

# ACCELERATING DEEP LEARNING RECONSTRUCTION USING CMS OPEN DATA

Daide Di Croce  
University of Alabama

# Outline

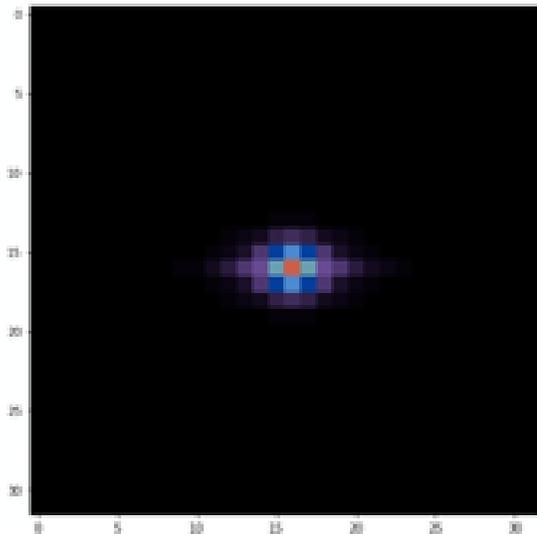
---

- Introduction
- End-To-End Deep Learning
- Benchmark E2E models on CPU
- Benchmark E2E models on different hardware architectures
- Single-GPU training comparison
- Scaling E2E training with multiple GPUs
- Conclusion

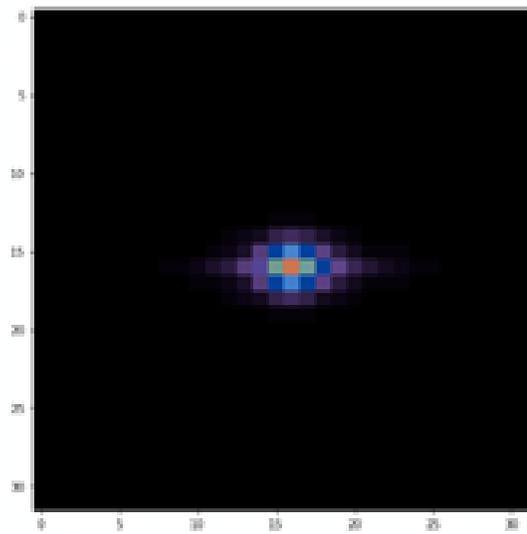
# Introduction

- PF approach converts raw detector data into physically-motivated quantities until arriving at particle-level data. This method is dependent on the full understanding of particle decay phenomenology.
- Deep Learning algorithms can be trained directly from raw data and learning the pertinent features **unassisted**: the End-to-End Deep Learning approach
- We have developed End-to-End Deep Learning approach for particle and event reconstruction, data analysis and simulation.

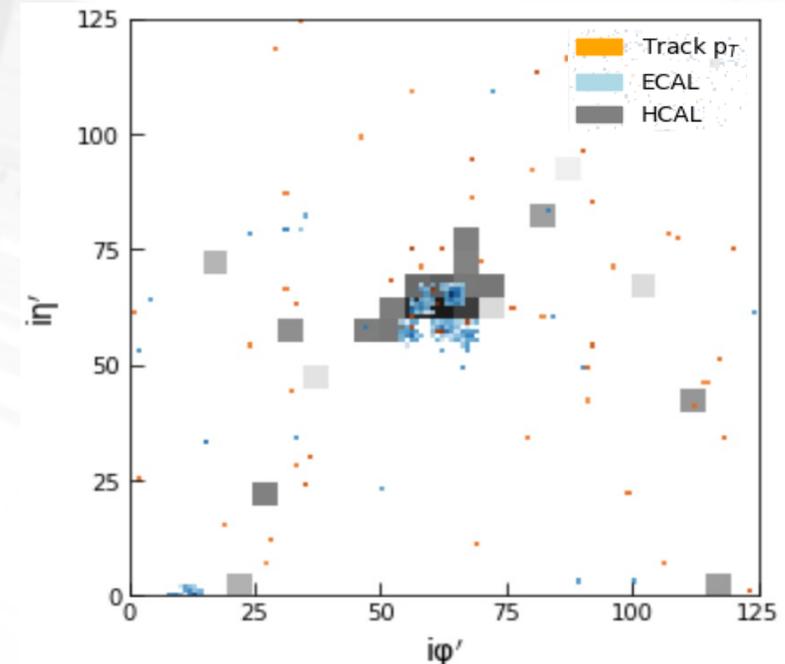
Photon-Induced EM Shower  
mean energy distribution over 10k events



Electron-Induced EM Shower  
mean energy distribution over 10k events



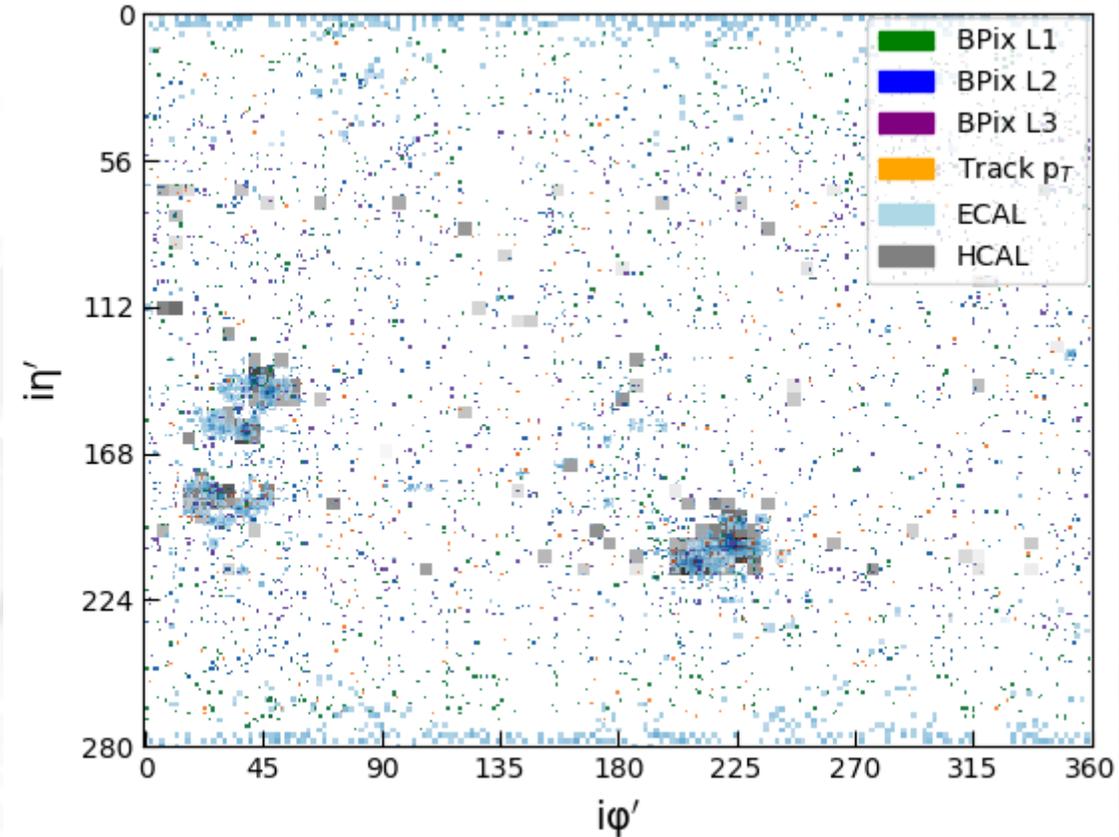
Photon (left) and electron (right) image - 1 channel: ECAL



Gluon-jet image - 3 channels: track  $p_T$ , ECAL & HCAL

# E2E deep learning

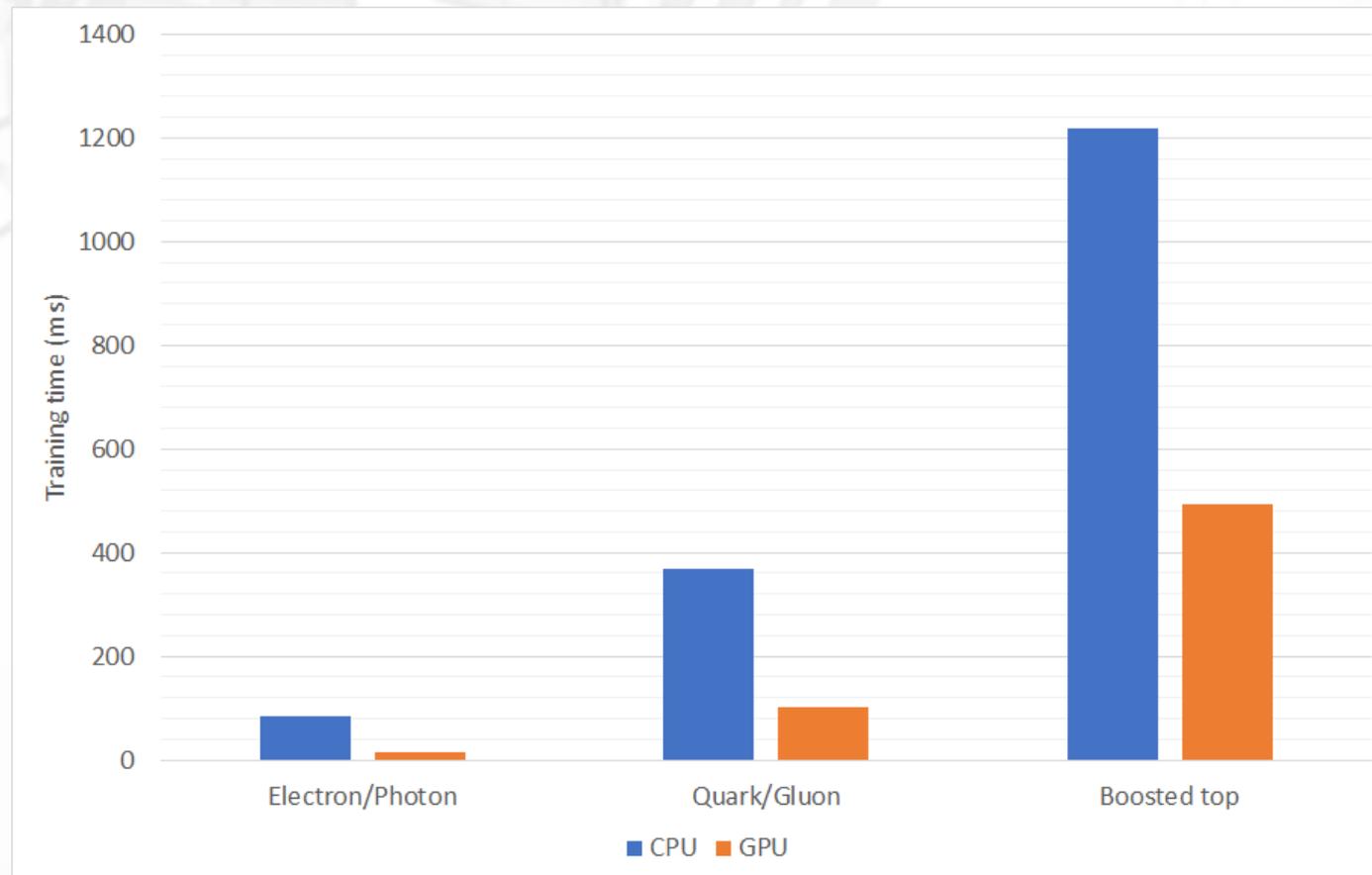
- End-to-End Deep Learning applications:
  - Single particle reconstruction: electron, photon
  - Jet classification: quark, gluon, boosted top, tau
  - Event reconstruction/classification:  $H \rightarrow aa \rightarrow \gamma\gamma\gamma\gamma$
- We demonstrate the E2E implementation and perform studies on different E2E benchmarks using CMS Run 1 open data
  - In this presentation, we will focus on the study of various hardware architectures with the E2E top quark benchmark, as it is more complex and uses 8 channels: track  $p_T$ ,  $d_0$  and  $d_z$ , pixel layers, ECAL and HCAL



$t\bar{t}$  event image - 8 channels:  
track  $p_T$ ,  $d_0$  and  $d_z$ , pixel layers, ECAL & HCAL

# CNN E2E training benchmark

- E2E electron/photon benchmark: 1 channel (ECAL)
- E2E quark/gluon benchmark: 3 channels (track  $p_T$ , ECAL & HCAL)
- E2E top quark jet benchmark: 8 channels (track  $p_T$ ,  $d_0$  and  $d_z$ , pixel layers, ECAL & HCAL)



# CNN E2E training benchmark

---

- We use the CNN E2E top quark benchmark to compare the training performance on single and multiple GPUs and TPU.

Cluster	Processor	CPU	Storage	HBM memory	Performance
Fermilab LPC	Tesla P100	Intel Xeon Silver 4110 8-core	HGST 1W10002 HD	16 GB	9.3 Single-Precision TeraFLOPS
NVIDIA Raplab	Tesla V100	4 Intel Xeon Gold 5118 12-core	SSD	32 GB	125 Mixed-Precision TeraFLOPS
Google Cloud	TPUv3-8	TPU	Google Cloud	128 GB	520 Mixed-Precision TeraFLOPS

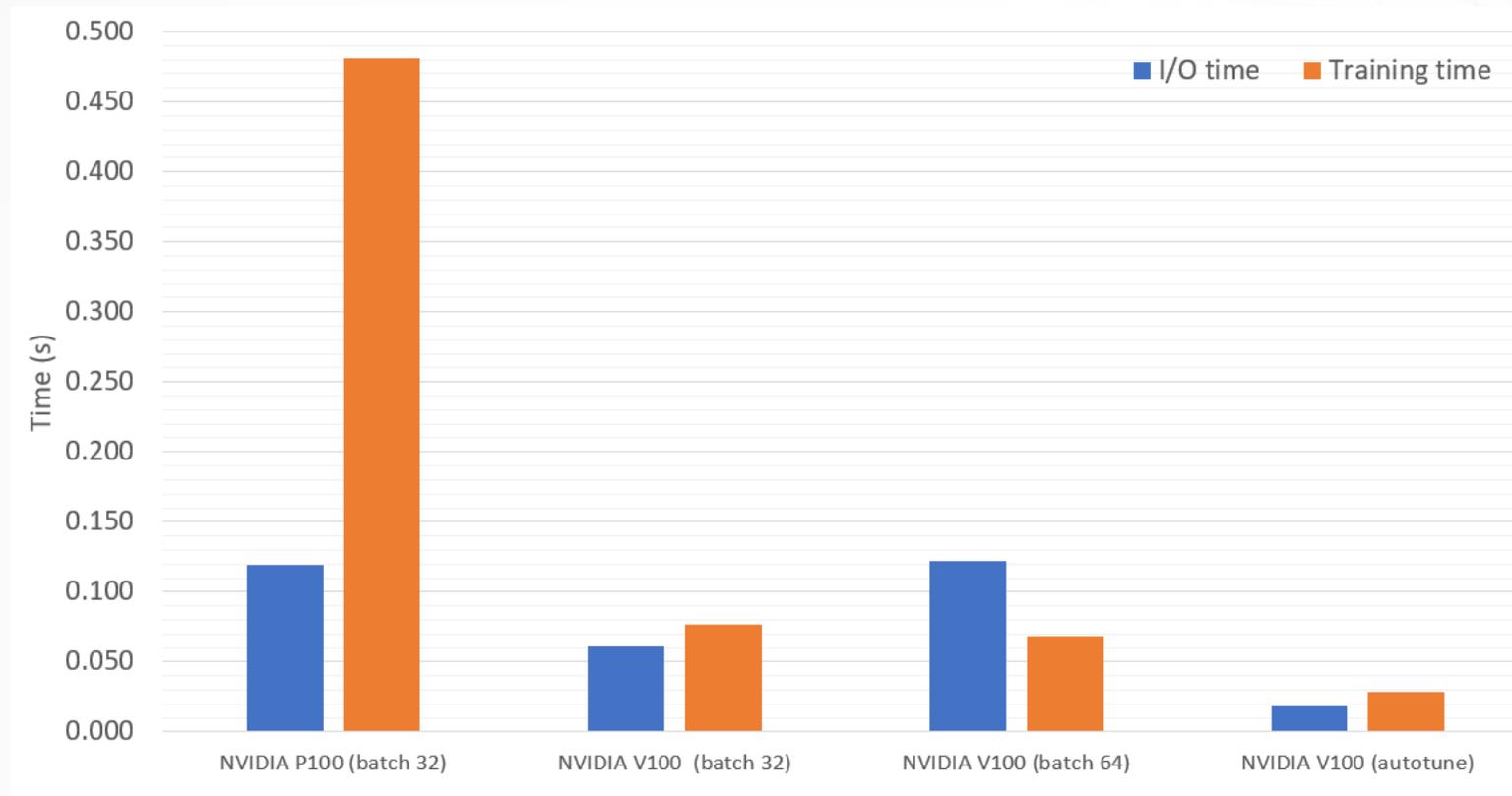
- We used Fermilab LPC, Google Cloud and NVIDIA Raplab clusters to evaluate the performance of the ML models on different hardware architectures (CPUs, GPUs and TPU)

# Benchmark E2E training on different architectures

Comparison of I/O and training time for different computing architectures				
Config.	Processor	Tesla P100	Tesla V100	TPUv3-8
Config.	Batch size	32	64	64
	Bathes per epoch	80 k	40 k	40 k
	I/O time (1 batch)	0.119 s	0.018 s	0.018 s
x1 res	Train time (1 epoch)	321 min	19 min	14 min
x3 res	I/O time (1 batch)	0.833 s	0.063 s	0.189 s
	Train time (1 epoch)	1663 min	105 min	131 min

- Tesla V100 takes advantage of SSD storage which provides higher I/O speed
- Tesla P100: fewer CPU nodes when fetching batches and sending to GPU (data load bottleneck)
- TPUv3-8 spends 2.6 ms on forward and back propagation calculations, that is 4 x faster than the V100
- Tesla V100 and TPUv3-8 provide stronger data loading and training performance compared to Tesla P100

# Single-GPU training comparison



- Compared to Tesla P100, Tesla V100 improvements come from:
  - Better hardware associated with reading, decompressing and pre-processing data.
  - 20/48 CPU cores, mixed precision and batch size optimisation were used to improve I/O speed
  - The number of parallel reads was set to autotune in order to mitigate bottleneck.

# Multi-GPU training/inference and performance

- Multi-GPU training on the standalone E2E boosted top jets benchmark using Horovod framework (<https://github.com/horovod/horovod>)
- Training performed on 2 Tesla V100 GPUs

Comparison of training time for different batch size configurations (2 GPUs)			
Batch size	64	512	1024
Train Time	11.8 min/epoch	7.5 min/epoch	7.4 min/epoch
ROC-AUC	0.981	0.979	0.976



Computation time improved by increasing batch size with small performance deterioration

Performance of classifiers for different layer combination (2 GPUs, batch size 64)			
Layer combi.	Track $p_T$	Track $p_T+d_0+d_z$	Track $p_T+d_0+d_z+ECAL+HCAL$
ROC-AUC	0.953	0.972	0.981

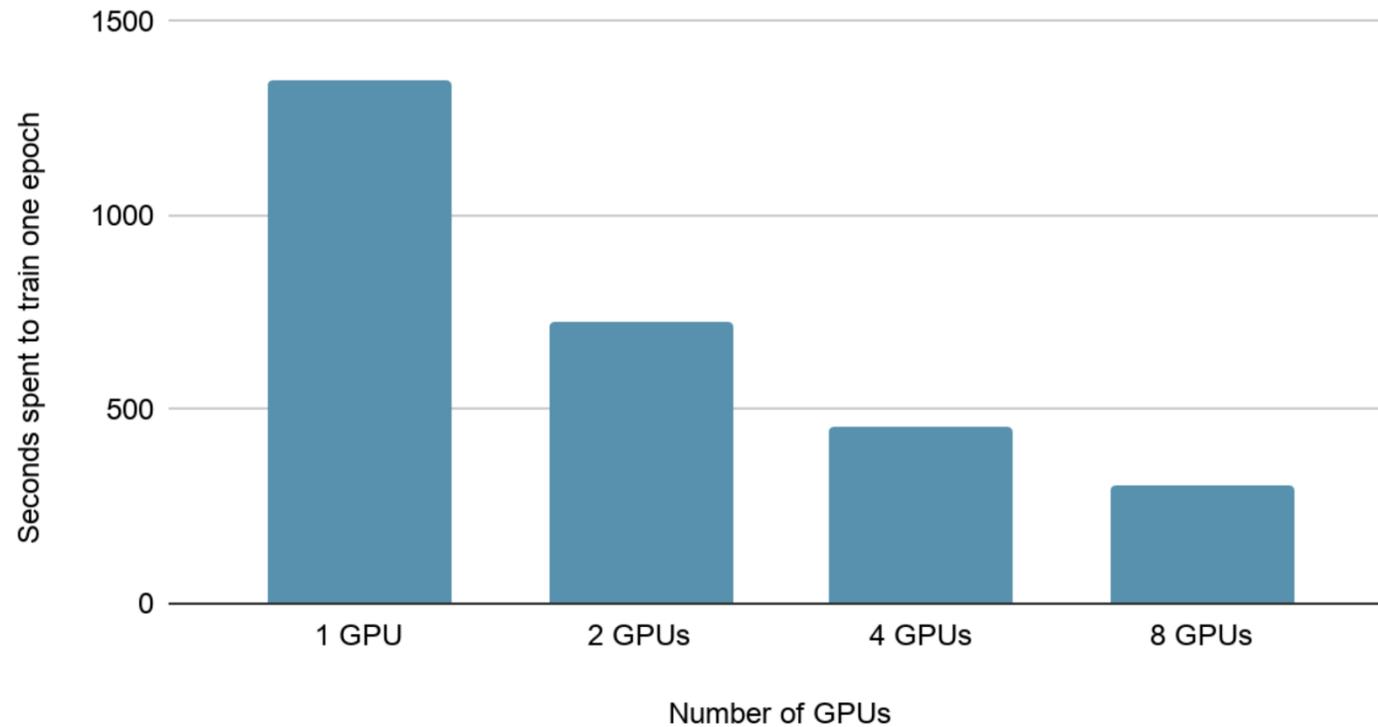


Performance in agreement with previous results

# Scaling multi-GPU training

- Scaling multi-GPU training on the standalone E2E boosted top jet benchmark performed on 8 Tesla V100 GPUs

Scalability Test



- Scaling with more GPUs improves the training time up to 5 times