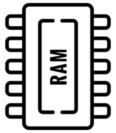

Less accurate

DNN compression

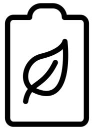
DNN compression methods:
- Pruning
- Weight sharing
- Efficient model architecture
- **Quantization**



Pros and Cons



Memory usage


Power consumption

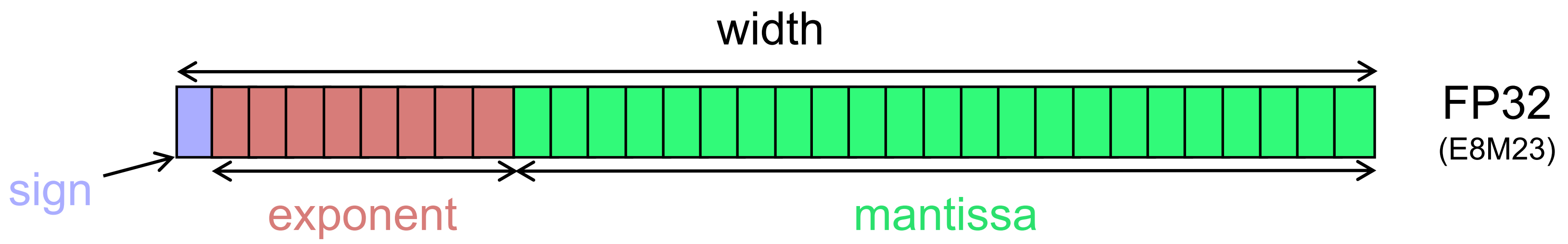
Latency


Less accurate

Efficient minifloat format for DNN inference

5.33x memory size reduction
22x more energy efficient multiplier
0.3% loss in accuracy
} over FP32

Floating-point: IEEE-754 standard



Floating expression: $(-1)^s \times 1.\underbrace{x_1 \ldots x_m}_{M_X} \times 2^{E_X - E_B}$

with $E_X \in [0, 2^e - 1]$

$M_X \in [0, 1)$

$E_B = 2^{e-1} - 1$

Special cases:

- **Zero** representation

  $E_X = 0 \;\&\; M_X = 0$

- **Subnormal** numbers

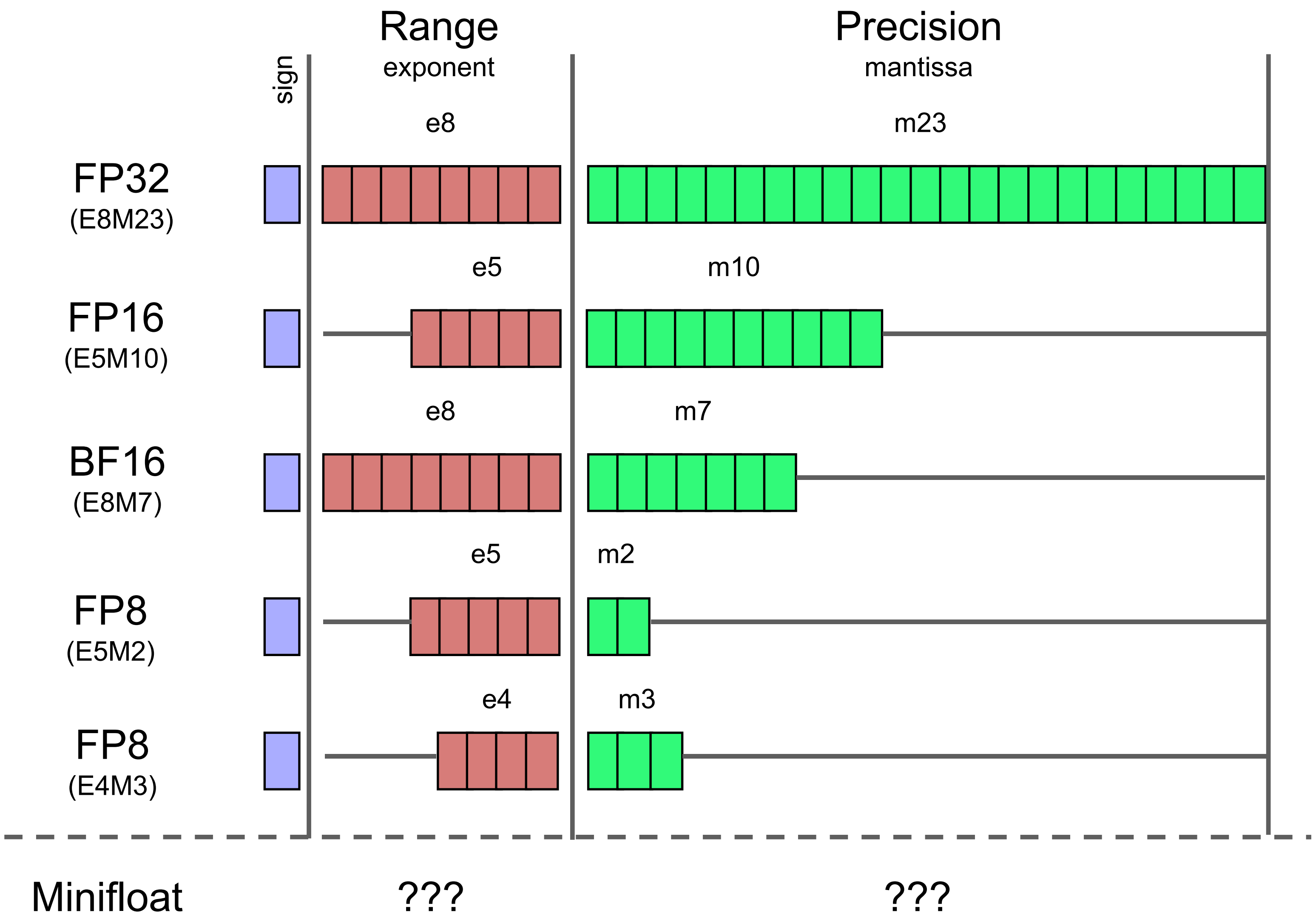
  $(-1)^s \times 0.x_1 \ldots x_{m-1}1 \times 2^{-E_B}$

- **NaN** and **Inf**

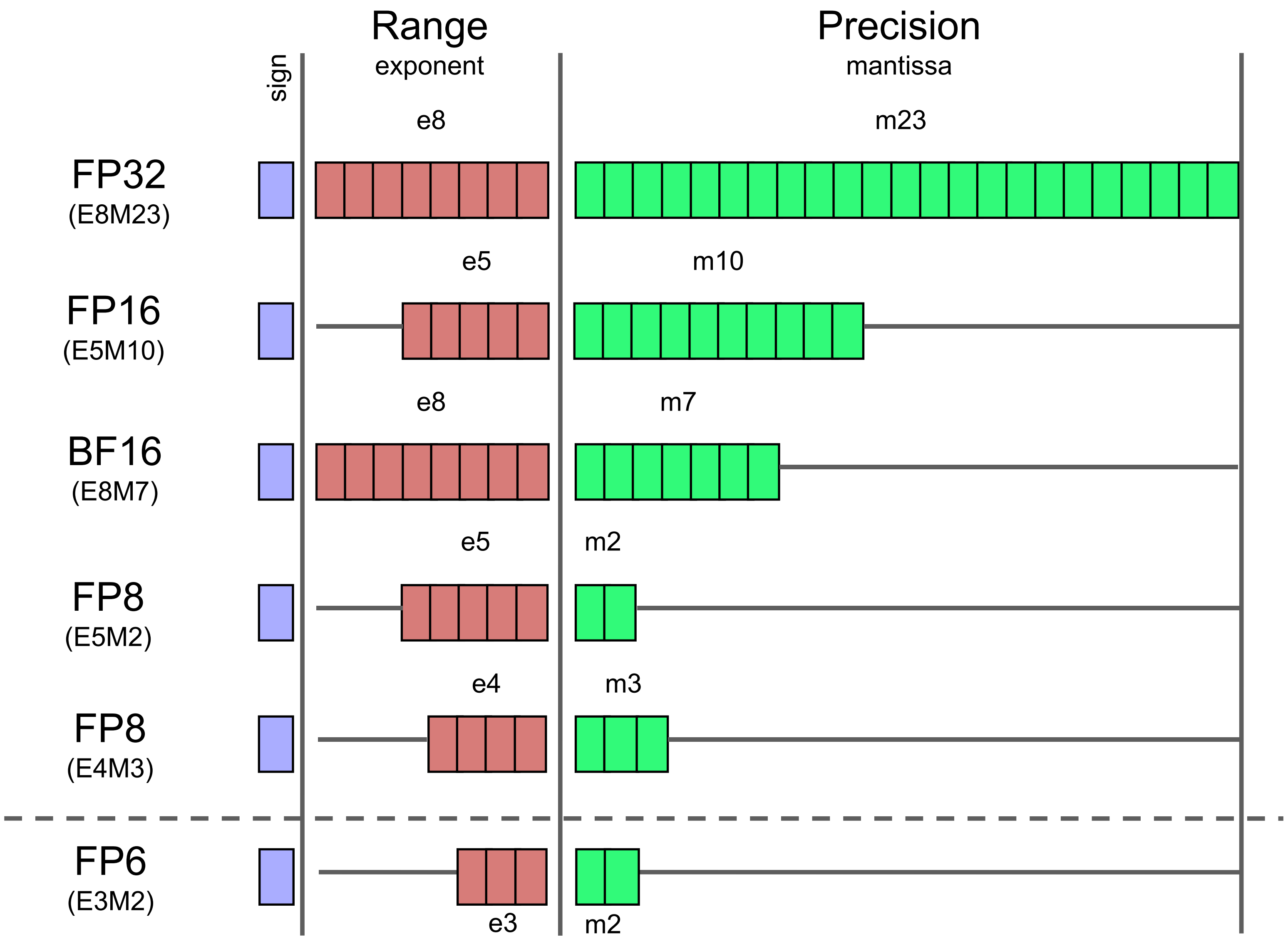
  NaN: $E_X = E_{max} \;\&\; M_X \neq 0$

  Inf: $E_X = E_{max} \;\&\; M_X = 0$

Floating-point: formats



Floating-point: formats



Floating-point: Minifloat

Minifloat expression:

$$(-1)^s \times 1.\underbrace{x_1 \dots x_m}_{M_X} \times 2^{E_X - E_B}$$

with $E_B = 2^{e-1} - 1$

Special cases:

- **Zero** representation
  $E_X = 0 \;\&\; M_X = 0$
- Not supporting **Subnormal** numbers, **NaN** and **Inf**

Floating-point: Minifloat

Minifloat expression:

$$(-1)^s \times 1.\underbrace{x_1 \ldots x_m}_{M_X} \times 2^{E_X - \lceil E_0 \rceil}$$

with E_0 **a learnable parameter**

Special cases:

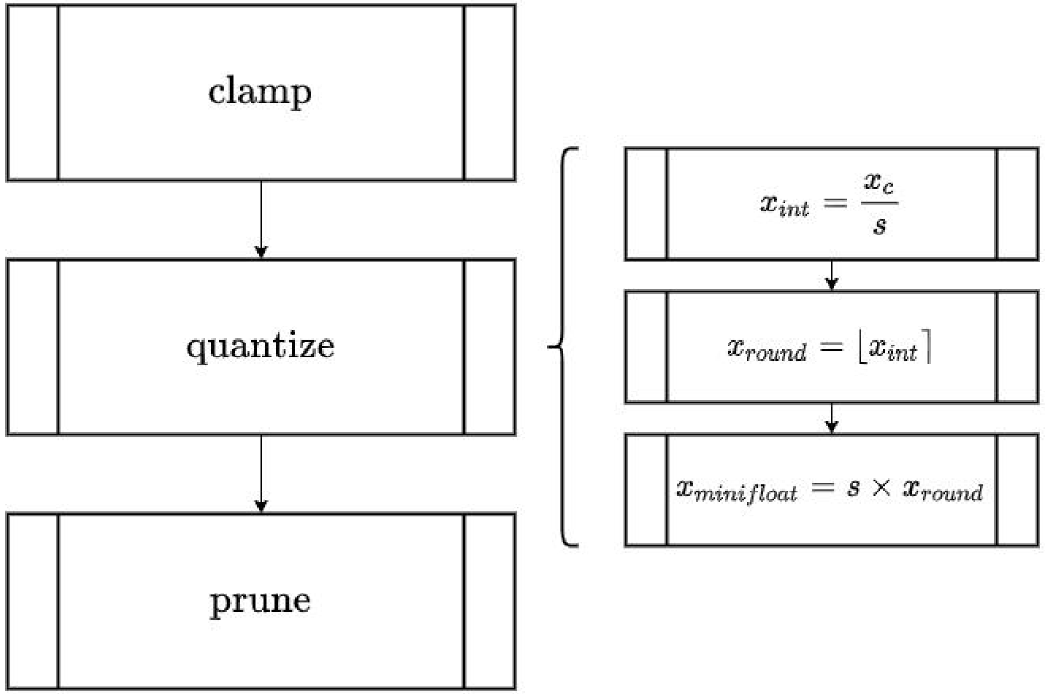
- **Zero** representation
  $E_X = 0 \; \& \; M_X = 0$
- Not supporting **Subnormal** numbers, **NaN** and **Inf**

Floating-point: Minifloat

Minifloat expression:

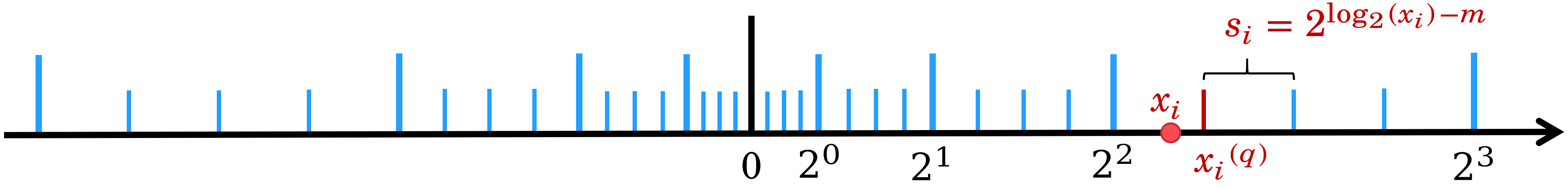
$$(-1)^s \times 1.\underbrace{x_1 \ldots x_m}_{M_X} \times 2^{E_X - \lceil E_0 \rfloor}$$

with E_0 **a learnable parameter**



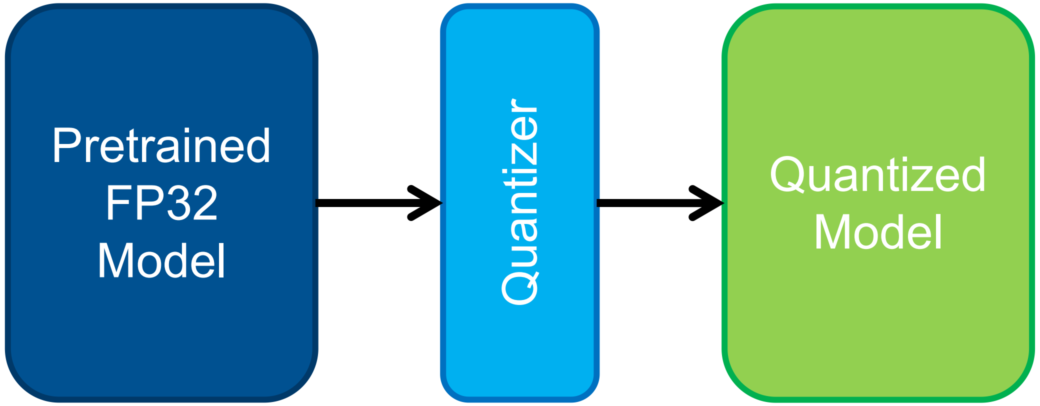
Special cases:

- **Zero** representation
  $E_X = 0 \,\&\, M_X = 0$
- Not supporting **Subnormal** numbers, **NaN** and **Inf**



Quantization approaches

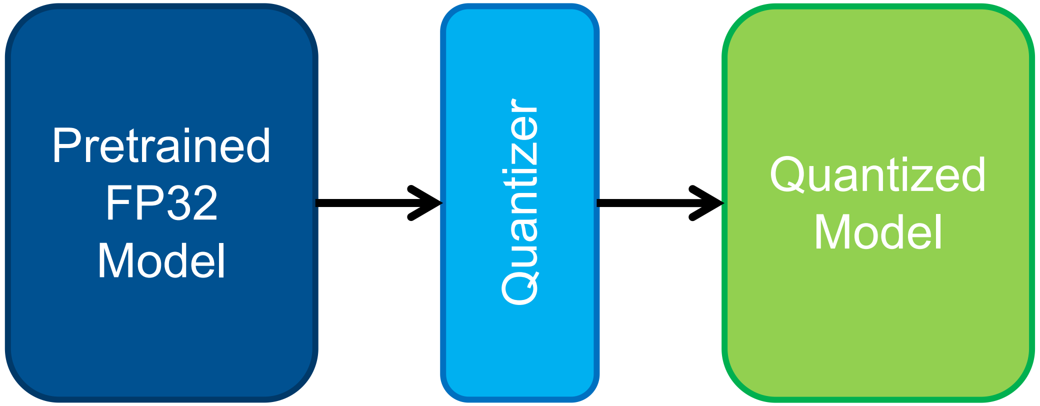
Post Training Quantization
(PTQ)



- Data free
- Low computational cost
- Accuracy loss

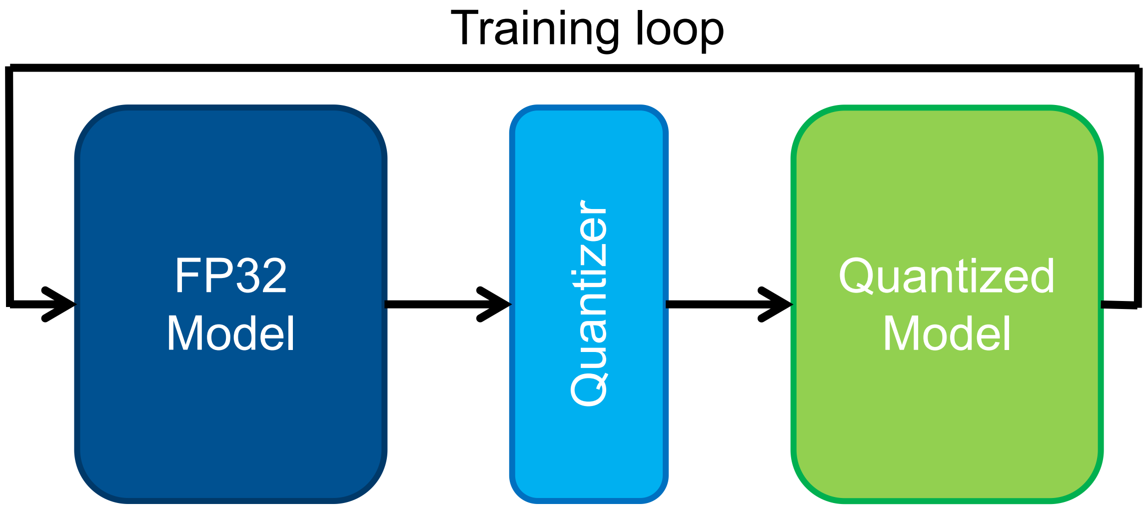
Quantization approaches

Post Training Quantization (PTQ)



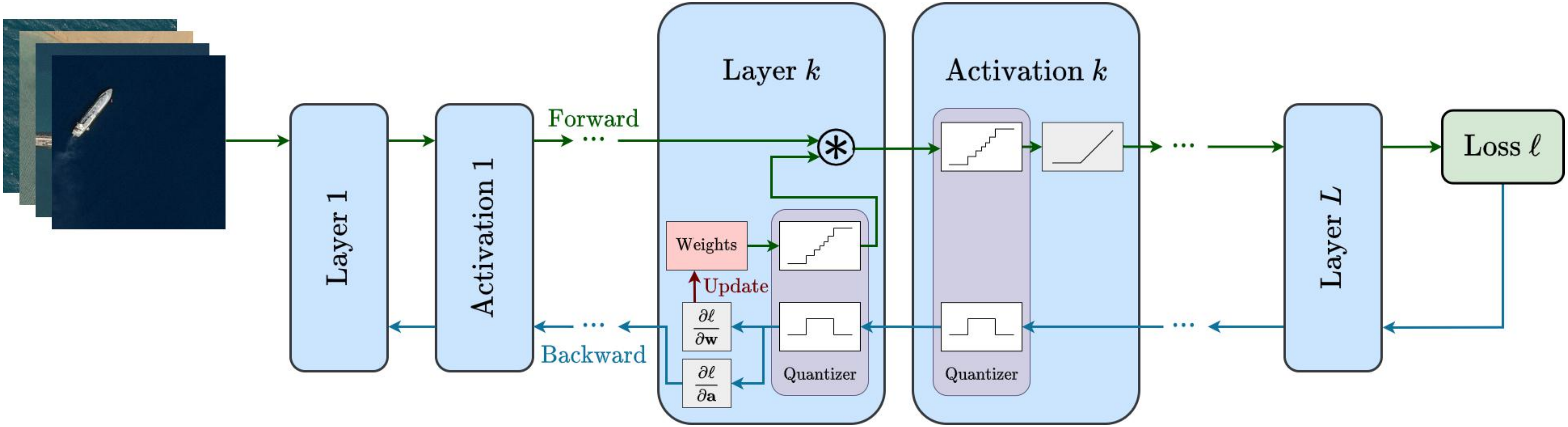
- Data free
- Low computational cost
- Accuracy loss

Quantization Aware Training (QAT)



- Better accuracy
- Computationally expensive

Quantization Aware Training



- Emulation of arithmetic operations with a floating-point quantizer
- Benefits:
  - Enables GPU acceleration
  - Flexibility of quantization format design

Image Segmentation for Ship Detection: Dataset

Airbus Ship Dataset:

- 768x768 RGB satellite images
- 192 555 labeled images
    - 150 000 empty

Training setup:

- Removal of 130 000 empty images
- Use of data augmentation

Image Segmentation for Ship Detection: Dataset

Airbus Ship Dataset:

- 768x768 RGB satellite images
- 192 555 labeled images
  - 150 000 empty

Training setup:
- Removal of 130 000 empty images
- Use of data augmentation

Image Segmentation:

Associate pixels to a defined class
- 0 background
- 1 ship

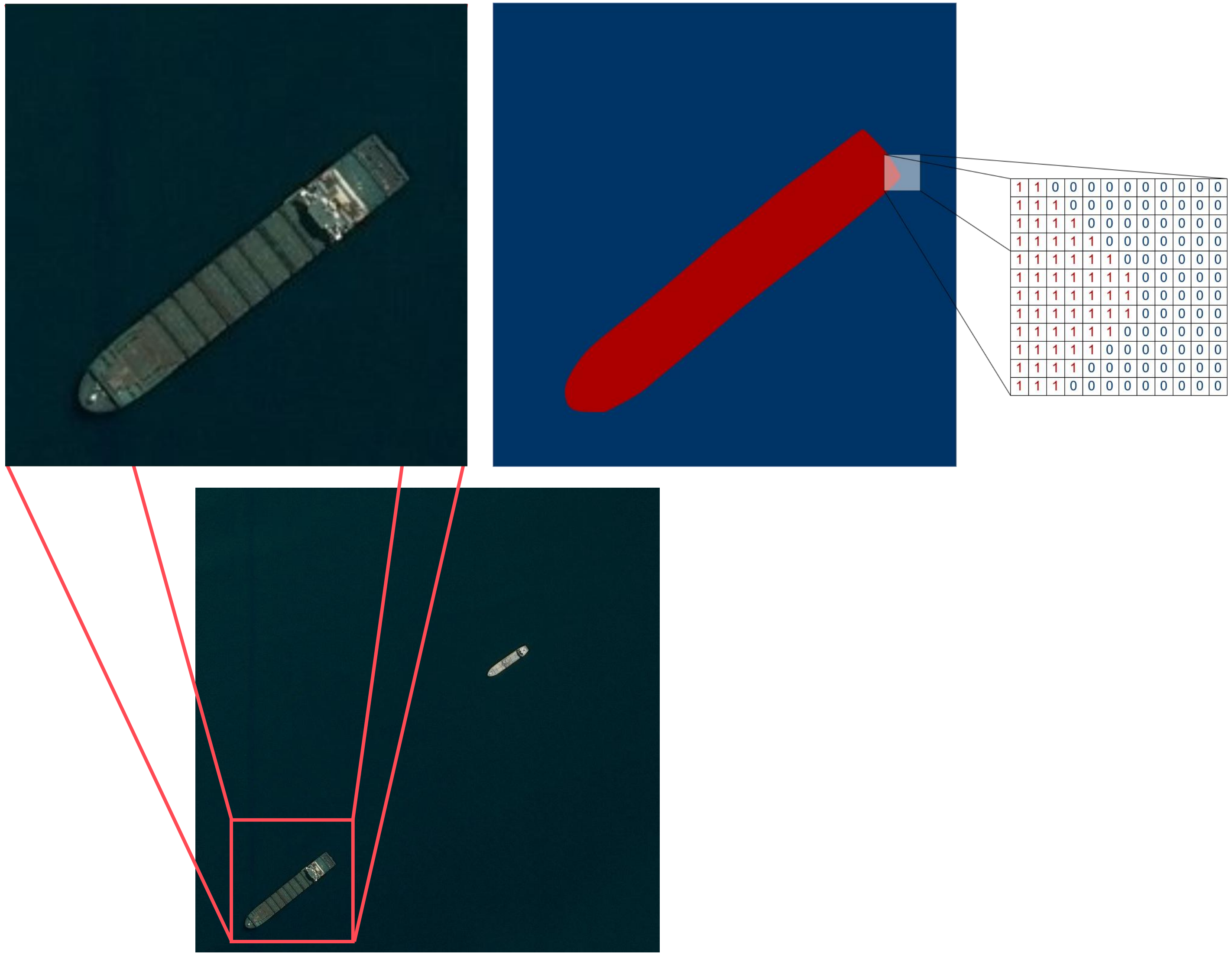


Image Segmentation for Ship Detection: Model

Thin U-Net 32 [1]:

- Small U-Net based model
  - 290x smaller
  - 32 channel depth for each convolution layer
- 5-stage encoder / 5-stage decoder

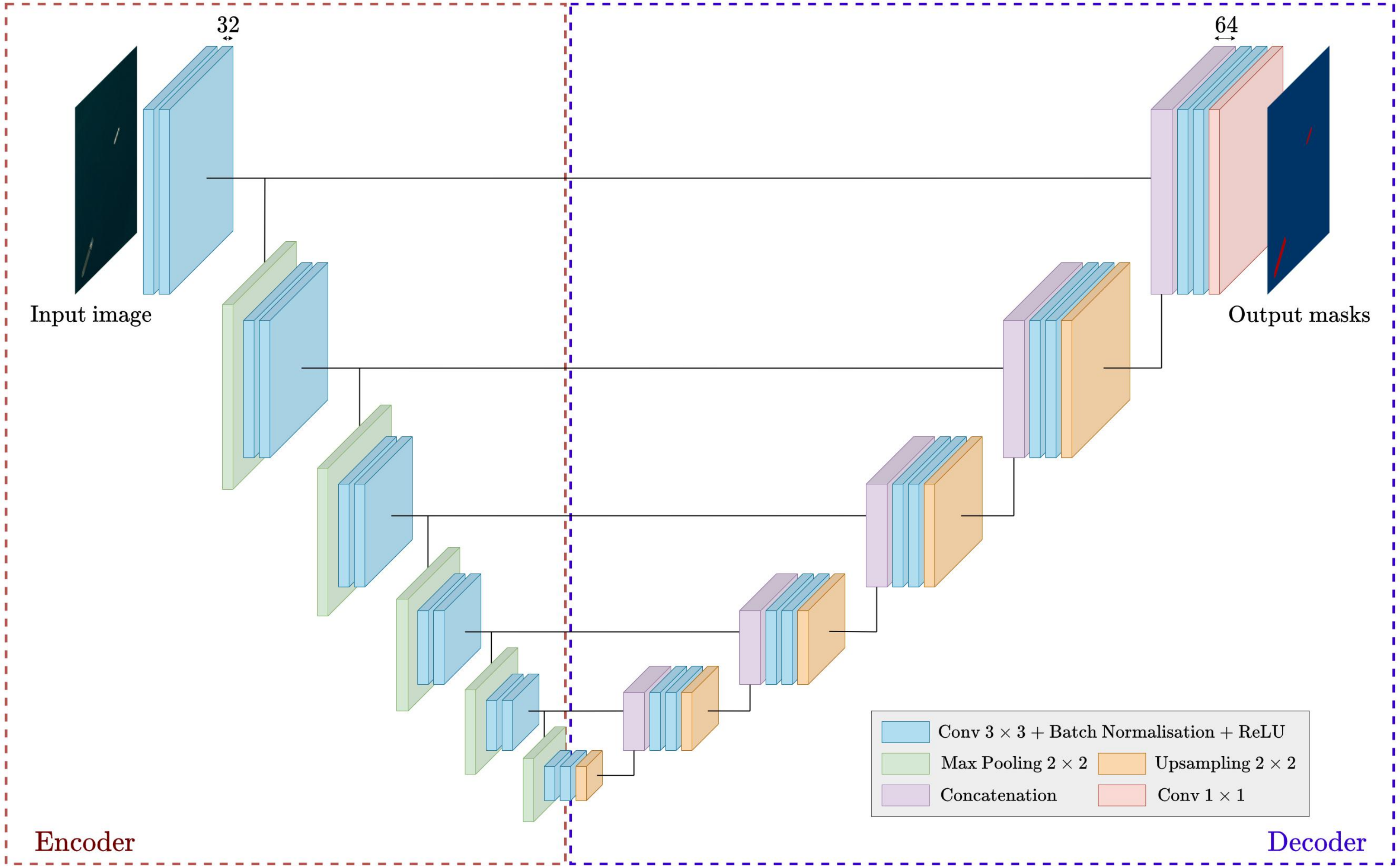


Image Segmentation for Ship Detection: Experiments

Format	mean IoU	W bit-width	A bit-width	scaling factor	zero encoding
FP32	71.0	M23E8	M23E8	/	$M_X = 0$ and $E_X = 0$
Fixed-point	44.5	6	6	$2^{\lceil \log_2(\max\lvert X\rvert)\rceil}$	Zero point $= 0$
Integer	70.5	6	5	learn	Zero point $= 0$
	68.3	5	4	learn	Zero point $= 0$
Minifloat	63.4	E3M2	E3M2	$2^{2^{e-1}}$	$E_X = 0$
	64.8	E3M2	E3M2	$2^{2^{e-1}}$	$M_X = 0$ and $E_X = 0$
	70.1	E3M3	E3M3	learn	$E_X = 0$
	70.0	E3M2	E3M2	learn	$E_X = 0$
	71.4	E4M2	E4M2	learn	$M_X = 0$ and $E_X = 0$
	70.9	E3M3	E3M3	learn	$M_X = 0$ and $E_X = 0$
	70.7	E3M2	E3M2	learn	$M_X = 0$ and $E_X = 0$
	68.1	E2M2	E2M2	learn	$M_X = 0$ and $E_X = 0$

Image Segmentation for Ship Detection: Experiments

Format	mean IoU	W bit-width	A bit-width	scaling factor	zero encoding
FP32	71.0	M23E8	M23E8	/	$M_X = 0$ and $E_X = 0$
Fixed-point	44.5	6	6	$2^{\lceil \log_2(\max\lvert X \rvert) \rceil}$	Zero point $= 0$
Integer	70.5	6	5	learn	Zero point $= 0$
	68.3	5	4	learn	Zero point $= 0$
Minifloat	63.4	E3M2	E3M2	$2^{2^{e-1}}$	$E_X = 0$
	64.8	E3M2	E3M2	$2^{2^{e-1}}$	$M_X = 0$ and $E_X = 0$
	70.1	E3M3	E3M3	learn	$E_X = 0$
	70.0	E3M2	E3M2	learn	$E_X = 0$
	71.4	E4M2	E4M2	learn	$M_X = 0$ and $E_X = 0$
	70.9	E3M3	E3M3	learn	$M_X = 0$ and $E_X = 0$
	70.7	E3M2	E3M2	learn	$M_X = 0$ and $E_X = 0$
	68.1	E2M2	E2M2	learn	$M_X = 0$ and $E_X = 0$

Image Segmentation for Ship Detection: Experiments

Format	mean IoU	W bit-width	A bit-width	scaling factor	zero encoding
FP32	71.0	M23E8	M23E8	/	$M_X = 0$ and $E_X = 0$
Fixed-point	44.5	6	6	$2^{\lceil \log_2(\max\|X\|) \rceil}$	Zero point $= 0$
Integer	70.5	6	5	learn	Zero point $= 0$
	68.3	5	4	learn	Zero point $= 0$
Minifloat	63.4	E3M2	E3M2	$2^{2^{e-1}}$	$E_X = 0$
	64.8	E3M2	E3M2	$2^{2^{e-1}}$	$M_X = 0$ and $E_X = 0$
	70.1	E3M3	E3M3	learn	$E_X = 0$
	70.0	E3M2	E3M2	learn	$E_X = 0$
	71.4	E4M2	E4M2	learn	$M_X = 0$ and $E_X = 0$
	70.9	E3M3	E3M3	learn	$M_X = 0$ and $E_X = 0$
	70.7	E3M2	E3M2	learn	$M_X = 0$ and $E_X = 0$
	68.1	E2M2	E2M2	learn	$M_X = 0$ and $E_X = 0$

Image Segmentation for Ship Detection: Experiments

Format	mean IoU	W bit-width	A bit-width	scaling factor	zero encoding
FP32	71.0	M23E8	M23E8	/	$M_X = 0$ and $E_X = 0$
Fixed-point	44.5	6	6	$2^{\lceil \log_2(\max\lvert X\rvert)\rceil}$	Zero point $= 0$
Integer	70.5	6	5	learn	Zero point $= 0$
	68.3	5	4	learn	Zero point $= 0$
Minifloat	63.4	E3M2	E3M2	$2^{2^{e-1}}$	$E_X = 0$
	64.8	E3M2	E3M2	$2^{2^{e-1}}$	$M_X = 0$ and $E_X = 0$
	70.1	E3M3	E3M3	learn	$E_X = 0$
	70.0	E3M2	E3M2	learn	$E_X = 0$
	71.4	E4M2	E4M2	learn	$M_X = 0$ and $E_X = 0$
	70.9	E3M3	E3M3	learn	$M_X = 0$ and $E_X = 0$
	70.7	E3M2	E3M2	learn	$M_X = 0$ and $E_X = 0$
	68.1	E2M2	E2M2	learn	$M_X = 0$ and $E_X = 0$

Image Segmentation for Ship Detection: Experiments

Format	mean IoU	W bit-width	A bit-width	scaling factor	zero encoding
FP32	71.0	M23E8	M23E8	/	$M_X = 0$ and $E_X = 0$
Fixed-point	44.5	6	6	$2^{\lceil \log_2(\max\|X\|) \rceil}$	Zero point $= 0$
Integer	70.5	6	5	learn	Zero point $= 0$
	68.3	5	4	learn	Zero point $= 0$
Minifloat	63.4	E3M2	E3M2	$2^{2^{e-1}}$	$E_X = 0$
	64.8	E3M2	E3M2	$2^{2^{e-1}}$	$M_X = 0$ and $E_X = 0$
	70.1	E3M3	E3M3	learn	$E_X = 0$
	70.0	E3M2	E3M2	learn	$E_X = 0$
	71.4	E4M2	E4M2	learn	$M_X = 0$ and $E_X = 0$
	70.9	E3M3	E3M3	learn	$M_X = 0$ and $E_X = 0$
	70.7	**E3M2**	**E3M2**	**learn**	$\mathbf{M_X = 0}$ **and** $\mathbf{E_X = 0}$
	68.1	E2M2	E2M2	learn	$M_X = 0$ and $E_X = 0$

Image Segmentation for Ship Detection: Experiments

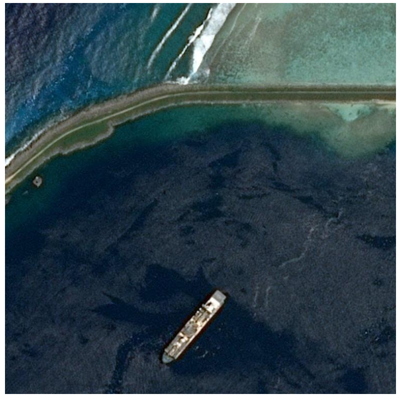
Format	mean IoU	W bit-width	A bit-width	scaling factor	zero encoding
FP32	71.0	M23E8	M23E8	/	$M_X = 0$ and $E_X = 0$
Fixed-point	44.5	6	6	$2^{\lceil \log_2(\max \lvert X \rvert) \rceil}$	Zero point $= 0$
Integer	70.5	6	5	learn	Zero point $= 0$
	68.3	5	4	learn	Zero point $= 0$
Minifloat	63.4	E3M2	E3M2	$2^{2^{e-1}}$	$E_X = 0$
	64.8	E3M2	E3M2	$2^{2^{e-1}}$	$M_X = 0$ and $E_X = 0$
	70.1	E3M3	E3M3	learn	$E_X = 0$
	70.0	E3M2	E3M2	learn	$E_X = 0$
	71.4	E4M2	E4M2	learn	$M_X = 0$ and $E_X = 0$
	70.9	E3M3	E3M3	learn	$M_X = 0$ and $E_X = 0$
	70.7	**E3M2**	**E3M2**	**learn**	$\mathbf{M_X = 0}$ **and** $\mathbf{E_X = 0}$
	68.1	E2M2	E2M2	learn	$M_X = 0$ and $E_X = 0$

Original image

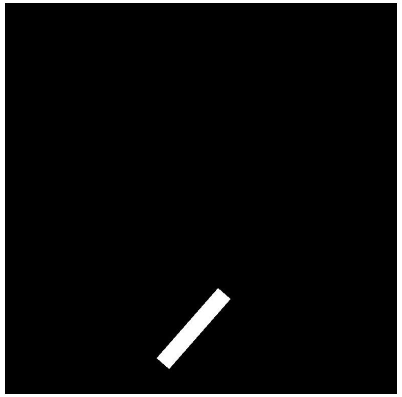
Ground truth

E3M2 Minifloat

Original image



Ground truth



E3M2 Minifloat

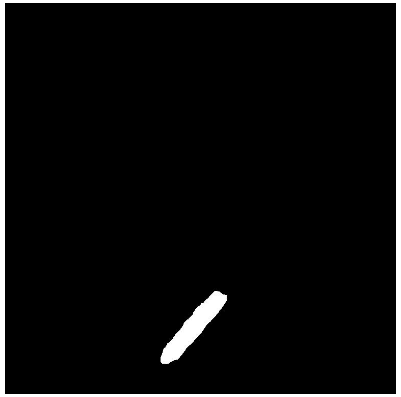


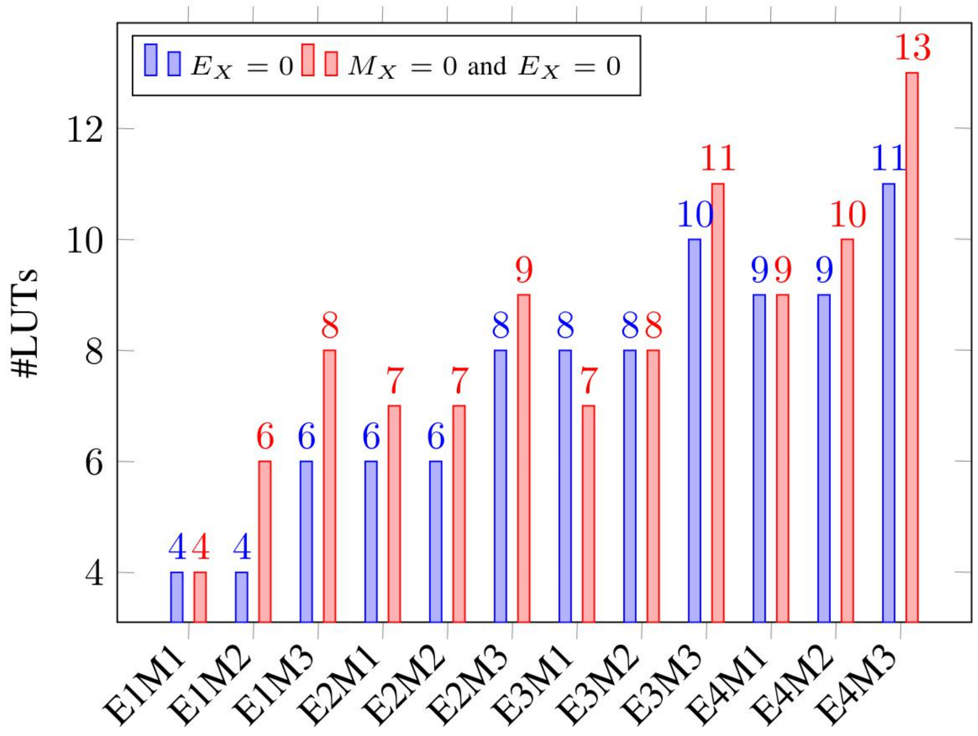
Image Segmentation for Ship Detection: Experiments

Format	mean IoU	W bit-width	A bit-width	scaling factor	zero encoding
FP32	71.0	M23E8	M23E8	/	$M_X = 0$ and $E_X = 0$
Fixed-point	44.5	6	6	$2^{\lceil \log_2(\max \lvert X \rvert) \rceil}$	Zero point $= 0$
Integer	70.5	6	5	learn	Zero point $= 0$
	68.3	5	4	learn	Zero point $= 0$
Minifloat	63.4	E3M2	E3M2	$2^{2^{e-1}}$	$E_X = 0$
	64.8	E3M2	E3M2	$2^{2^{e-1}}$	$M_X = 0$ and $E_X = 0$
	70.1	E3M3	E3M3	learn	$E_X = 0$
	70.0	E3M2	E3M2	learn	$E_X = 0$
	71.4	E4M2	E4M2	learn	$M_X = 0$ and $E_X = 0$
	70.9	E3M3	E3M3	learn	$M_X = 0$ and $E_X = 0$
	70.7	**E3M2**	**E3M2**	**learn**	$\mathbf{M_X = 0}$ **and** $\mathbf{E_X = 0}$
	68.1	E2M2	E2M2	learn	$M_X = 0$ and $E_X = 0$

Original image Ground truth E3M2 Minifloat Original image Ground truth E3M2 Minifloat

Hardware Implementation Aspects

- **60% higher performance** and **12.5% memory traffic reduction** for E3M3 minifloat over INT8 [2]
  - Use of a **hybrid MAC operator**: **LUT-based minifloat multiplier** and **fixed-point adder**.
  - Key details: **real-valued scaling factor** with a **symmetric exponent bias** and **zero encoding as E_X = 0**



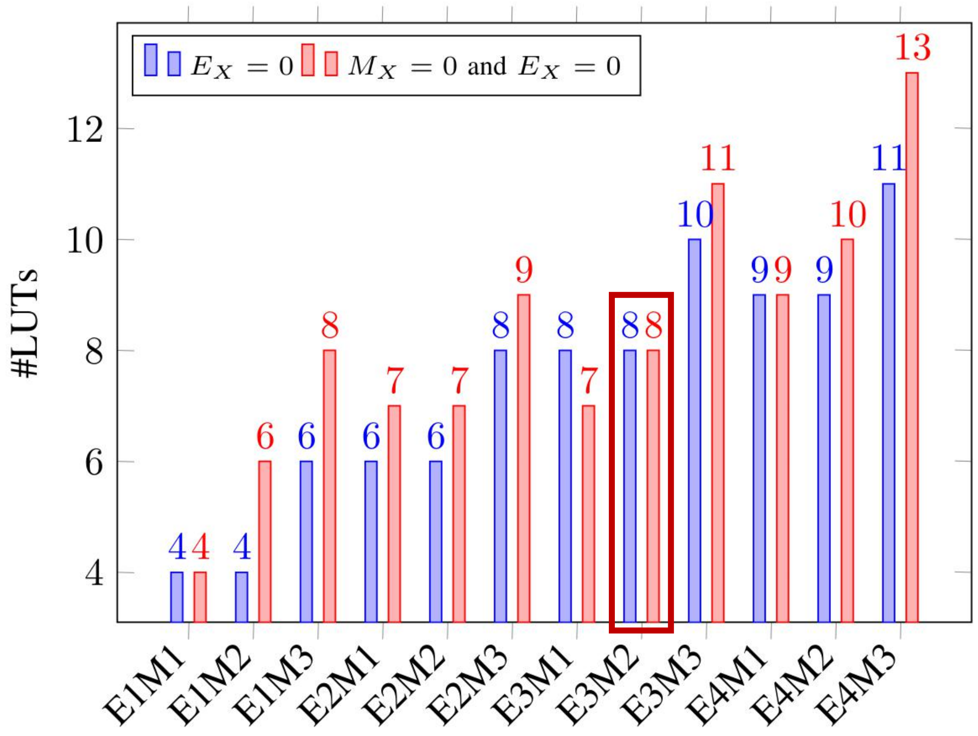
LUT consumption of a minifloat multiplier

- Our Minifloat format reduces the scaling logic required and the zero encoding slightly increases the LUT usage.

[2] P. Metzgen, S. Fang, S. Pareek, B. Tian, Y. Shan, E. Delaye, and A. Sirasao. Higher Performance Neural Networks with Small FloatingPoint. Technical Report WP530, AMD-Xilinx, 2021

Hardware Implementation Aspects

- **60% higher performance** and **12.5% memory traffic reduction** for E3M3 minifloat over INT8 [2]
  - Use of a **hybrid MAC operator**: **LUT-based minifloat multiplier** and **fixed-point adder**.
  - Key details: **real-valued scaling factor** with a **symmetric exponent bias** and **zero encoding as E_X = 0**



LUT consumption of a minifloat multiplier

- Our Minifloat format reduces the scaling logic required and the zero encoding slightly increases the LUT usage.

[2] P. Metzgen, S. Fang, S. Pareek, B. Tian, Y. Shan, E. Delaye, and A. Sirasao. Higher Performance Neural Networks with Small FloatingPoint. Technical Report WP530, AMD-Xilinx, 2021

Conclusion

- Propose a **QAT algorithm** to train **low-precision floating-point**
  - **learnable exponent bias** at layer granularity for both weights and activations

- Experiments on Airbus Ship dataset show good results: **E3M2 minifloat** model is competitive with **single precision baselines** and **INT6**

- Propose an **efficient minifloat multiplier** implementation -> basis for a full DNN inference accelerator

Future work

⟶ Test and deploy a quantized Thin U-Net 32 accelerator on FPGA targets

Thank you for your attention

contact: cedric.gernigon@inria.fr

Questions