Benchmarking New Hardware for Machine Learning in Particle Physics

Fermilab's New Perspectives 2020, July 20th

Stefano Vergani - sv408@hep.phy.cam.ac.uk

OVERVIEW

1. Professional details and introduction
2. A few words about processors
3. The dataset
4. TensorFlow Lite
5. Results, comments, and future work
6. Q & A

PROFESSIONAL DETAILS

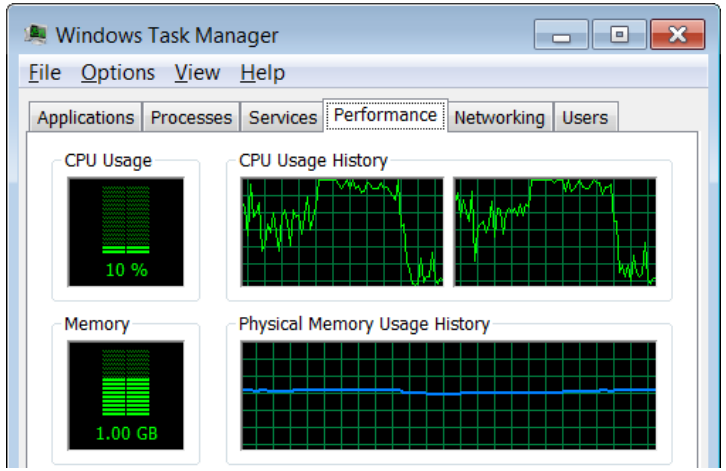
- PhD student in High Energy Physics at the University of Cambridge,
- BSc in Physics (Milan,), MSc in Physics (ETH Zurich,)
- Supervisors: Leigh Whitehead and Melissa Uchida (Cambridge), Michael Wang (Fermilab)
- Works in the Pandora team -> Software development for Pandora, data analysis for ProtoDUNE-SP, specialised in Machine Learning
- email: sv408@hep.phy.cam.ac.uk

GOAL OF THE PROJECT

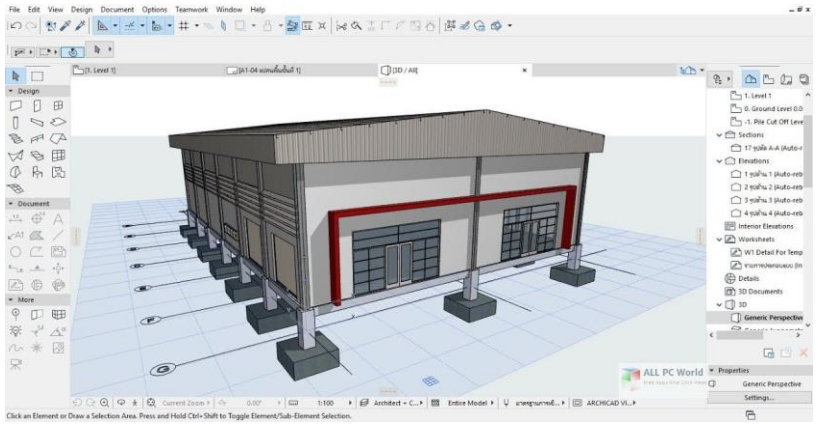
Should we buy one Edge TPU for each desktop at Fermilab and perform inference with it?

CPU vs GPU vs NPU

Central Processing Unit



Graphics Processing Unit



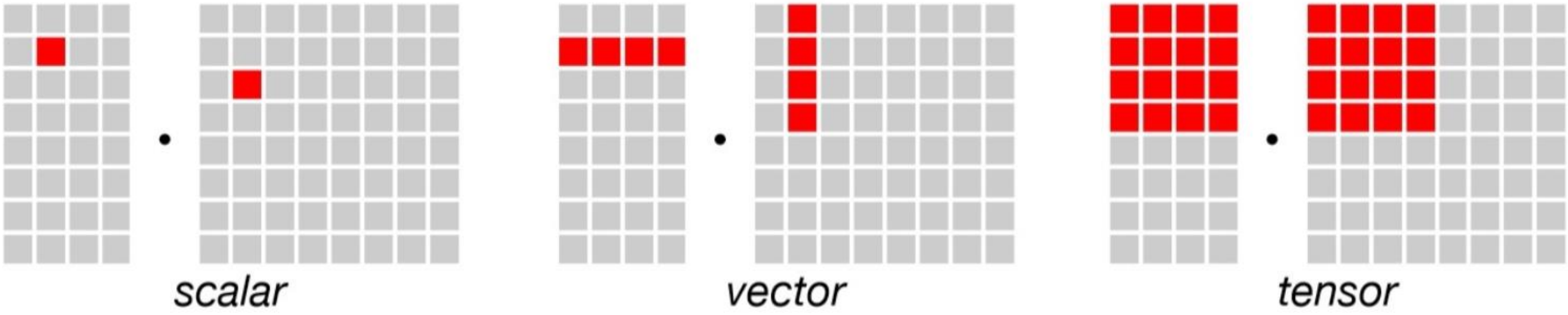
Neural Processing Unit

CPU vs GPU vs TPU

CPU O(10) operation per cycle

GPU O(10^4) operation per cycle

TPU O(128*10^3) operation per cycle



source https://iq.opengenus.org/cpu-vs-gpu-vs-tpu/

EDGE COMPUTING

Why Edge? *Edge computing is the practice of processing data near the edge of your network, where the data is being generated, instead of in a centralised data-processing warehouse.*

30 mm

65 mm

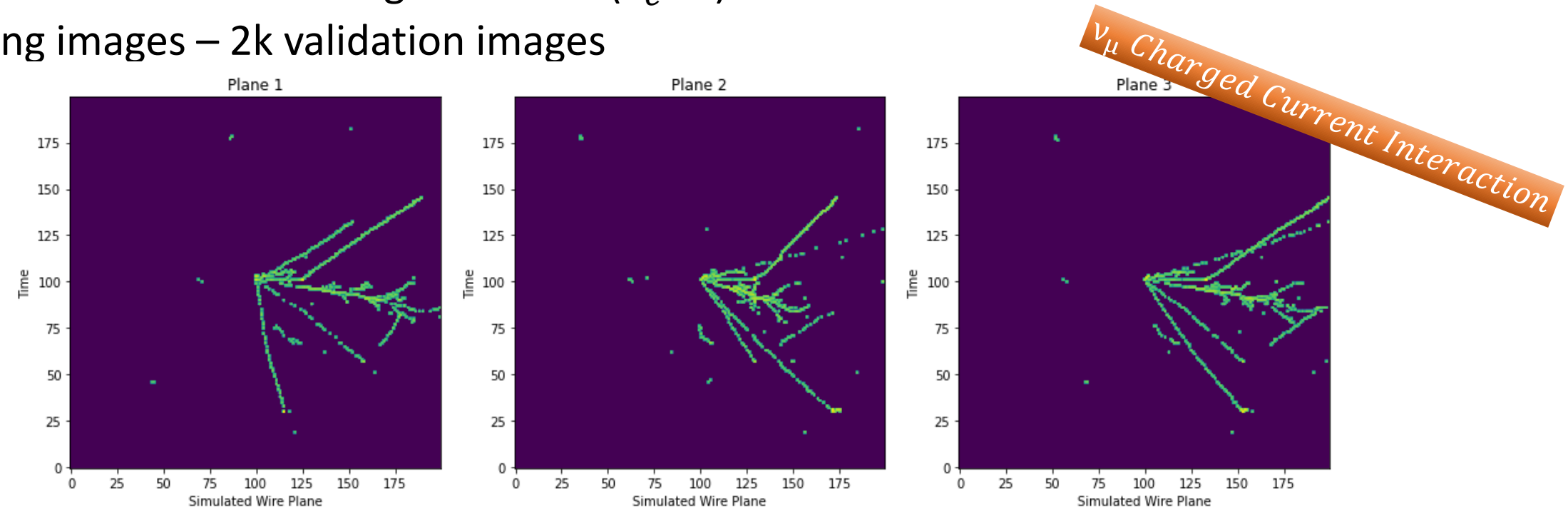
SPECIFICATIONS

	CPU	GPU	Edge TPU
Model	Intel ® Core ™ i7-8550U @ 1.8 GHz (4 cores)	NVIDIA Tesla P100 16 GB	Coral Edge TPU
TDP* (w)	15	250	2
Price (USD)	409	7500 or 9.99/month	80

*Thermal Design Power (TDP) represents the average power, in watts, the processor dissipates when operating at Base Frequency with all cores active under an Intel-defined, high-complexity workload.

NEUTRINO SIMULATED DATASET

- Liquid Argon Time-Projection Chamber (LArTPC) simulated images
- 3 simulated wire planes, 200x200 pixels which mimic wire readouts
- 3 classes of interaction: neutrino neutral current (NN) , muon neutrino charged current (ν_μCC), electron neutrino charged current (ν_eCC)
- 10k training images – 2k validation images



TensorFlow Lite

How it works

Pick a model

Pick a new model or retrain an existing one.

Convert

Convert a TensorFlow model into a compressed flat buffer with the TensorFlow Lite Converter.

Deploy

Take the compressed .tflite file and load it into a mobile or embedded device.

Optimize

Quantize by converting 32-bit floats to more efficient 8-bit integers or run on GPU.

Taken from https://www.tensorflow.org/lite

QUANTISATION AND OPTIMISATION IN TF-LITE

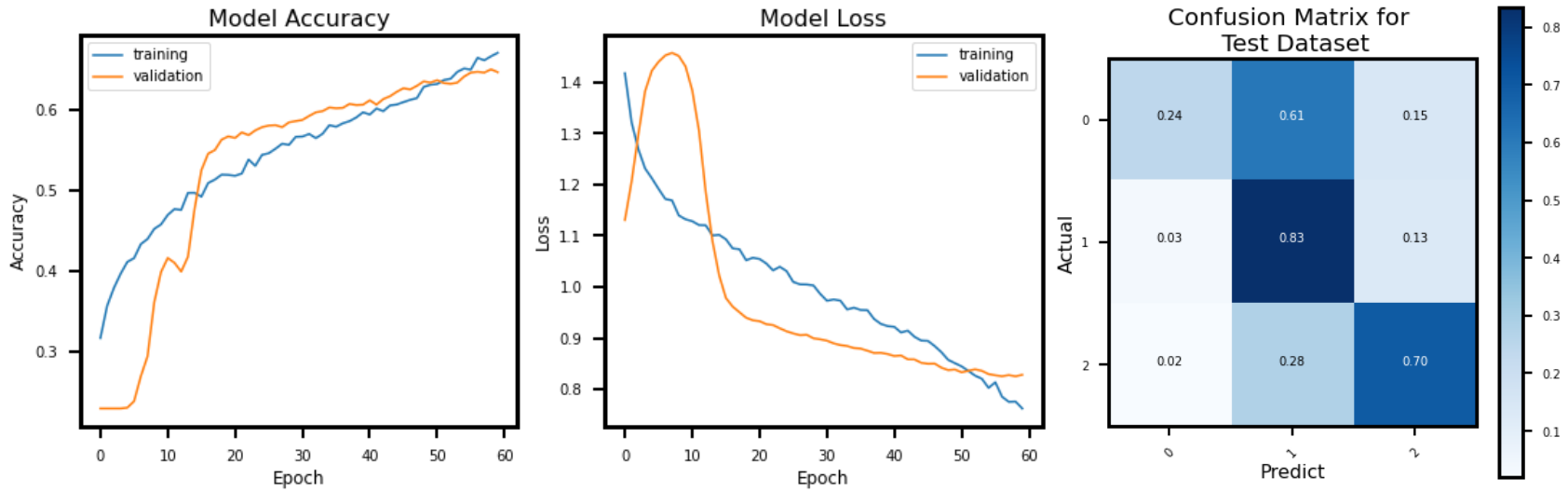
TF Lite Optimisation	Supported Hardware	Size Reduction with respect to TF	Tensor Type	Weights	Activations
Not Optimised	CPU, GPU (Android)	~70%	Float 32	float32	float32
Post-Training Dynamic Range Quantisation	CPU, GPU (Android)	~90%	Float 32	int8	float32
Post-Training Float16 Quantisation	CPU, optimised for GPU (Android)	~80%	Float 32	float 16	float32
Post-Training Integer Quantisation	CPU, GPU, Edge TPU (model must be specifically compiled)	~90%	Float 32 or Int 8 or Uint 8 (Edge TPU only with Uint 8)	int8	int8

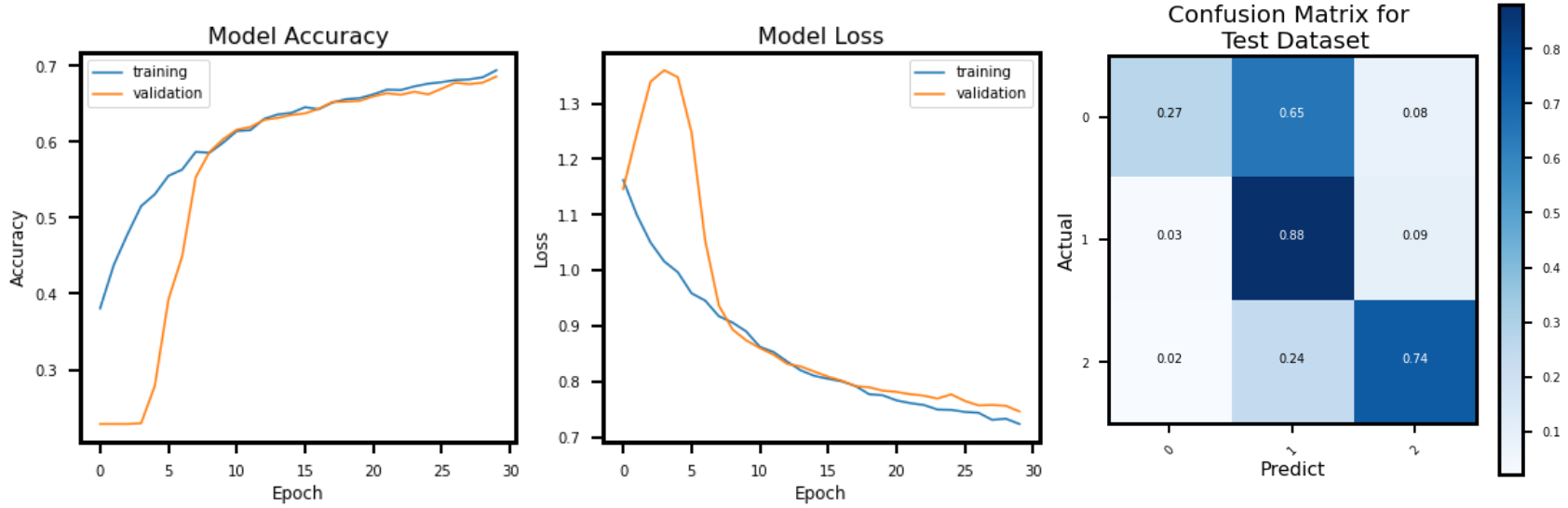
Quantisation is a conversion technique that can reduce model size while also improving CPU and hardware accelerator latency, with little degradation in model accuracy.

QUANTISATION AWARE TRAINING

- Smallest accuracy loss (close to none)
- 90% size reduction
- weights and activations int8
- Coming soon…

ResNet-50 V2 and DenseNet-169





ResNet-50 V2

Validation accuracy 64.15%

Confusion Matrix Values:

0. NN
1. ν_μCC
2. ν_eCC

DenseNet-169

Validation accuracy 68.6%

RESULTS for ResNet-50 V2

Optimisation	Size (MB)	Accuracy (same for all hardware)	Speed on CPU	Speed on GPU	Speed on Edge TPU
TensorFlow	274	64.15%	40.88 ms	4.91 ms	-
TF Lite Not Optimised	90	64.15%	89.36 ms	94.4 ms	-
Post-Training Dynamic Range Quantisation	23	54.35%	548.54 ms	314.76 ms	-
Post-Training Float 16 Quantisation	45	64.15%	114.16 ms	64.05 ms	-
Post-Training Integer Quantisation	24	N.A. (predicted 60-62%)	6.1 s	9.3 s	42.41 ms

RESULTS for DenseNet-169

Optimisation	Size (MB)	Accuracy (same for all hardware)	Speed on CPU	Speed on GPU	Speed on Edge TPU
TensorFlow	157	68.6%	48.04 ms	1.83 ms	-
TF Lite Not Optimised	48	68.6%	81.02 ms	112.14 ms	-
Post-Training Dynamic Range Quantisation	13	63.65%	530.95 ms	296.88 ms	-
Post-Training Float 16 Quantisation	25	68.6%	81.92 ms	114.21 ms	-
Post-Training Integer Quantisation	13	N.A. (predicted 65–67%)	7.6 s	7.2 s	23.67 ms

Costs

K function = TDP*time/inference [w*ms]

ResNet-50 V2

- best CPU = 613.2
- best GPU = 1227.5
- best Edge TPU = 84.42

DenseNet-169

- best CPU = 720.6
- best GPU = 457.5
- best Edge TPU = 47.34

RESULTS

- In terms of pure performance, GPU appears to be by far the fastest piece of hardware.
- In comparison to CPU, Edge TPU is faster and costs less money.
- In comparison to GPU, Edge TPU is one order of magnitude slower but one order of magnitude cheaper
- Some TensorFlow-Lite optimizations could be very useful for old/not expensive pieces of hardware

FUTURE WORK

- Complete tests with all of the Keras models available
- Solve accuracy issue with Post-Training Quantisation and test Quantisation-Aware Training as soon as available
- Publication

Thank you for your attention!

Q & A

BACK-UP: BENCHMARKED NETWORKS

Table 1. Time per inference, in milliseconds (ms)

Model architecture	Desktop CPU [1]	Desktop CPU [1] + USB Accelerator (USB 3.0) *with Edge TPU*	Embedded CPU [2]	Dev Board [3] *with Edge TPU*
DeepLab[4]* (513x513)	301	35	1210	156
DenseNet* (224x224)	298	20	1035	25
Inception v1 (224x224)	92	3.6	406	3.9
Inception v4 (299x299)	792	100	3,463	100
Inception-ResNet V2 (299x299)	703	57	3082	69
MobileNet v1 (224x224)	47	2.2	179	2.2
MobileNet v2 (224x224)	45	2.3	150	2.5
MobileNet v1 SSD (224x224)	95	6.5	380	11
MobileNet v2 SSD (224x224)	88	7.2	314	14
ResNet-50 V1 (299x299)	458	48	1944	57
ResNet-50 V2 (299x299)	557	50	2009	59
ResNet-152 V2 (299x299)	1652	128	6053	151
SqueezeNet (224x224)	55	2	253	2
VGG16 (224x224)	1106	296	5068	343
VGG19 (224x224)	1216	308	6174	357
EfficientNet-EdgeTpu-S (224x224)	4684	4.9	4642	4.9
EfficientNet-EdgeTpu-M (240x240)	7174	8.5	7223	9.0
EfficientNet-EdgeTpu-L (300x300)	18736	25.4	18937	25.4

BACK-UP: MNIST, FASHION MNIST, CIFAR10 with Post-Training

MNIST: 56x56 pixel pictures, 10 labels – 2% accuracy loss
FASHION MNIST: 56x56 pixel pictures, 10 labels – 2% accuracy loss
CIFAR 10: 224x224 pixel pictures 2 channels, 10 labels – 3% accuracy loss