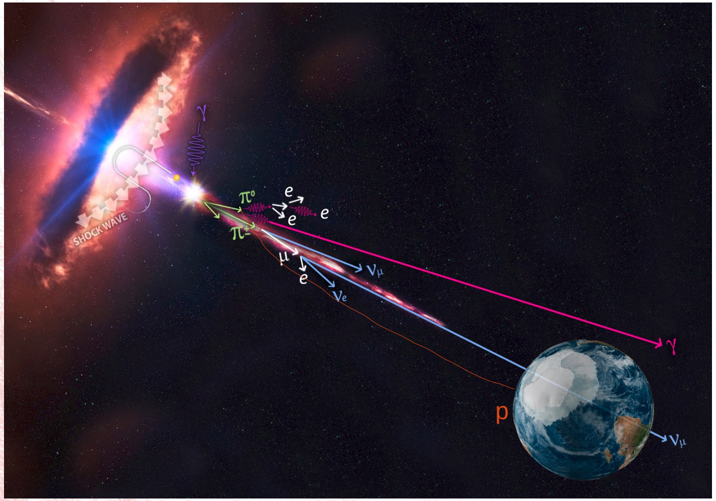
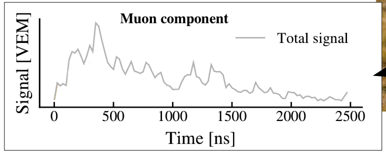
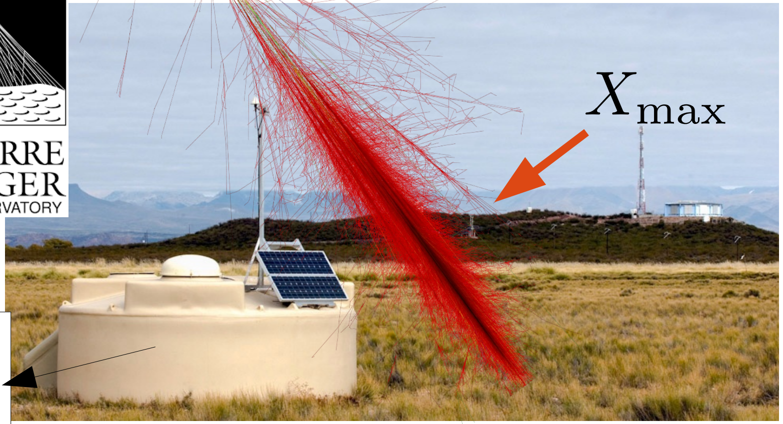
Pierre Auger Cosmic-Ray Observatory



Measure signal traces





[1] The Pierre Auger Collaboration, NIM-A, 798 (2015) 172-213

Ultra-high energy cosmic rays

- measure cosmic-ray-induced air showers (10^{17} to $>10^{20}$ eV)
- investigate nature & origin of UHECRs
  - unknown for more than 100 years

The Pierre Auger Observatory

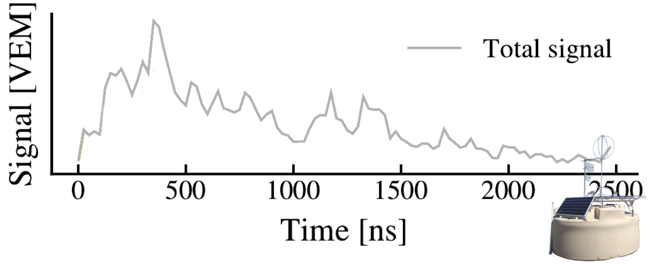
- world’s largest cosmic ray observatory
- Size: 3000 km² → 15x size of Amsterdam
- **hybrid detection** of air showers
  - 1,660 water-Cherenkov detectors 100% duty cycle
  - 27 fluorescence telescopes

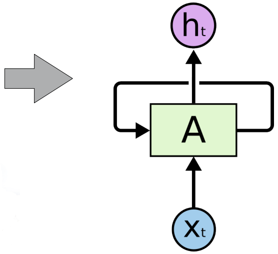
Mass Composition Studies using DNNs

- Xmax is estimator for primary mass
- Directly observed by fluorescence telescopes
- Reconstructed by DNN using detector traces
- Calibrated and crosschecked with telescopes

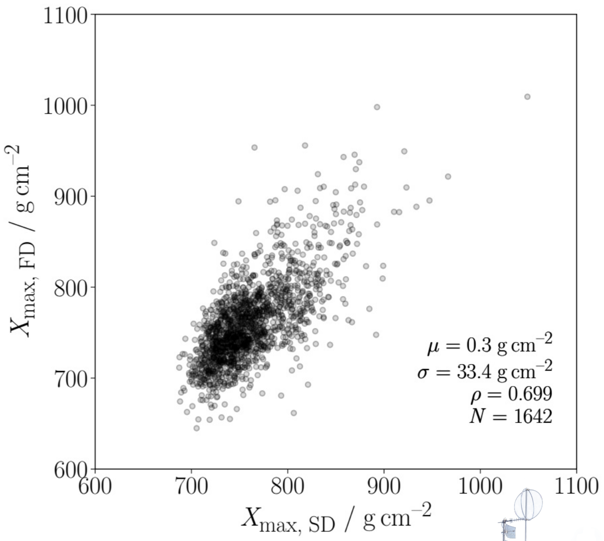
→ new insights in cosmic ray composition!

Would need to operate telescopes for >100 years to collect similar statistics

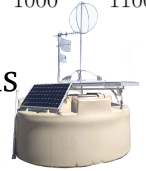




telescopes



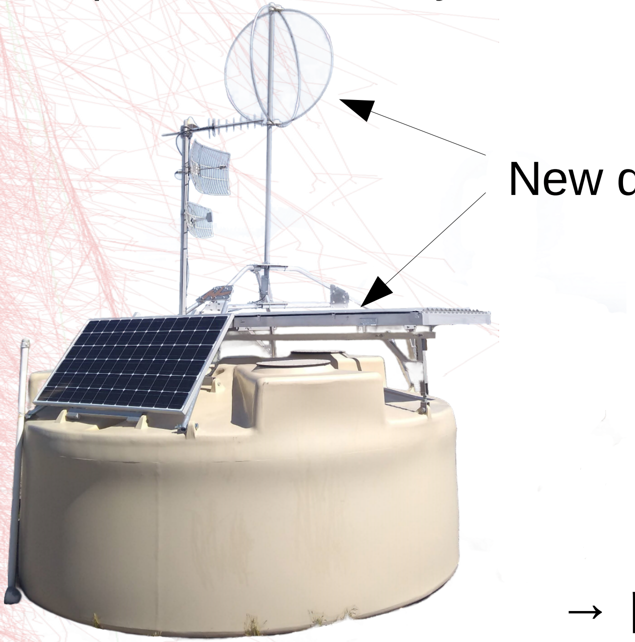
Detector stations

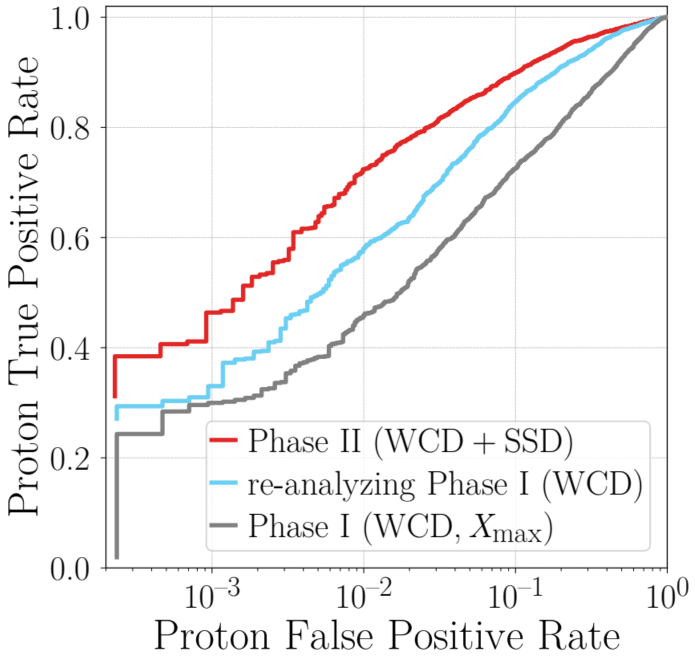


[2] A. Aab (Pierre Auger Collaboration) et al., JINST 16 P07019 (2021)
[3] A. Aab (Pierre Auger Collaboration) et al., JINST 16 P07016 (2021)

Machine Learning in the AugerPrime era

- Ongoing upgrade
  - add radio antenna & plastic scintillator
- ➔ improved sensitivity





→ promising potential to re-analyze previous data with improved sensitivity

[4] A. Aab (Pierre Auger Collaboration) et al., ArXiv:1604.03637
[5] N. Langner on behalf of the Pierre Auger Collaboration, PoS(ICRC2023)371

ROOT
Data Analysis Framework
https://root.cern

CERN

Fast Inference of Machine Learning Models with SOFIE

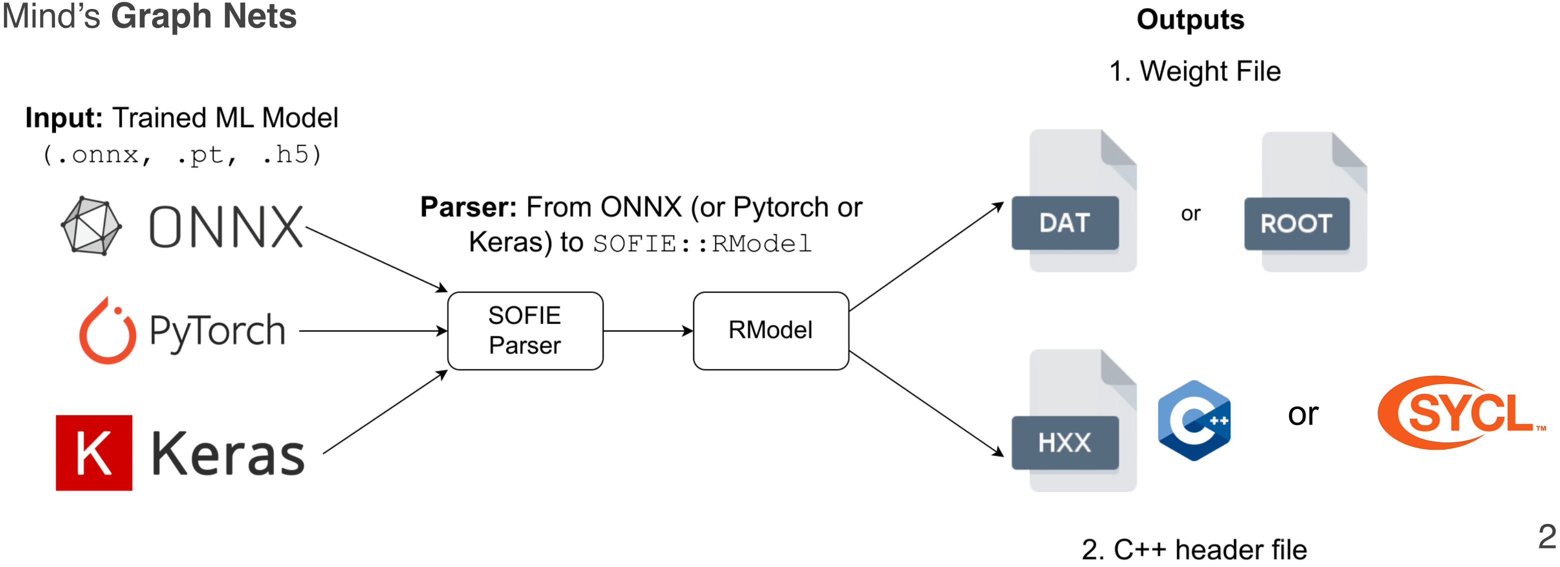
Lorenzo Moneta, Ioanna Panagou, Sanjiban Sengupta

EUROPEAN AI FOR FUNDAMENTAL PHYSICS CONFERENCE
EuCAIFCon 2024

Machine Learning Inference in ROOT

SOFIE : System for Optimised Fast Inference code Emit

- **Input**: trained ML model file
  - **ONNX**: Common standard for ML models
  - **Tensorflow/Keras** and **PyTorch** models (with reduced support than ONNX)
  - Since 6.32 support message passing **GNNs** from DeepMind's **Graph Nets**
- **Output**: **generated C++ code**
  - Easily **invokable directly** from C++ (plug-and-use)
  - **Minimal dependency** (on BLAS only)
  - Can be **compiled at run time** using ROOT Cling JIT and can be **used in Python.**



GPU Extension of SOFIE

- Extended SOFIE functionality to produce **GPU** code using **SYCL**

```
// generate SYCL code internally
model.GenerateGPU();
// write output header and data weight file
model.OutputGeneratedGPU();
```

- **Minimise overhead of data transfers** between host and device
- **Manage buffers efficiently, declaring them at the beginning**
- Use libraries for **GPU Offloading**: GPU BLAS from Intel one API and PortBLAS for other GPUs
- **Fuse operators** when possible in a single kernel
- **Replace conditional check** with relational functions

model.hxx

```
namespace TMVA_SOFIE_Linear_event{

struct Session {

Session(std::string filename ="") {
   if (filename.empty()) filename =
"Linear_event.dat";
   std::ifstream f;
   f.open(filename);
   // read weight data file
   ………………..
}
std::vector<float> infer(float*
tensor_input1){
```

with SYCL code

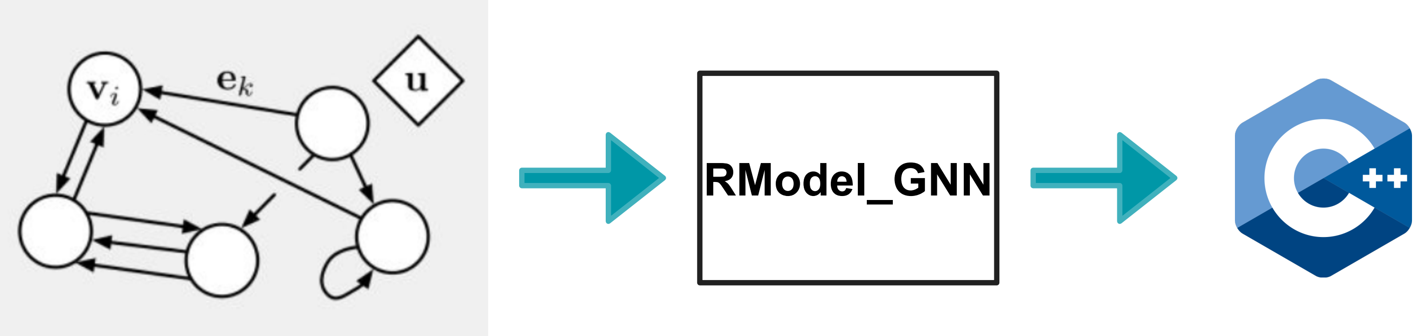
SYCL™

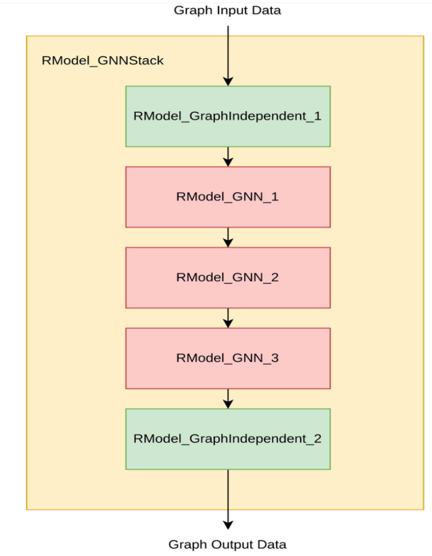
```
#include "Model.hxx"

// create session class

TMVA_SOFIE_Model::Session
ses("model_weights.dat");

//-- event loop

for (ievt = 0; ievt < N; ievt++) {

   // evaluate model: input is a C float array

   float * input = event[ievt].GetData();

   auto result = ses.infer(input);
```

Inference code needs to be linked against oneAPI MKL libraries and compiled using SYCL compiler

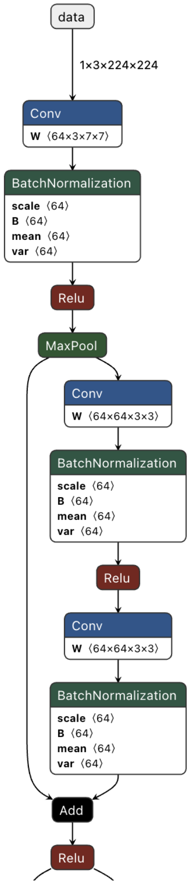
SOFIE GNN Support

- Since ROOT version 6.32 support inference of **GNNs**
  - parsing available for GNNs built from DeepMind's Graph Net library
  - supporting a LHCb model for full event interpretation (arXiv:2304.08610)





ONNX Supported Operators

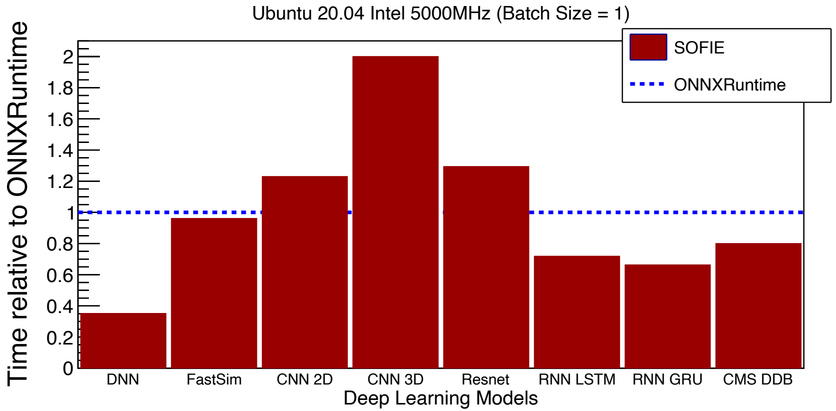


Operators implemented in ROOT	CPU	GPU
Perceptron: Gemm	✓	✓
Activations: Relu, Selu, Sigmoid, Softmax, Tanh, LeakyRelu, Swish	✓	✓
Convolution and **Deconvolution** (1D, 2D and 3D)	✓	✓
Pooling: MaxPool, AveragePool, GlobalAverage	✓	✓
Recurrent: RNN, GRU, LSTM	✓	✓
Layer Unary operators: Neg, Exp, Sqrt, Reciprocal, Identity	✓	✓
Layer Binary operators: Add, Sum, Mul, Div	✓	✓
Other Layer operators: Reshape, Flatten, Transpose, Squeeze, Unsqueeze, Slice, Concat, Reduce, Gather	✓	✓
BatchNormalization, LayerNormalization	✓	✓
Custom operator	✓	

- current CPU support available in **ROOT 6.30**
- GPU/SYCL is implemented in a ROOT PR

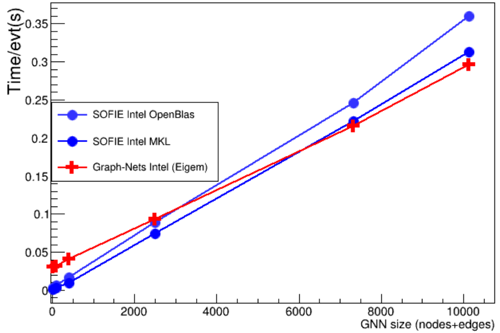
Benchmarking Time of Inference

CPU event performance of **SOFIE** vs **ONNXRuntime**



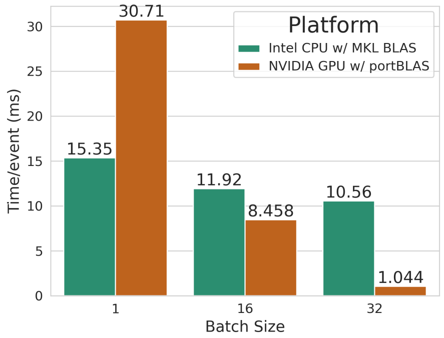
CPU time for **GNN inference**

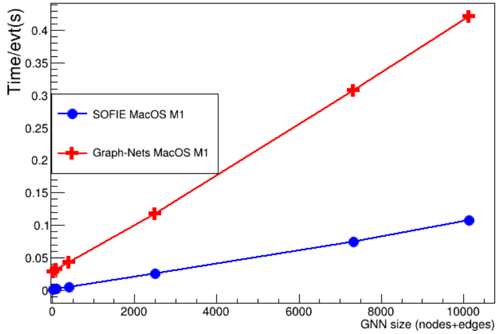
- varying GNN size (node + edges)



GPU (SYCL) vs CPU performance

- using a Resnet model with varying batch size





Summary

- **SOFIE**, fast and easy-to-use inference engine for Deep Learning models, is available in ROOT
  - Can be easily integrated with other ROOT tools (*RDataFrame*) for ML inference in end-user analysis
  - Supporting several **ONNX** operators and also **GNN**s
  - A prototype implementation for **GPU** using **SYCL** has been developed
    - plan to extend to **CUDA** and/or **ALPAKA** following some interest by experiments to deploy in their GPU-based trigger system
- Future developments according to user needs and received feedback
  - aim to support the latest production model of experiments (GNN and transformers)
  - models used for fast simulations (GAN and VAE)

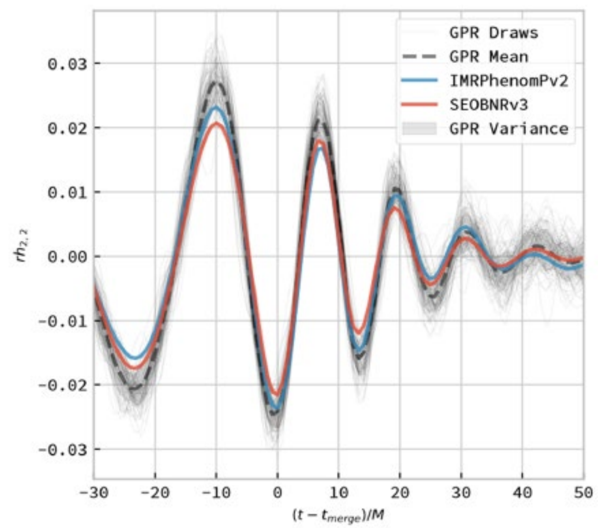
Useful Links

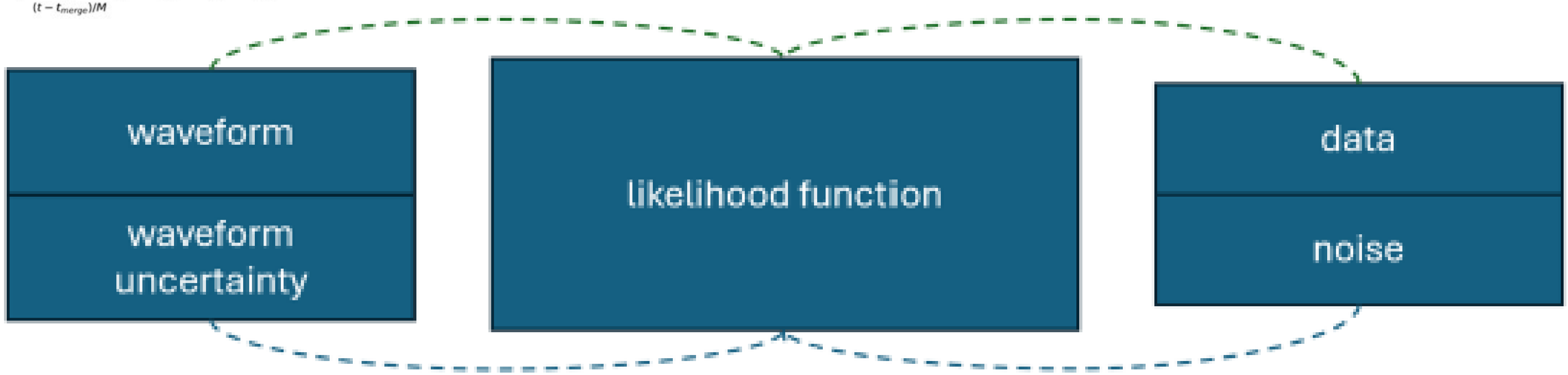
- **Examples and tutorials** are available in the tutorial/tmva directory
  - C++ (`TMVA_SOFIE_*.C`) and Python examples (`TMVA_SOFIE_*.py`)
- Link to **SOFIE code** in current ROOT master in GitHub
- Example **notebooks** on using SOFIE:
  - https://github.com/lmoneta/tmva-tutorial/tree/master/sofie
- Link to PR implementing SYCL code generation
- Link to benchmarks in *rootbench* repository

GAUSSIAN PROCESSES FOR MANAGING MODEL UNCERTAINTY IN GRAVITATIONAL WAVE ANALYSES

Daniel Williams

John Veitch


GPR Draws
GPR Mean
IMRPhenomPv2
SEOBNRv3
GPR Variance
0.03
0.02
0.01
0.00
-0.01
-0.02
-0.03
-30
-20
-10
0
10
20
30
40
50
$(t - t_{merge})/M$
$rh_{2,2}$


waveform
waveform uncertainty
likelihood function
data
noise

Full-event reconstruction using CNN-based models on calibrated waveforms for the Large-Sized Telescope prototype of the Cherenkov Telescope Array Observatory

Tjark Miener[*,1], B. Lacave[1], D. Nieto[2],
M. Heller[1], T. Montaruli[1], L. Burmistrov[1]
and I. Bezshyiko[3,+] for the CTA-LST Project

[*] tjark.miener@unige.ch
[1] *Université de Genève*
[2] *IPARCOS, Universidad Complutense de Madrid*
[3] *University of Zurich*
[+] *speaker*

ctao.org

https://ctlearn.readthedocs.io

DOI 10.5281/zenodo.4576196

UNIVERSITÉ DE GENÈVE
FACULTÉ DES SCIENCES

IPARCOS

Abstract

The next-generation ground-based gamma-ray observatory, the Cherenkov Telescope Array Observatory (CTAO), will consist of two arrays of tens of imaging atmospheric Cherenkov telescopes (IACTs) to be built in the Northern and Southern Hemispheres, aiming to improve the sensitivity of current-generation instruments by a factor of five to ten. Three different sizes of IACTs are proposed to cover an energy range from 20 GeV to more than 300 TeV. This contribution focuses on the analysis scheme of the Large-Sized Telescope (LST), which is in charge of the reconstruction of the lower energy gamma rays of tens of GeV. The Large-Sized Telescope prototype (LST-1) of CTAO is in the final stage of its commissioning phase collecting a significant amount of scientific data.

The working principle of IACTs consists of the observation of extended air showers (EASs) initiated by the interaction of very-high-energy (VHE) gamma rays and cosmic rays with the atmosphere. Cherenkov photons induced by a given EAS are recorded in fast-imaging cameras containing the spatial and temporal development of the EAS together with the calorimetric information. The properties of the originating VHE particle (type, energy and incoming direction) can be inferred from those recordings by reconstructing the full-event using machine learning techniques. We explore a novel full-event reconstruction technique based on deep convolutional neural networks (CNNs) applied on calibrated waveforms of the IACT camera pixels using CTLearn. CTLearn is a package that includes modules for loading and manipulating IACT data and for running deep learning models, using pixel-wise camera data as input.

Introduction

In this contribution, we show how deep convolutional neural networks (CNNs) can be utilized to analyze Monte Carlo (MC) simulations of gamma-ray events using *CTLearn*[a], a Deep Learning (DL) framework for IACT event reconstruction, and *DL1-Data-Handler*[b], a package designed for the data management of DL-based image and waveform analysis techniques for IACT data.

IACT data analysis

The IACT data analysis flow consists of several stages (see Fig.1). At the first stage of the low-level analysis, the raw data products are waveforms, i.e., signal intensities recorded by the photodetectors over a given time interval. The first step is to calibrate and extract the signal. The obtained data products are images containing the integrated charges and peak arrival times of the event in each camera pixel. Then, the images are cleaned to remove most of the Night Sky Background (NSB) noise in order to properly parameterize the Cherenkov shower signal in form of an ellipsoid, which reduces the information of the event to a small set of parameters. MC simulations are nowadays utilized for training Machine Learning (ML)-based algorithms, e.g. Random Forest (RF), with the set of extracted parameters to infer the properties of the primary particles. Once the particle type, arrival direction, and energy are reconstructed, the Instrument Response Functions (IRFs) are obtained from MC simulations. Within the CTLearn framework, CNN-based model are developed to reconstruct the event properties from low-level data (waveforms or images) to access as much information as possible.

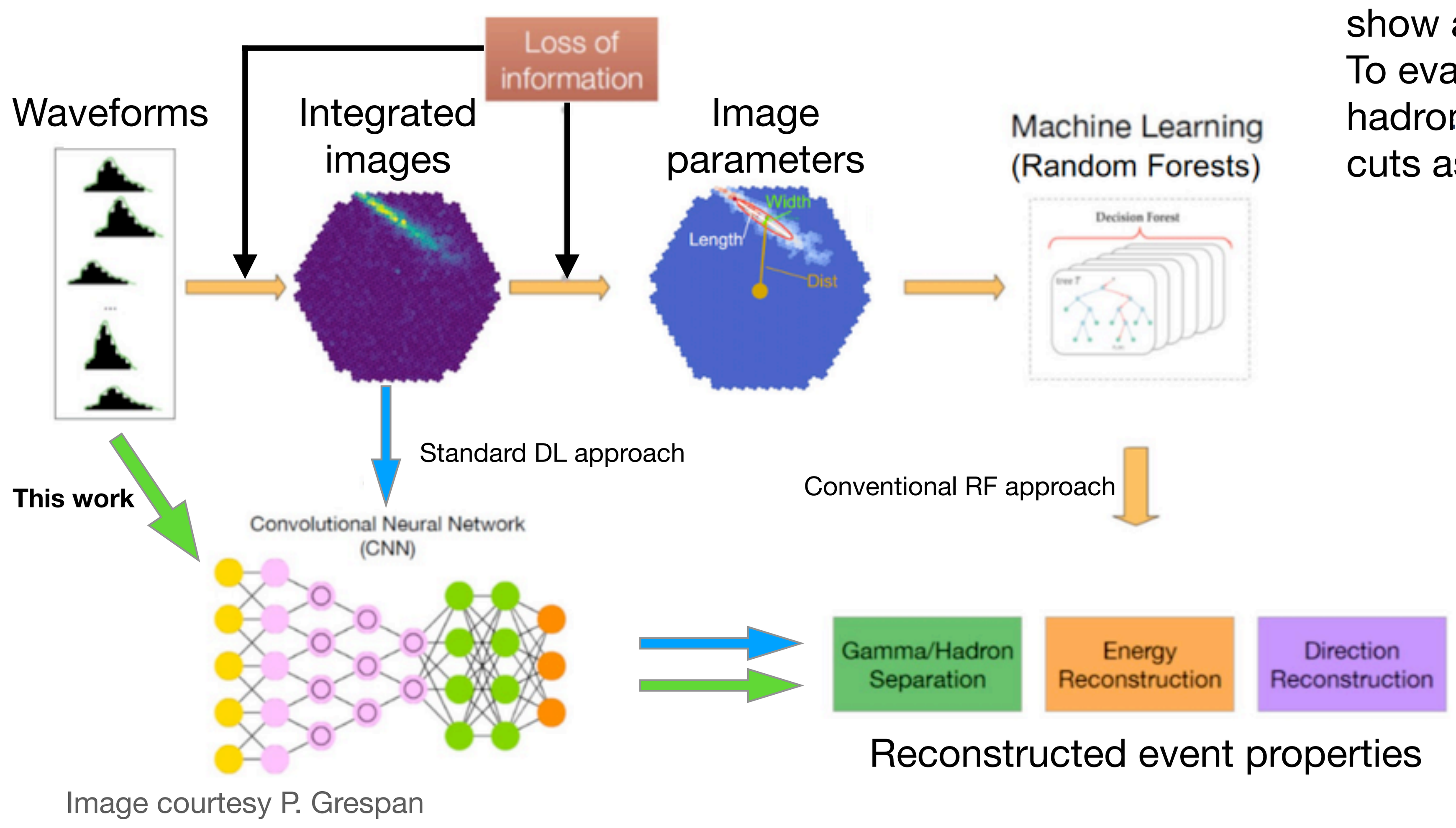
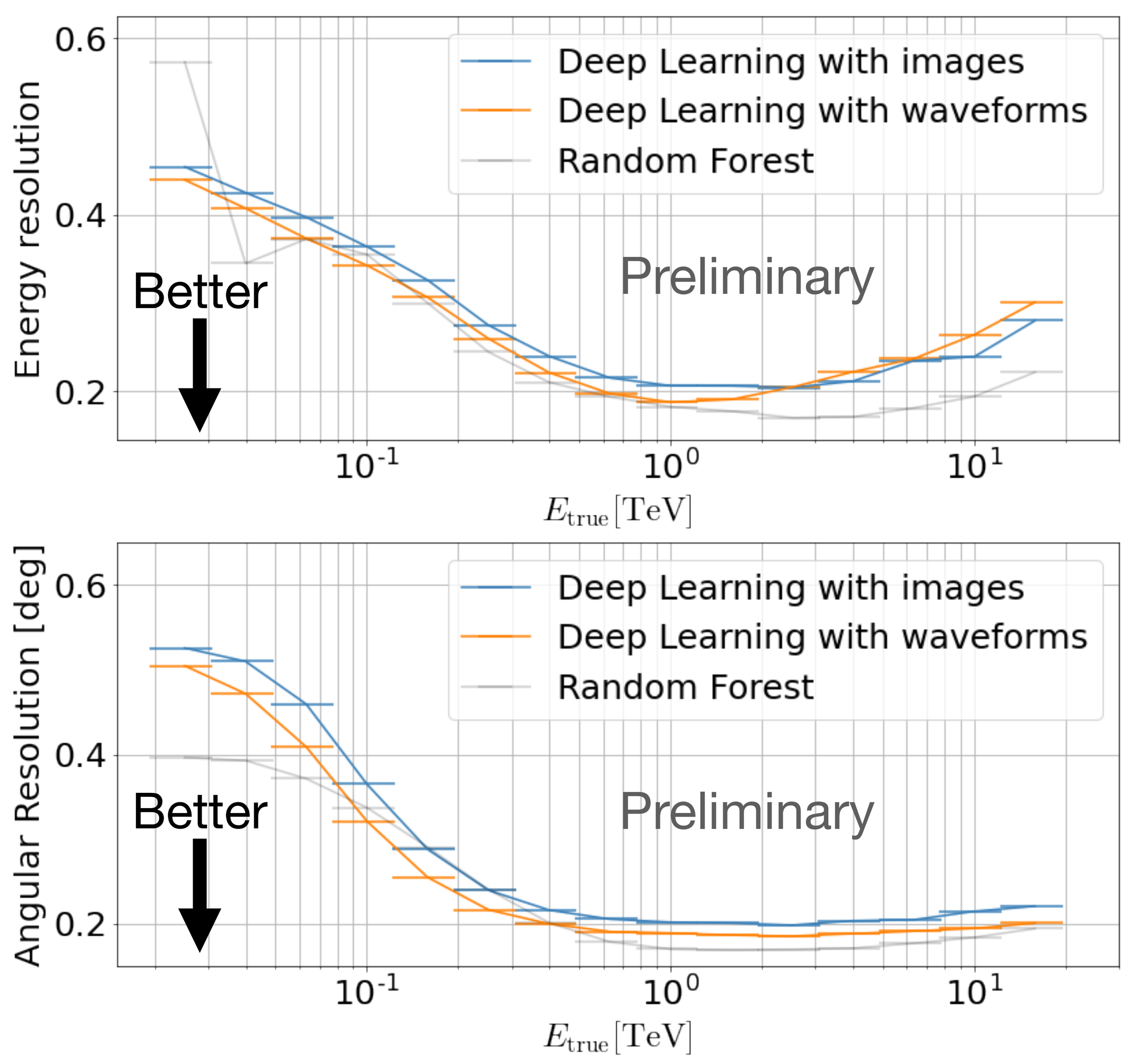


Fig. 1: Low-level data analysis scheme for IACTs (modified image from [1]).



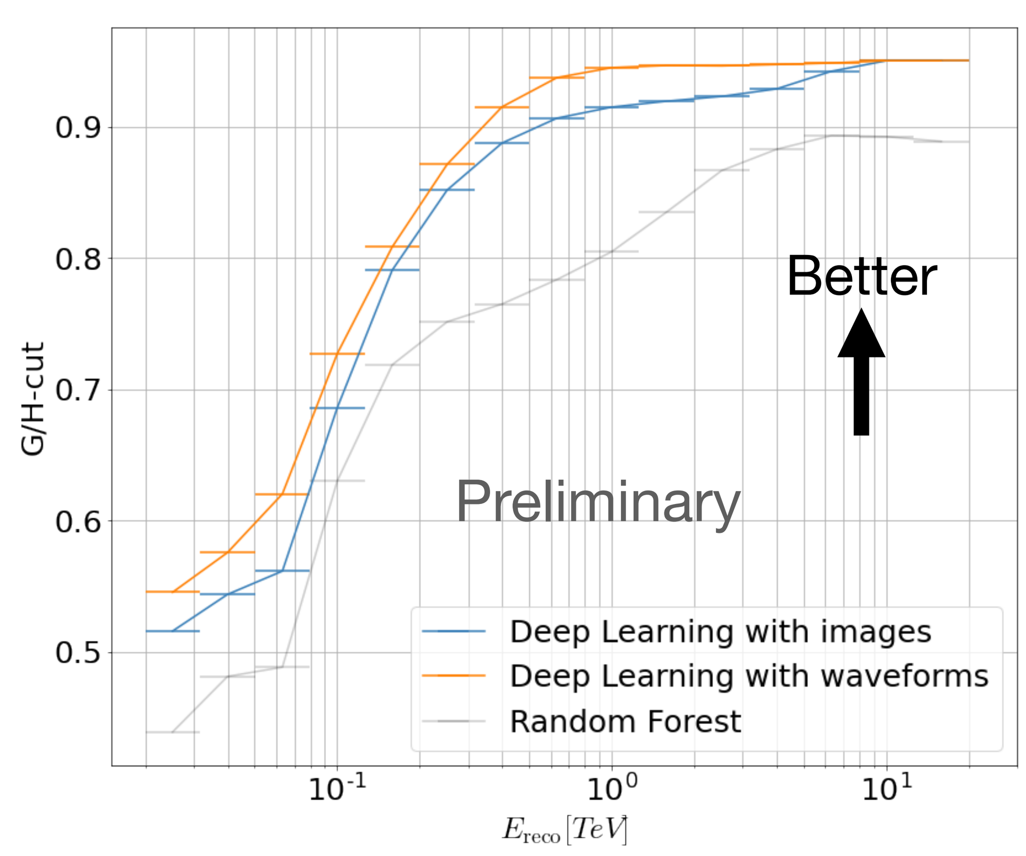


Fig. 2: Reconstruction performance of the two different DL approaches on MC simulations of the LST-1 of CTAO: Energy resolution (top left), angular resolution (bottom left) and gamma/hadron cuts for keeping 70% of the signal (right).

Validation on MC simulations

For this work, we selected *CTLearn*'s Thin-ResNet (TRN) model, which is a shallow residual neural network with 34 layers. We explore two DL approaches by feeding the TRN with calibrated waveforms or integrated images of the LST-1 of CTAO. The waveforms and images are cleaned by *ctapipe*, the low-level data processing pipeline software for CTAO, to suppress the major fraction of the NSB. It was demonstrated in [2] that CNNs trained with cleaned images rather than raw images show a stronger robustness, when applying them to observational data of the MAGIC telescopes. To evaluate the performance, IRFs like the energy and angular resolution curves and the gamma/hadron cuts for keeping 70% of the signal are computed (see Fig. 2), applying the same quality cuts as the conventional RF analysis.

Conclusion

CNN-based full-event reconstruction works for MC simulated data of the LST-1 prototype of CTAO using cleaned waveforms or images as input. For the first time, IACT full-event reconstruction has been achieved on the waveform-level with CNN-based models. Additionally, a clear performance gain is observed in comparison to the same network trained on IACT images. We plan to evaluate the full performance of the LST-1 of CTAO with the different CNN-based analyses under various observation conditions in the future and compare the results to the conventional analysis on observational data [3].

References

[1] P. Grespan et al., *Deep-learning-driven event reconstruction applied to simulated data from a single Large-Sized Telescope of CTA* [arXiv:2109.14262]
[2] T. Miener et al., *IACT event analysis with the MAGIC telescopes using deep convolutional neural networks with CTLearn* [arXiv:2112.01828]
[3] H. Abe et al., *Observations of the Crab Nebula and Pulsar with the Large-Sized Telescope Prototype of the Cherenkov Telescope Array* [arXiv:2306.12960]

Acknowledgements

This work was conducted in the context of the CTA-LST Project. We gratefully acknowledge financial support from the agencies and organizations listed here: https://www.ctao.org/for-scientists/library/acknowledgments/

[a] https://github.com/ctlearn-project/ctlearn
[b] https://github.com/cta-observatory/dl1-data-handler

lst1.iac.es