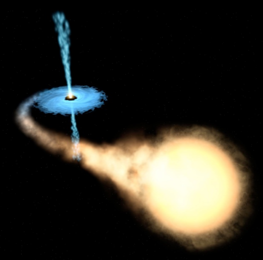
Deep-learning techniques in ground-based imaging gamma-ray observatories and the CTLearn package

Alexander Cerviño on behalf of the CTLearn Team

(rcervino@ucm.es)

- Introduction
- Imaging Atmospheric Cherenkov Technique
- Deep Learning applied to IACTs
  - CTLearn
  - AC-GAN's
  - AI Trigger
- Conclusions

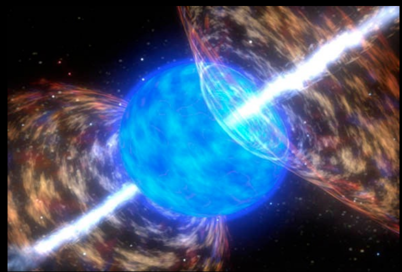
INTRODUCTION



Gamma-ray Binaries

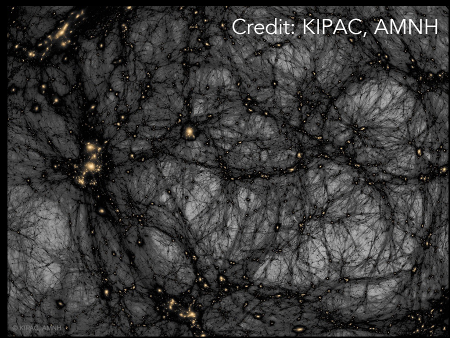
Supernova

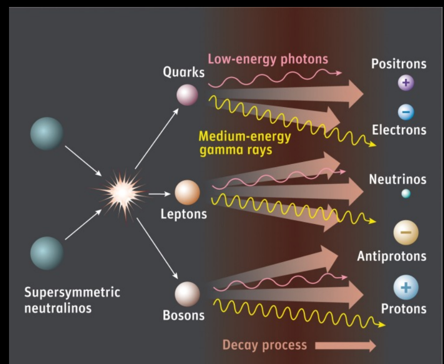
Active Galactic Nuclei



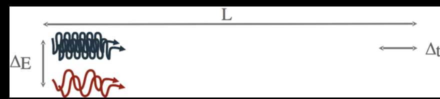
Gamma-ray Bursts

Dark matter searches



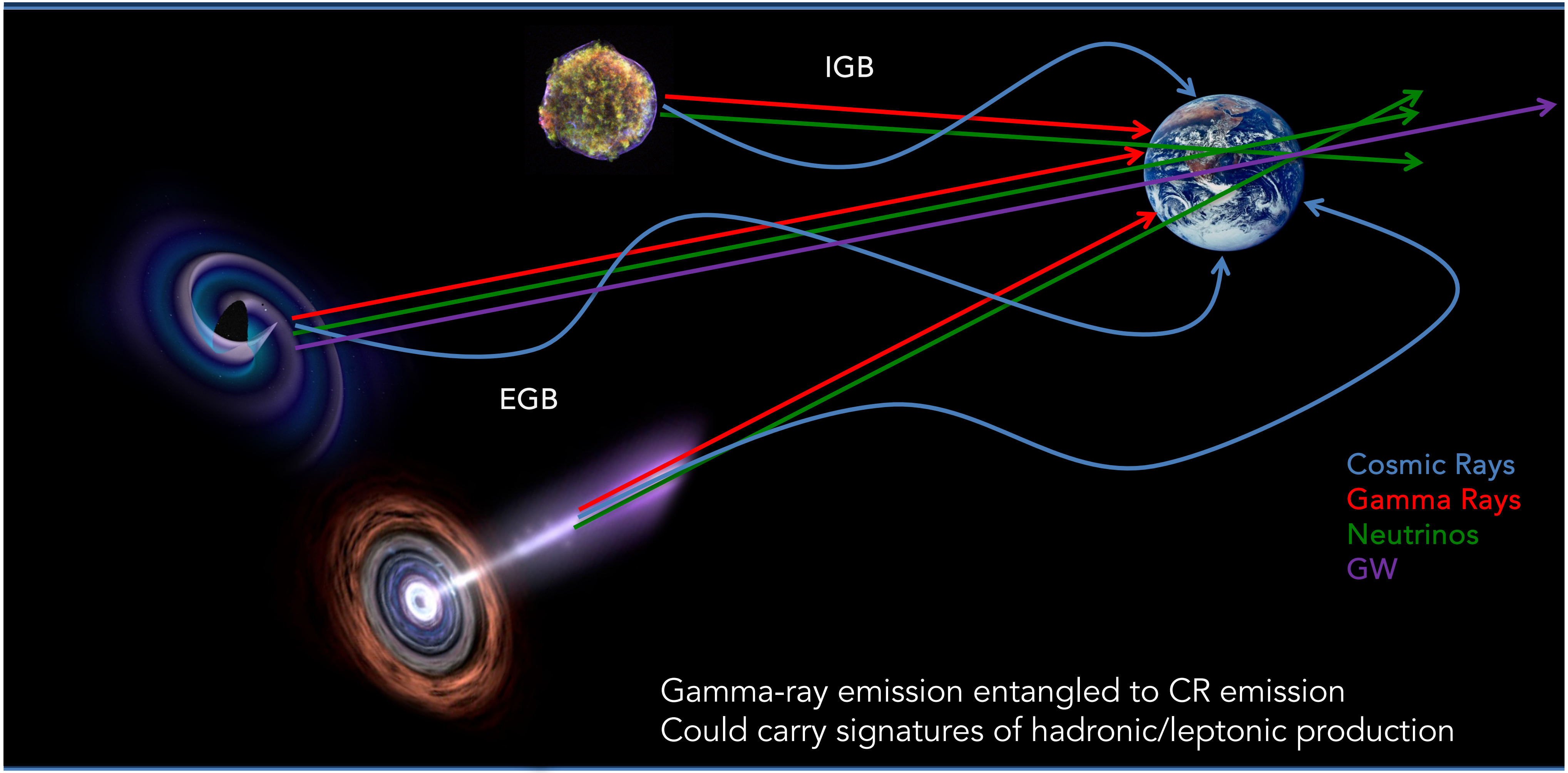


Lorentz invariance

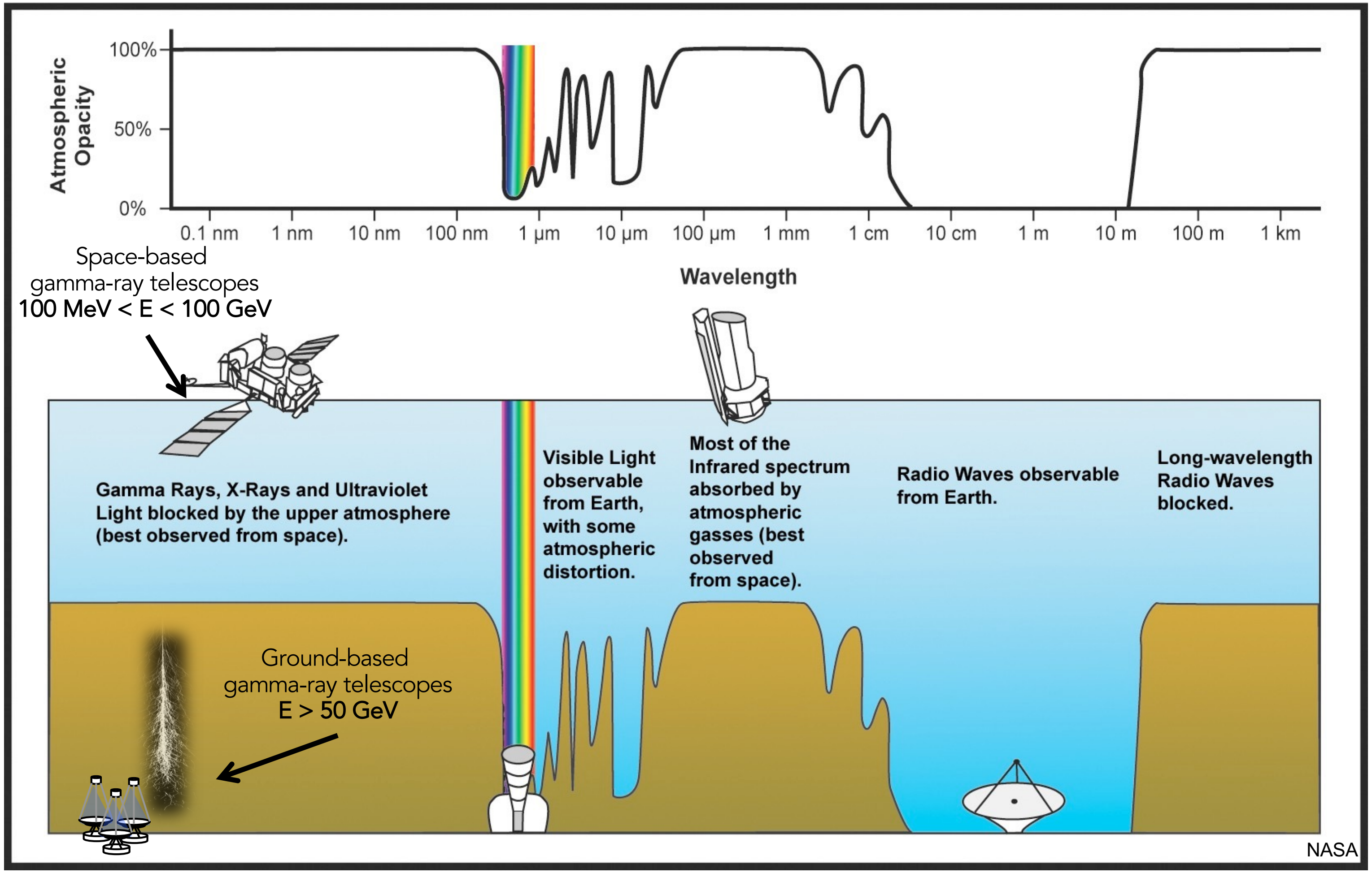


Credit: University of Warwick

Multimessenger astronomy

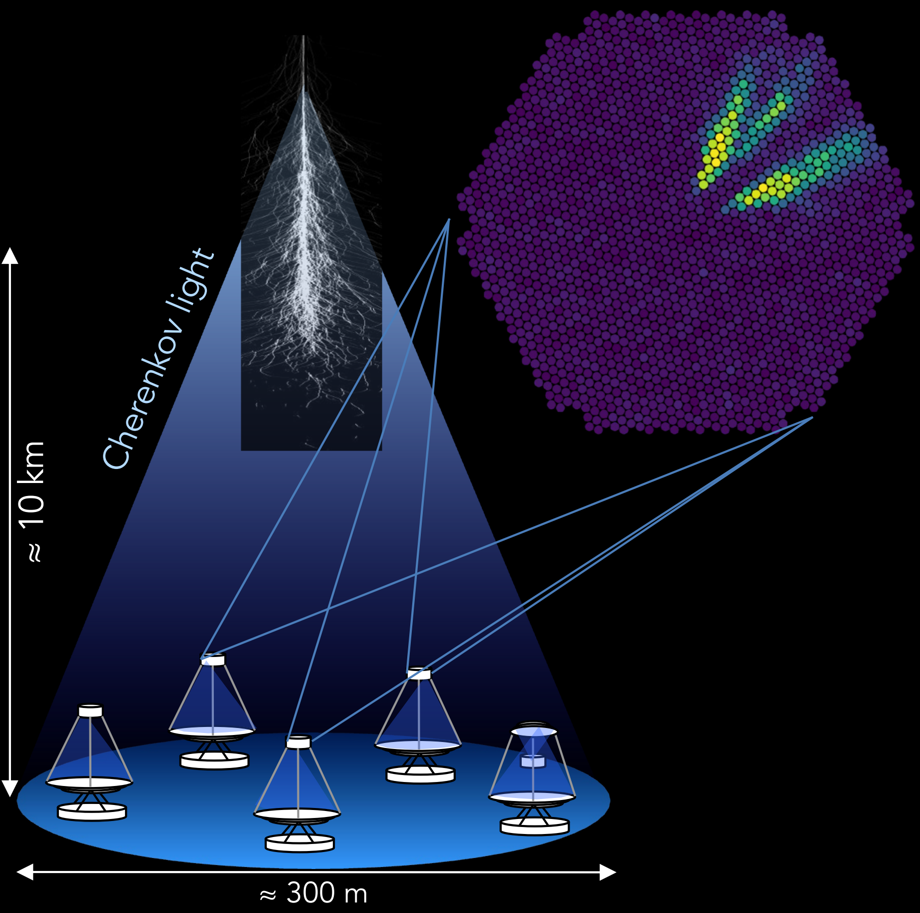


Gamma-ray detectors

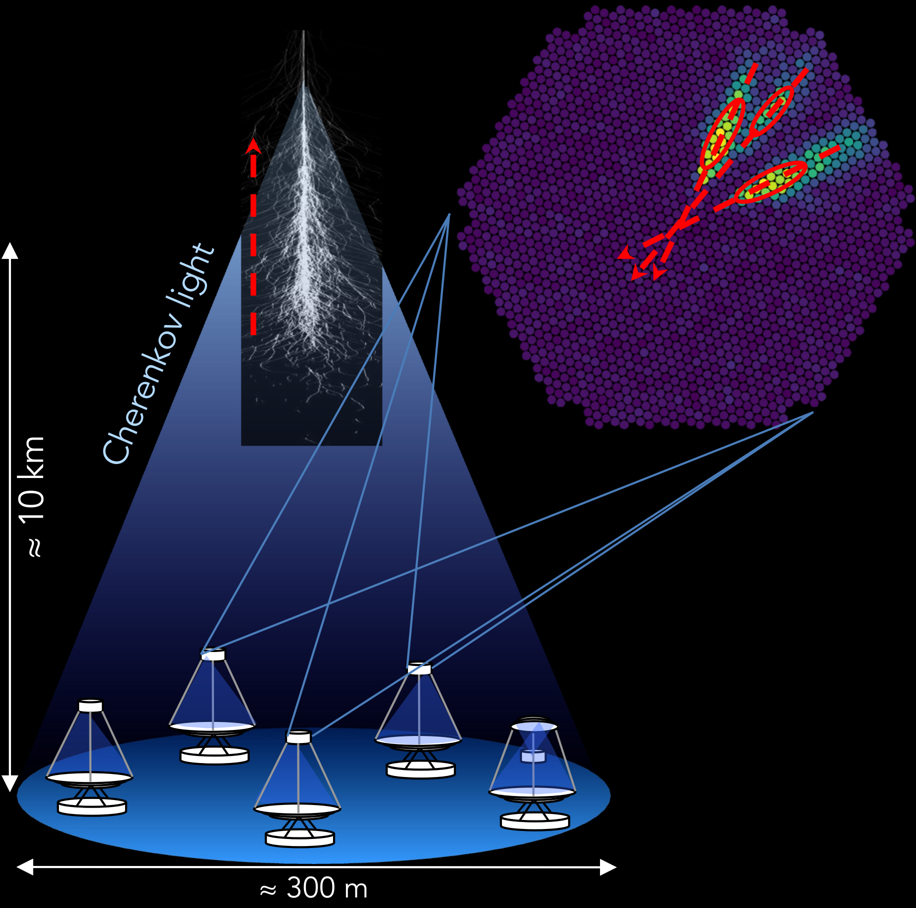


IMAGING ATMOSPHERIC CHERENKOV TECHNIQUE

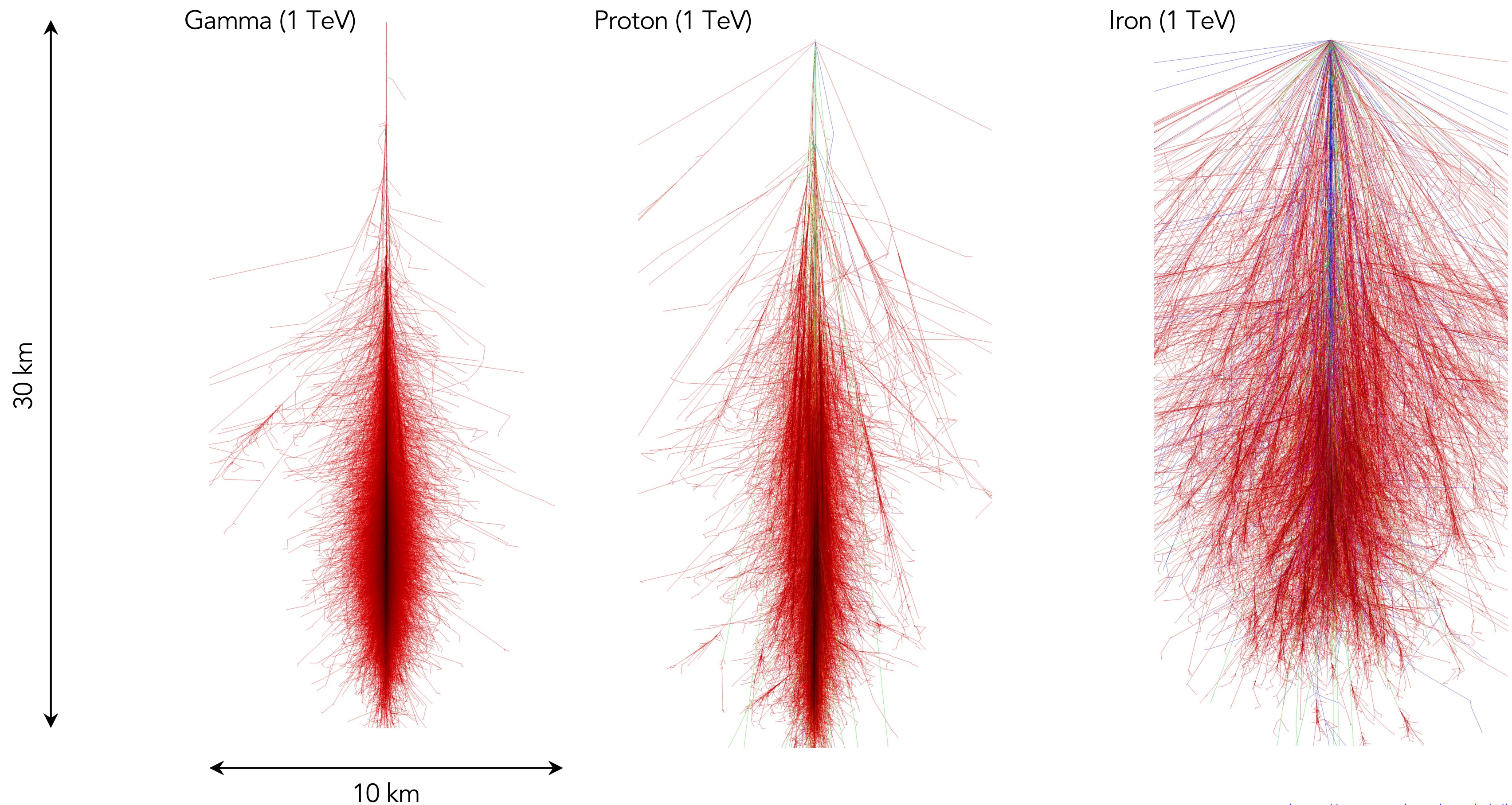
Imaging atmospheric Cherenkov technique



- Imaging Atmospheric Cherenkov Telescope (IACT)
- Detection of extended air showers using the atmosphere as a calorimeter
- Huge γ–ray collection area (~10^5 m^2)
- Large background from charged CR
  - Partly irreducible (e^-/e^+ , single-EM, with current methods)
- Energy window: tens GeV - tens TeV
- Event reconstruction from image:
  - Type of primary event
  - Primary energy estimation
  - Primary arrival direction

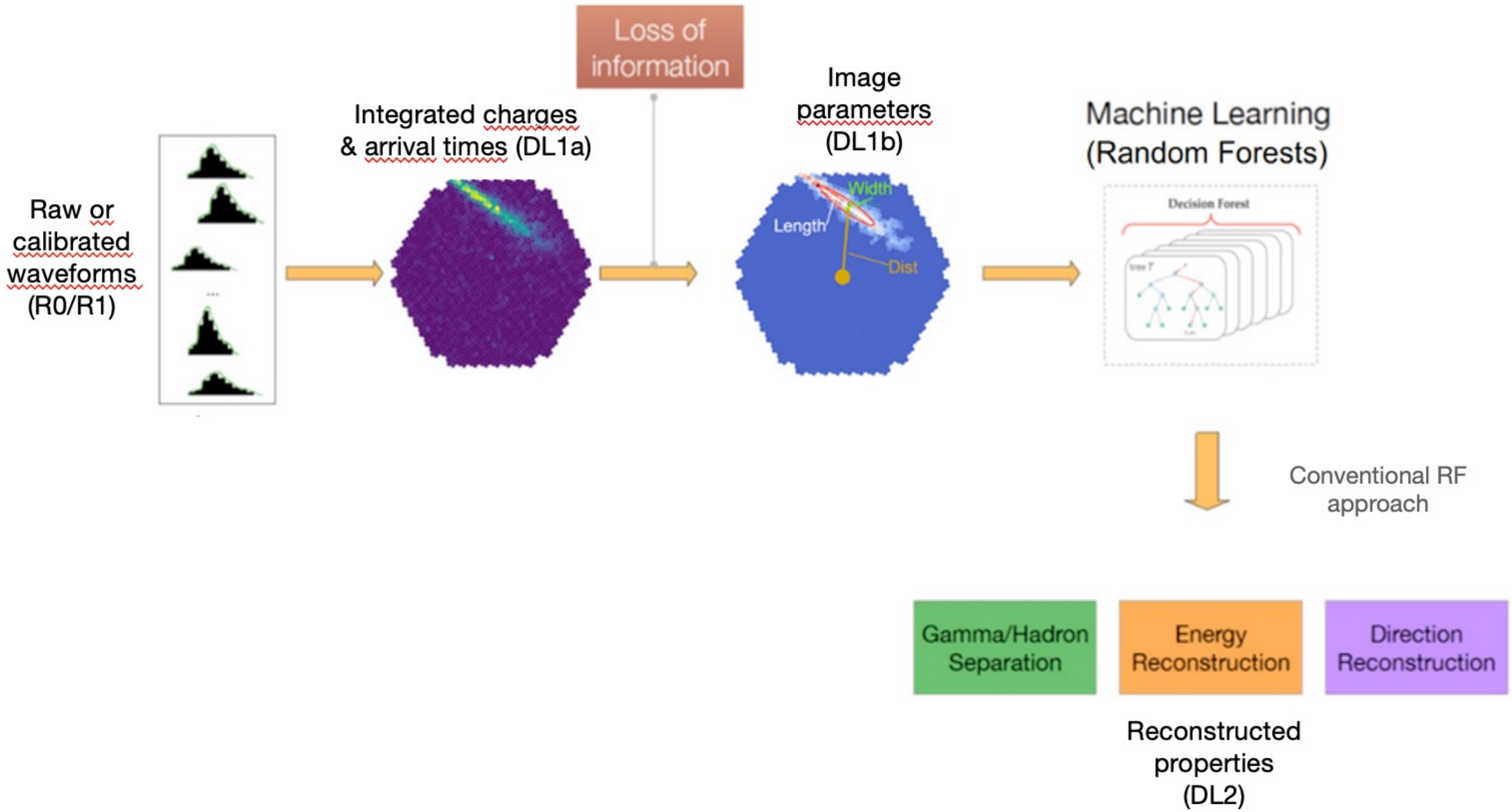


- Imaging Atmospheric Cherenkov Telescope (IACT)
- Detection of extended air showers using the atmosphere as a calorimeter
- Huge γ–ray collection area (~10^5 m^2)
- Large background from charged CR
  - Partly irreducible (e^-/e^+ , single-EM, with current methods)
- Energy window: tens GeV - tens TeV
- Event reconstruction from image:
  - Type of primary event
  - Primary energy estimation
  - Primary arrival direction



https://www-zeuthen.desy.de/~jknapp/fs/showerimages.html

Event reconstruction in IACTs – Current state



Next-generation IACT: The Cherenkov Telescope Array

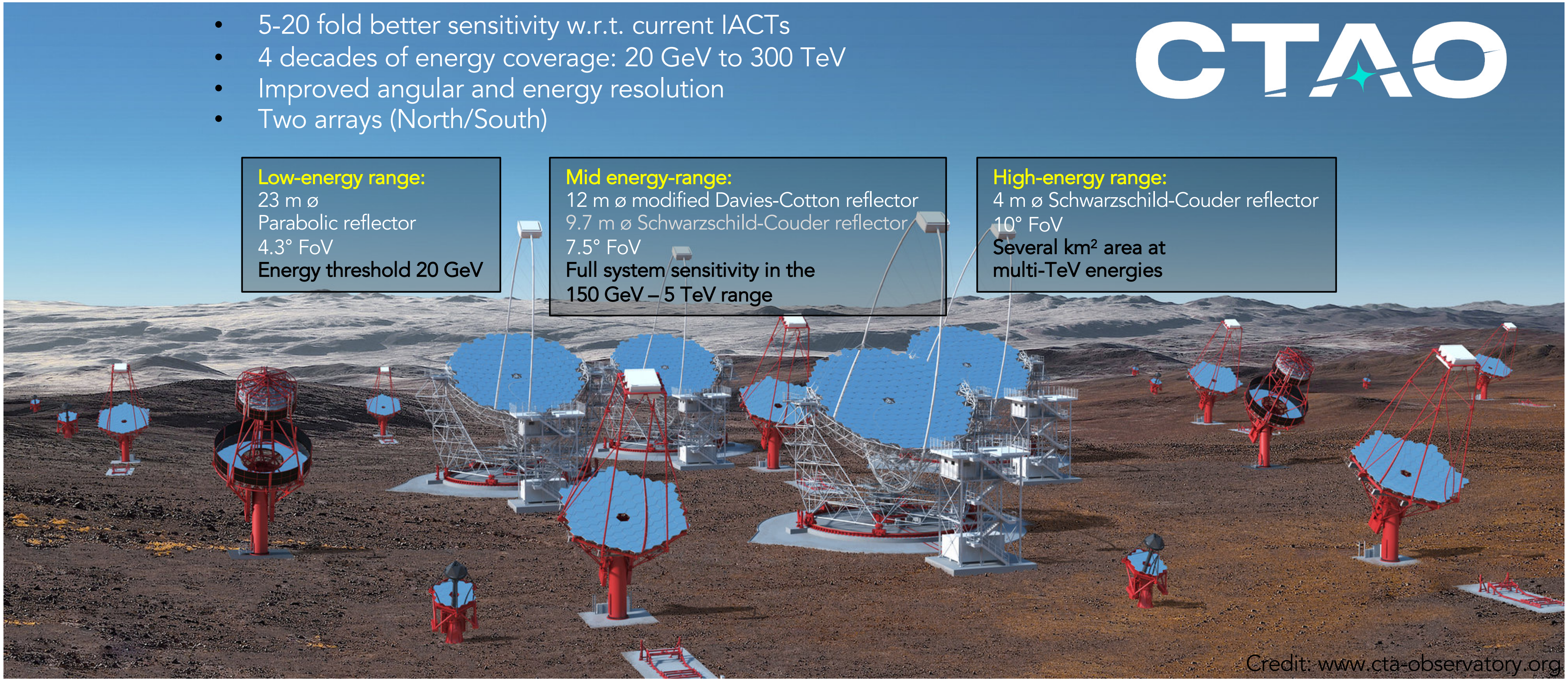
- 5-20 fold better sensitivity w.r.t. current IACTs
- 4 decades of energy coverage: 20 GeV to 300 TeV
- Improved angular and energy resolution
- Two arrays (North/South)

CTAO

Low-energy range:
23 m ø
Parabolic reflector
4.3° FoV
Energy threshold 20 GeV

Mid energy-range:
12 m ø modified Davies-Cotton reflector
9.7 m ø Schwarzschild-Couder reflector
7.5° FoV
Full system sensitivity in the 150 GeV – 5 TeV range

High-energy range:
4 m ø Schwarzschild-Couder reflector
10° FoV
Several km^2 area at multi-TeV energies



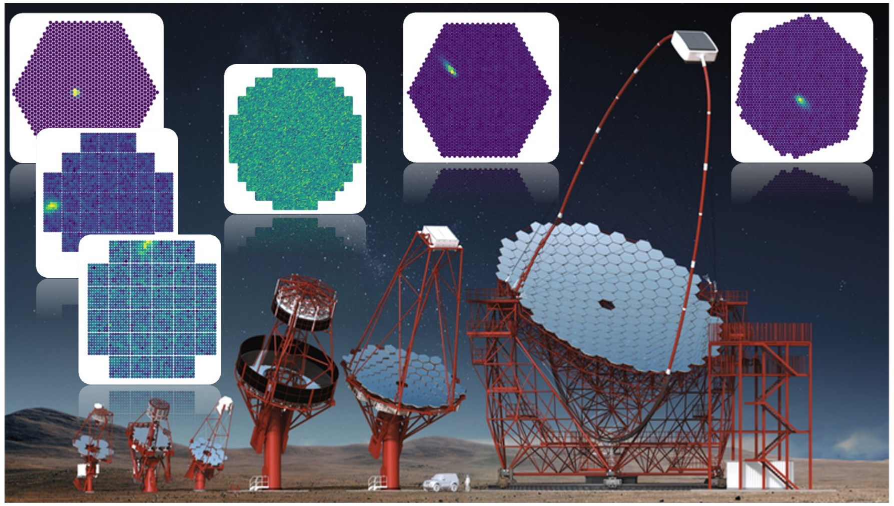
Credit: www.cta-observatory.org

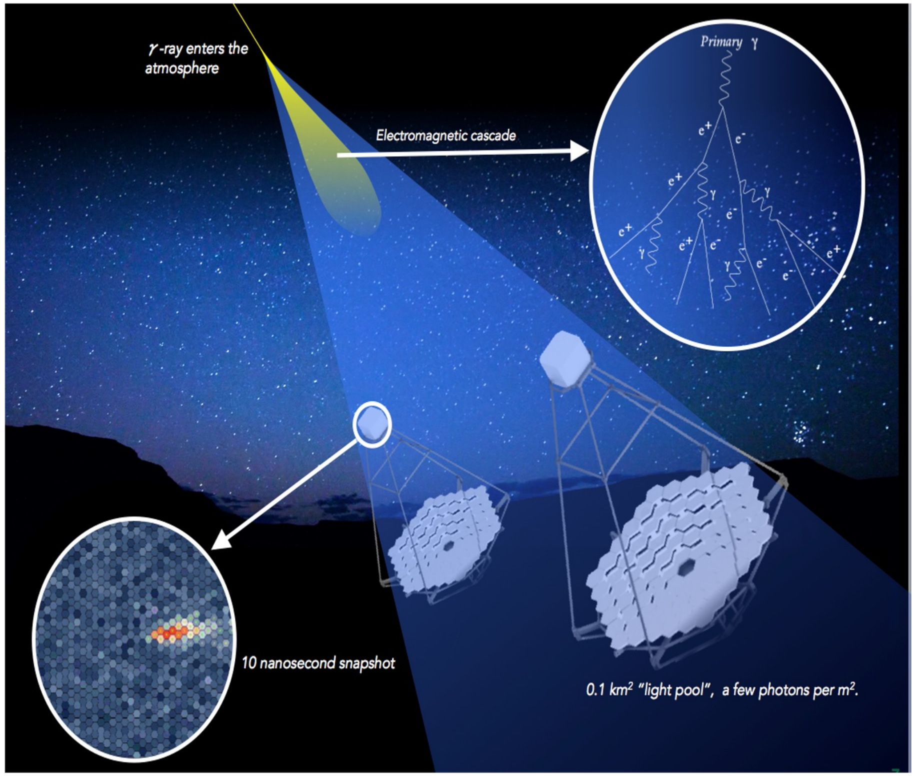
www.ctao.org

Science with CTA, arXiv:1709.07997

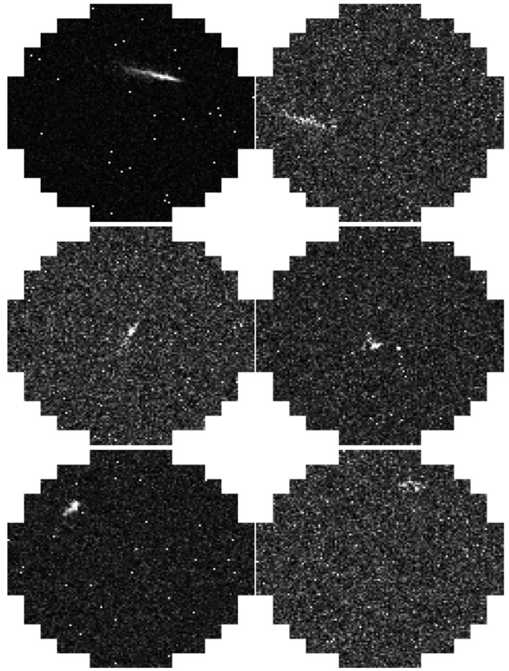
Challenges for machine learning from IACT data

- Stereoscopy
  - Stereoscopic view of the extended air showers
  - Compact “videos” rather than single snapshots
  - Events effectively recorded in 4D!
- Final metrics are far from trivial and entangled
- Heterogeneity of instruments

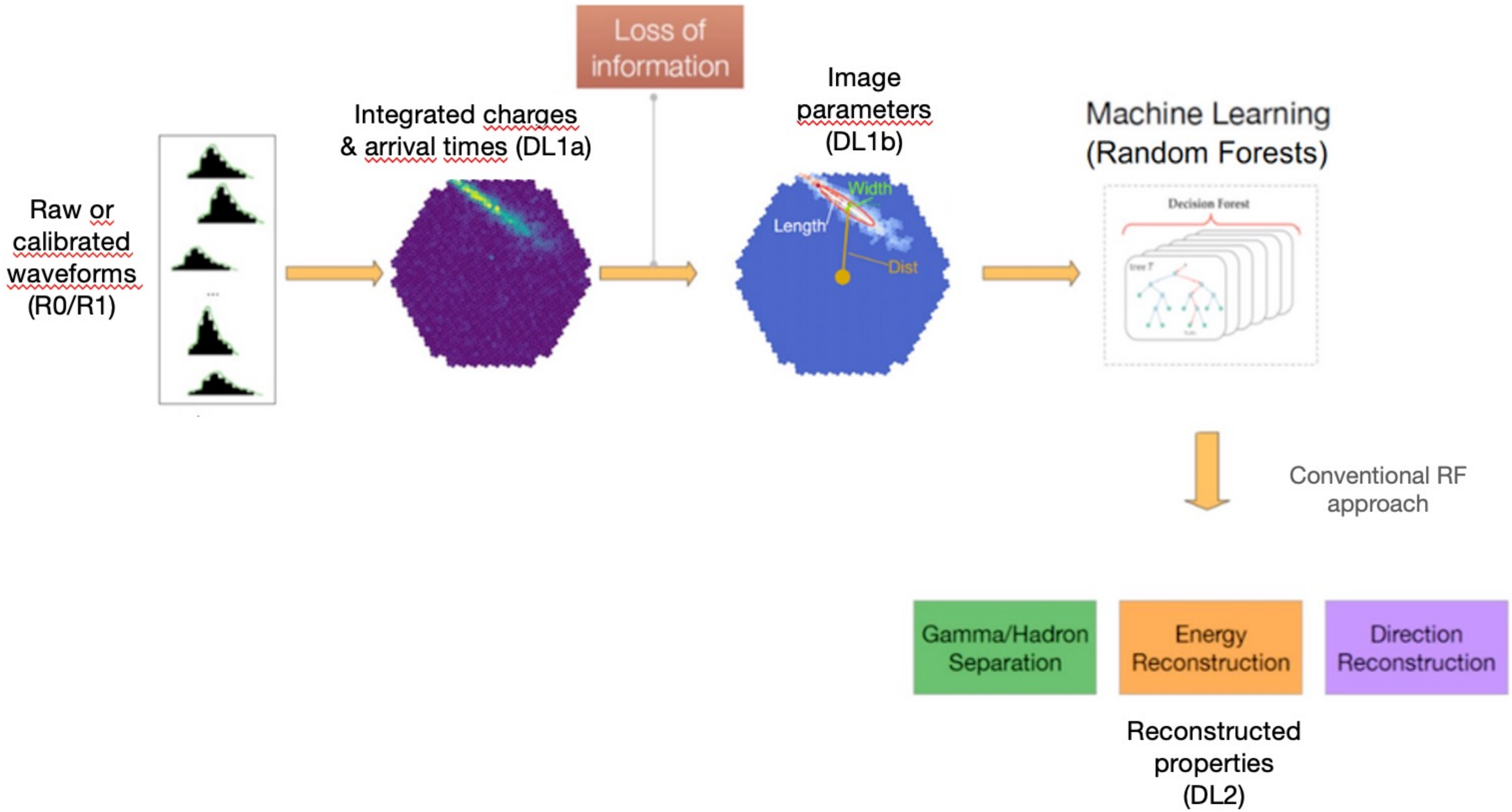

Camera images courtesy of T. Vuillaume



DEEP LEARNING APPLIED TO IACTs



Event reconstruction in IACTs – Current state



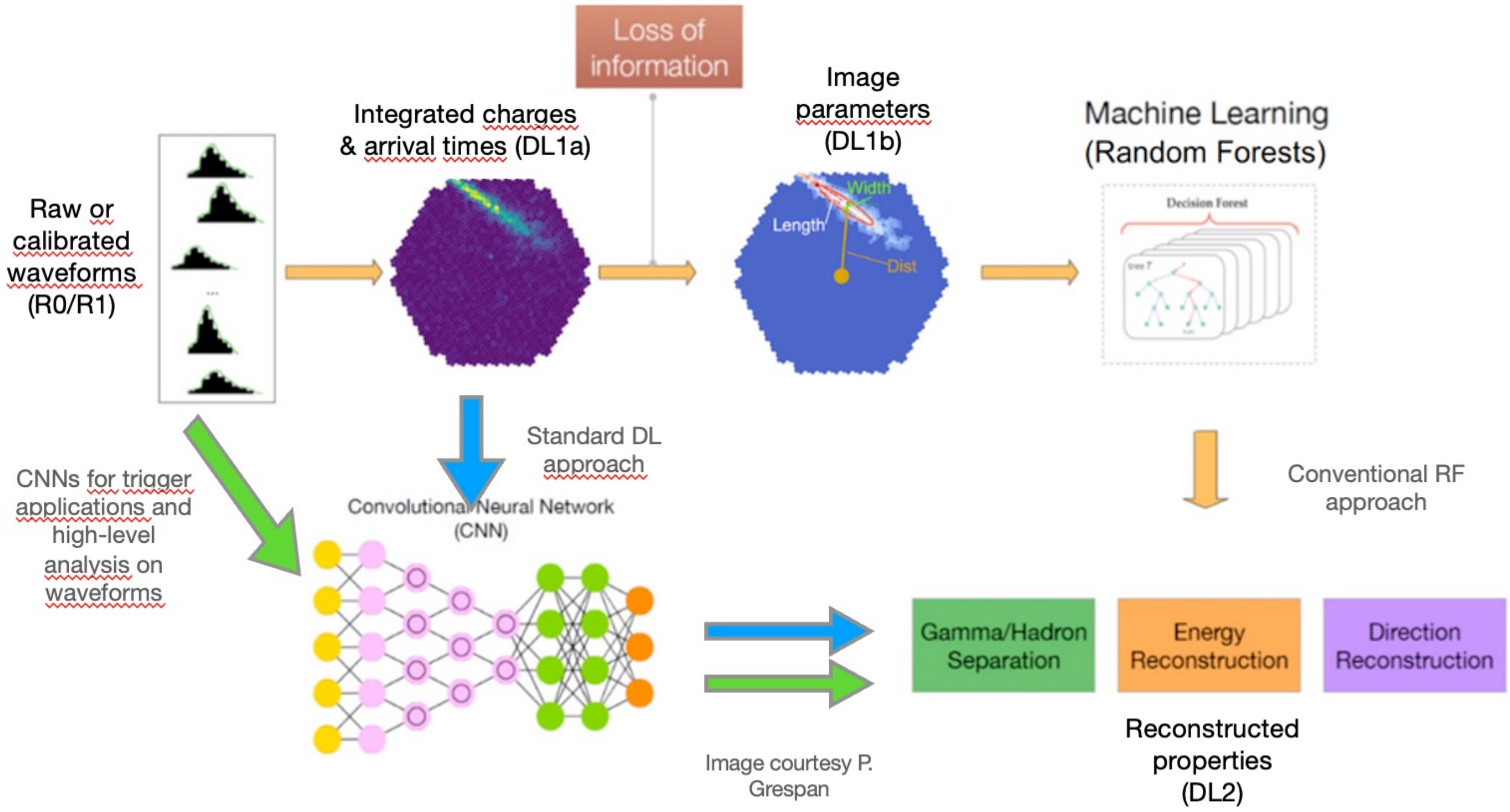


Image courtesy P. Grespan

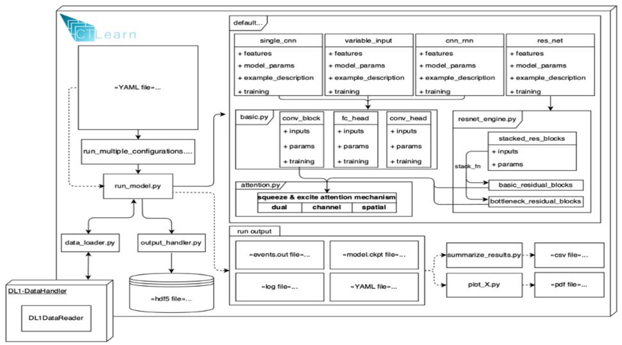
- High-level Python package for using deep learning for IACT event reconstruction
- Configuration-file-based workflow and installation with conda drive reproducible training and prediction
- Supports any TensorFlow model that obeys a generic signature
- Open source on GitHub:

https://github.com/ctlearn-project/ctlearn

DOI 10.5281/zenodo.11475531

(Latest release: **CTLearn v0.9.0**, 07/15/2024)

<u>Core developers</u>
Tjark Miener (U. Geneva), Daniel Nieto (IPARCOS-UCM), Bastien Lacave (U. Geneva) , Alexander Cerviño (IPARCOS-UCM), Ari Brill, Qi Feng (Columbia)
Bryan Kim (UCLA, now at Stanford)



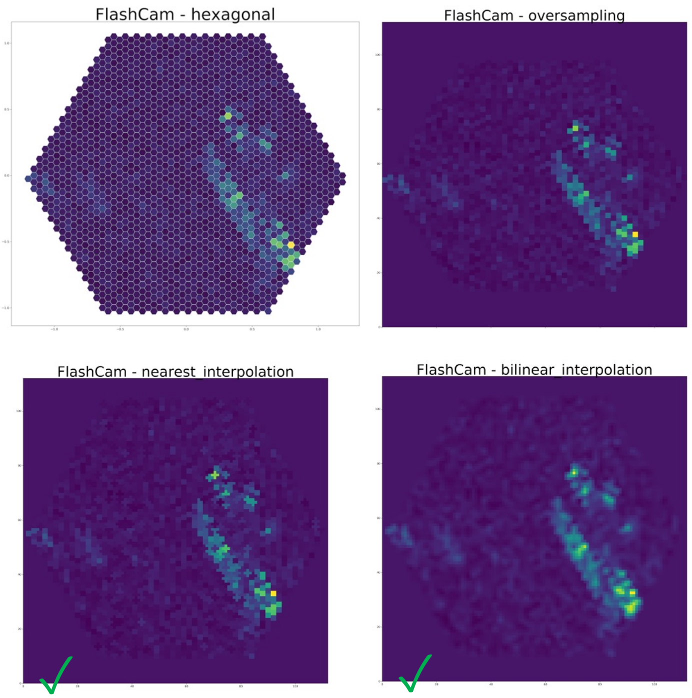
Tackling the hexagonal-pixel challenge

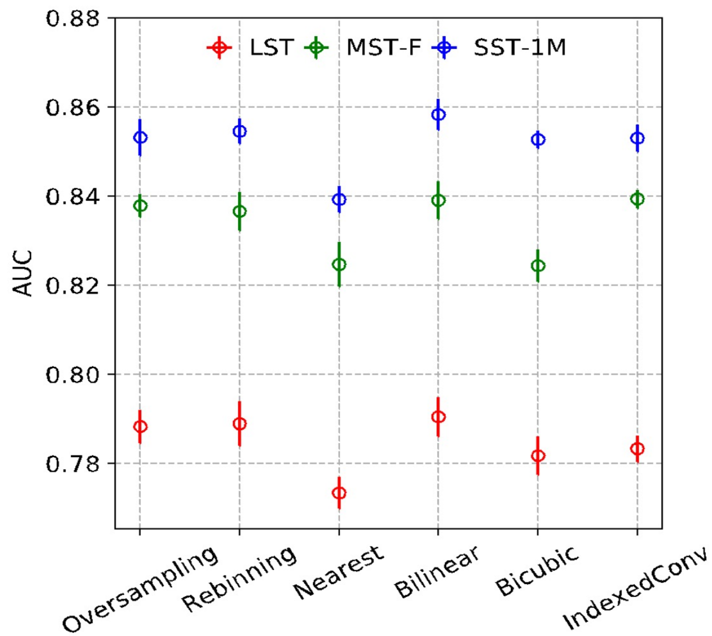
A. Brill, B. Kim, Q. Feng
D. Nieto, T. Miener,
et al.

CTLearn

https://github.com/ctlearn-project/

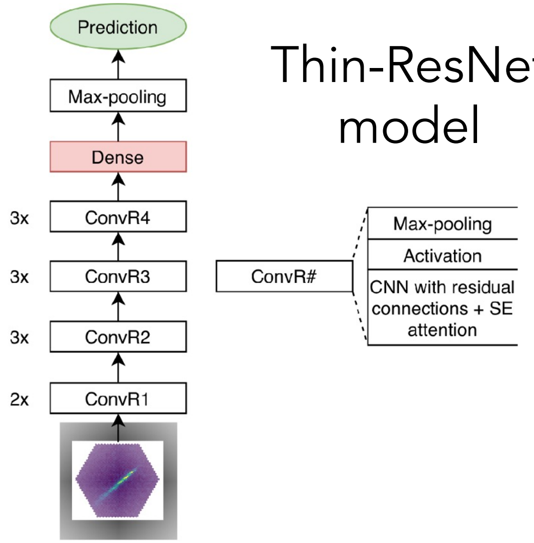
- Image mapping (preprocessing)



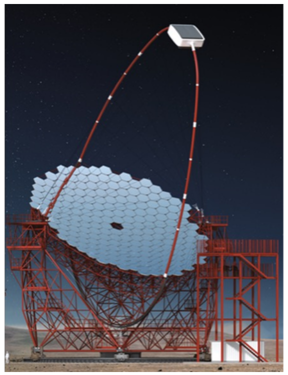


D. Nieto et al. PoS(ICRC2019)753

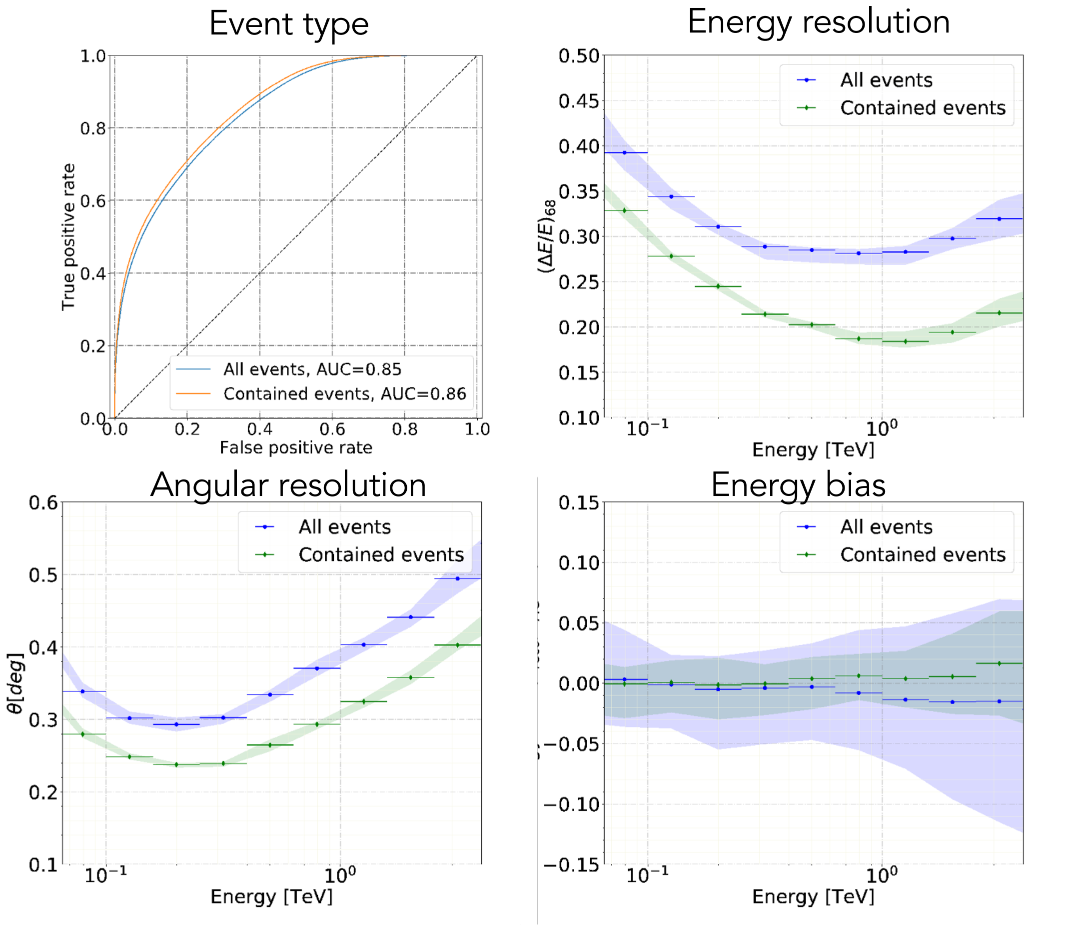
CTLearn: single-telescope full-event reconstruction



Thin-ResNet model

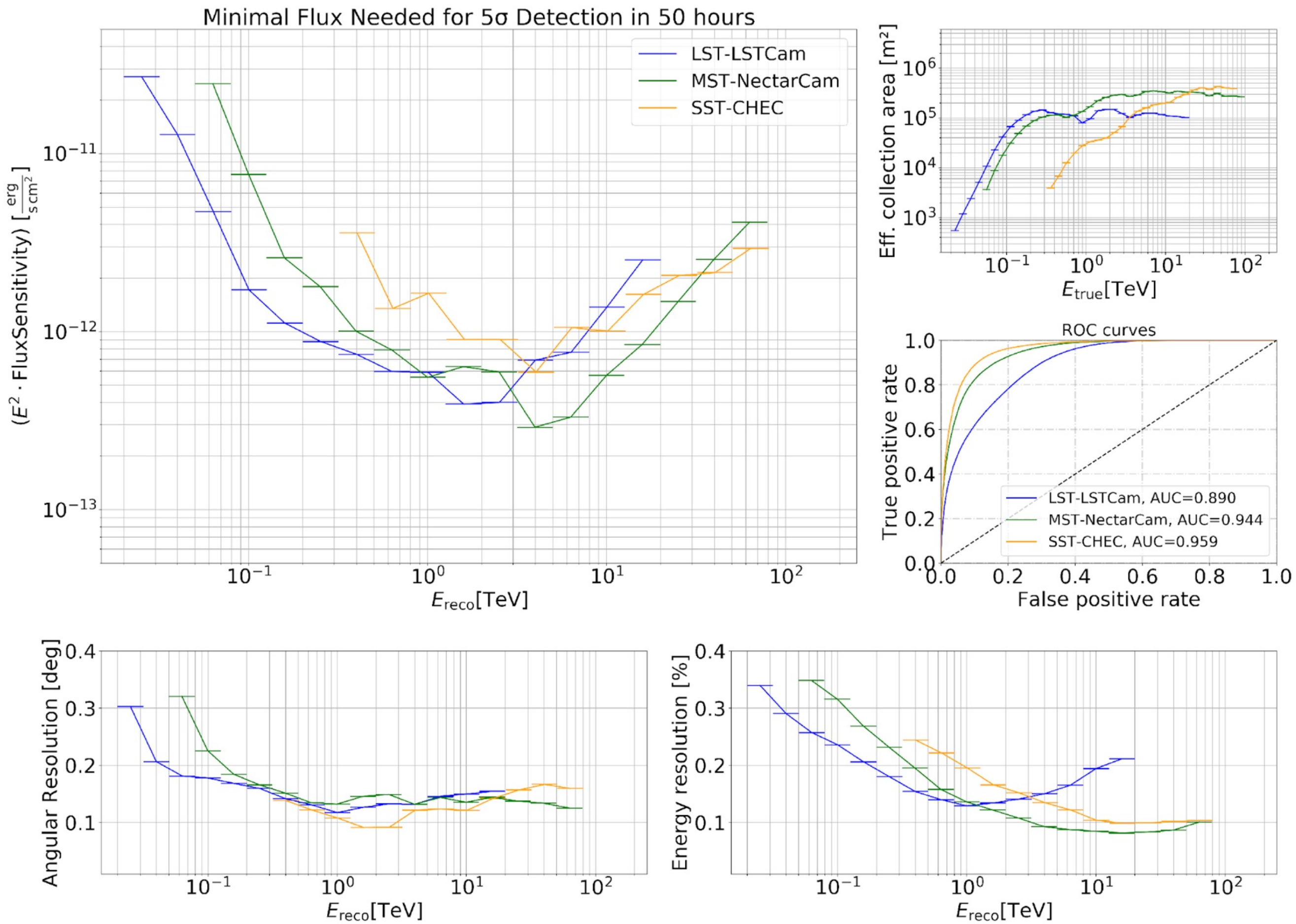


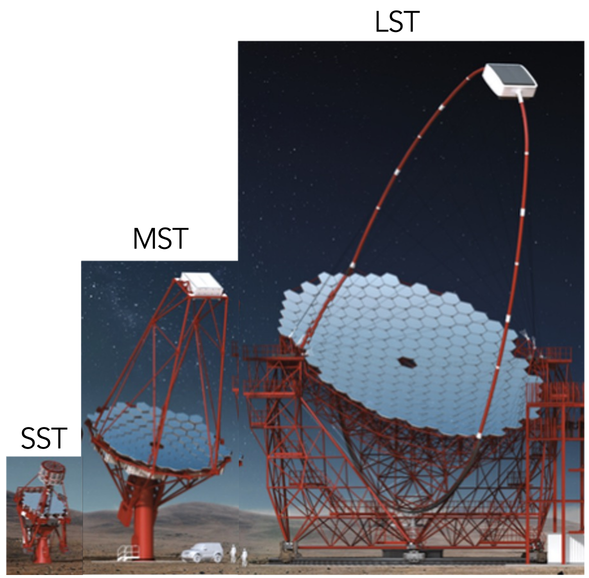
Full-event reconstruction for single-telescope data achieved!



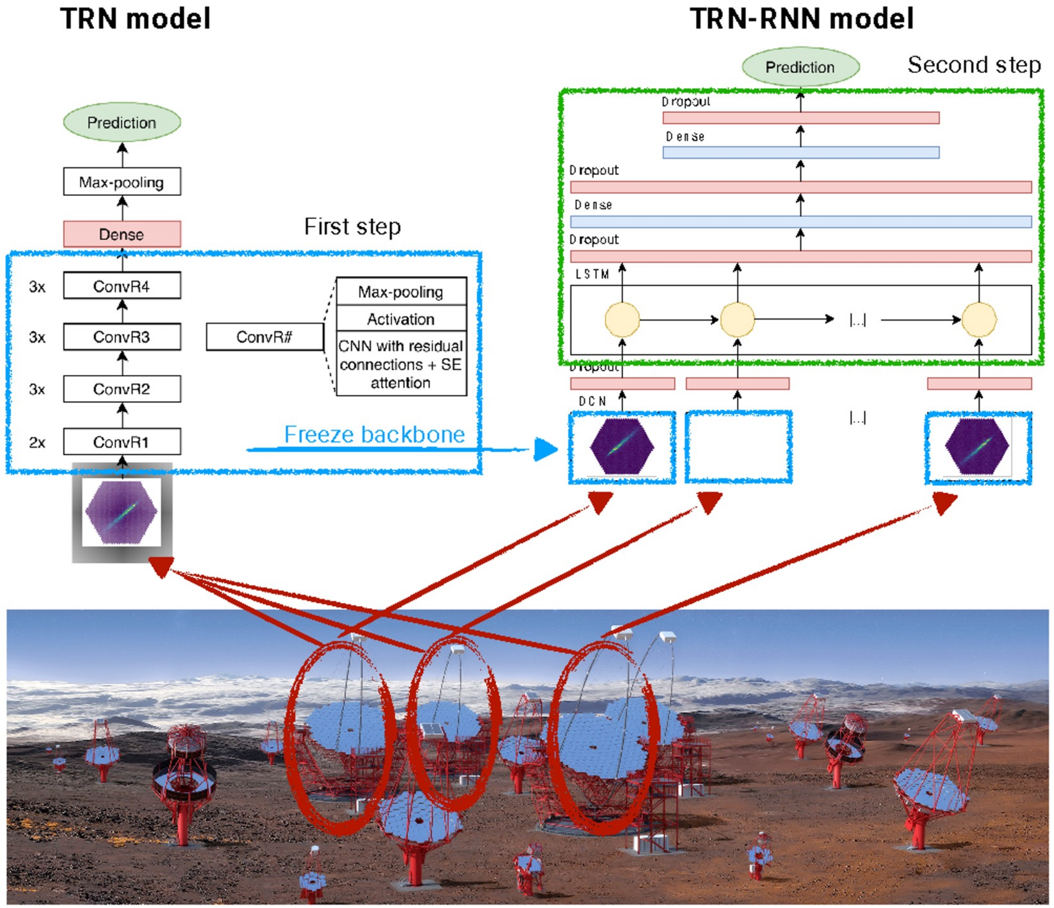
D. Nieto et al. ADASS XXX 2020

CTLearn: single-telescope full-event reconstruction



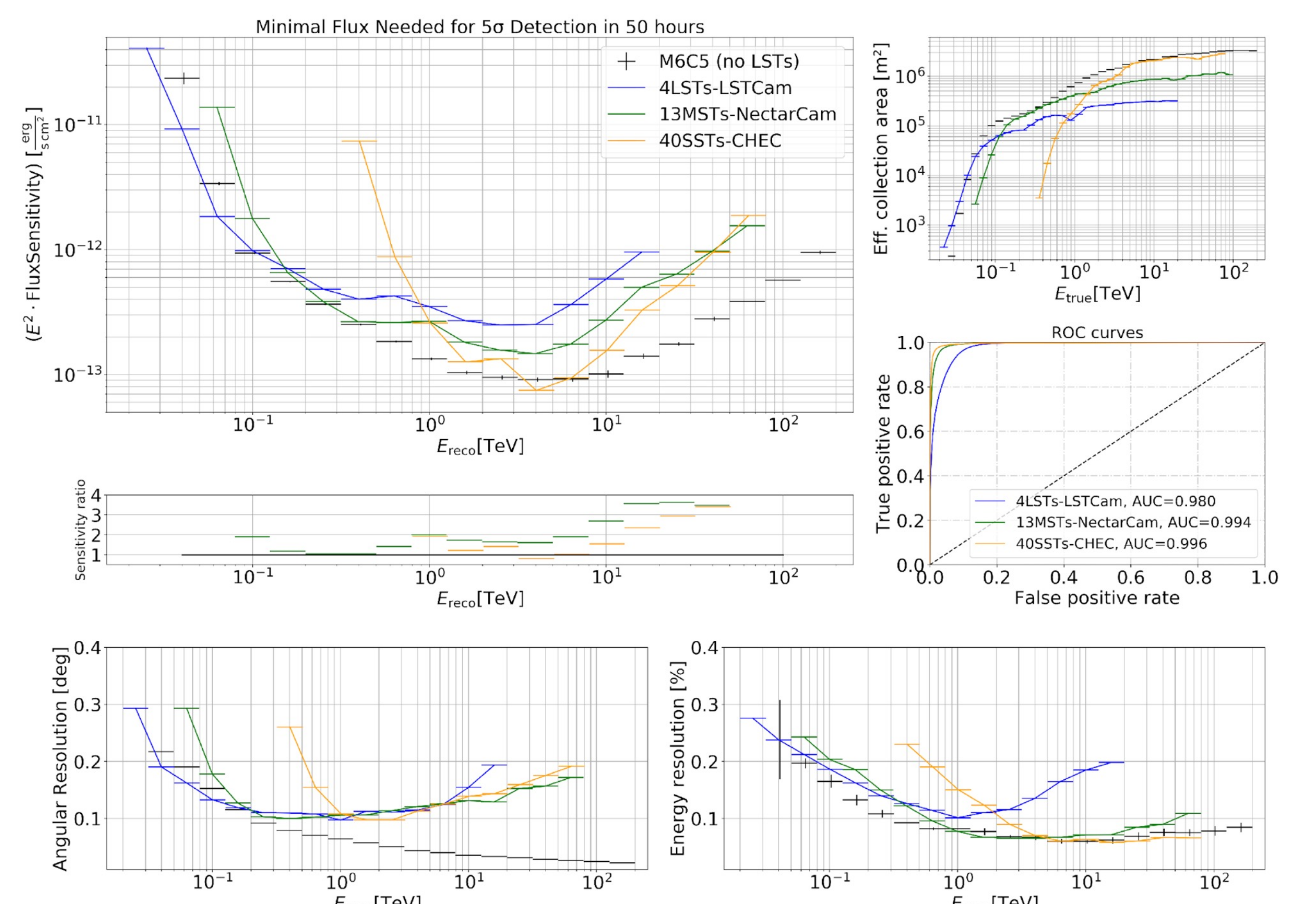


T. Miener et al., PoS(ICRC2021) 730

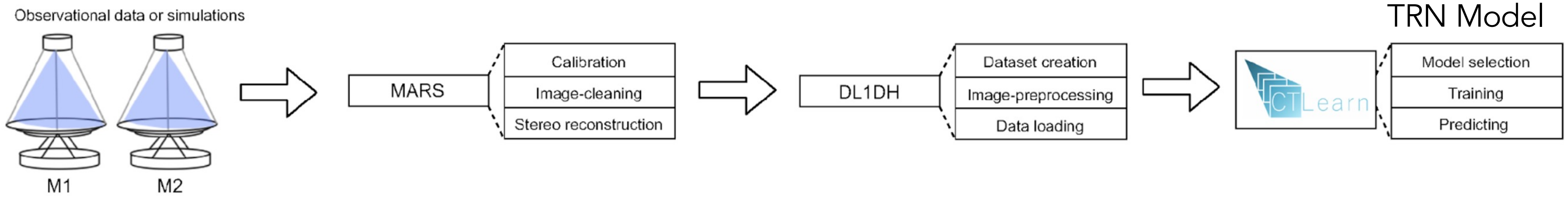


T. Miener et al., PoS(ICRC2021) 730

CTLearn: multiple-telescope full-event reconstruction

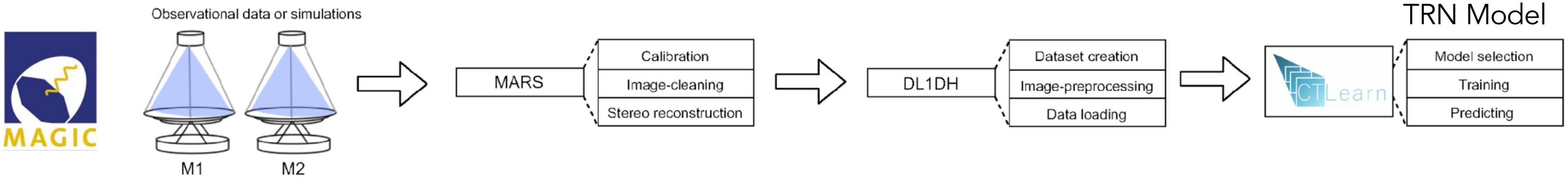


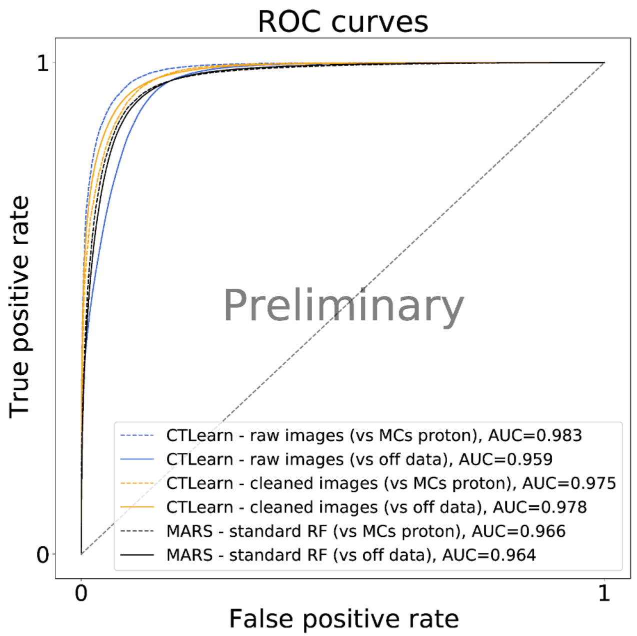
T. Miener et al., PoS(ICRC2021) 730

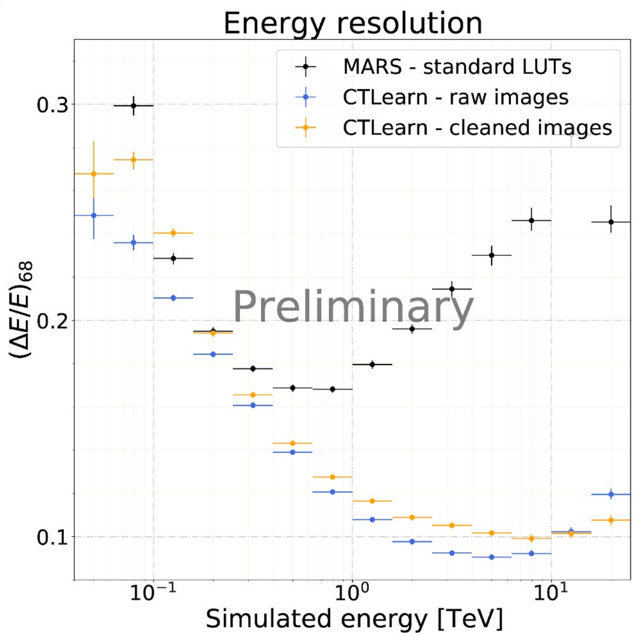


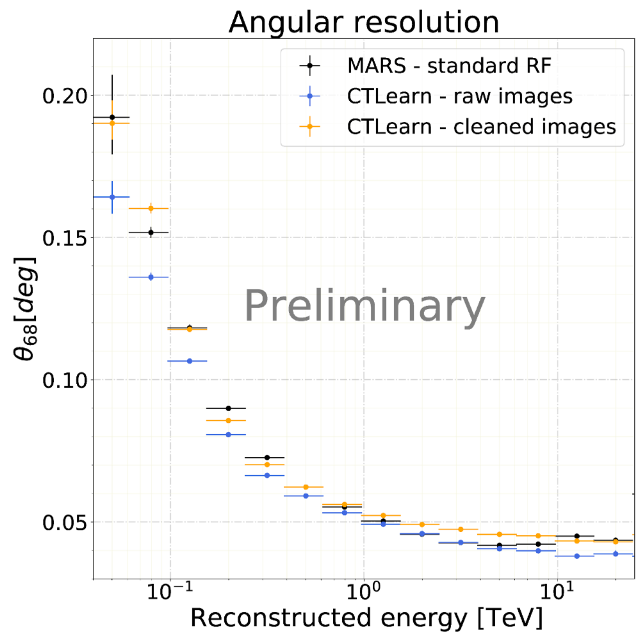
- 2 IACTs in La Palma, Canary Islands, Spain
- Energies > 30GeV

CTLearn: application to real data



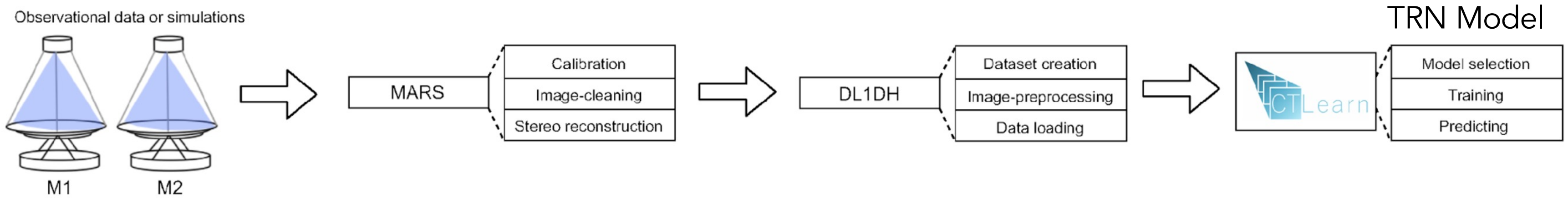


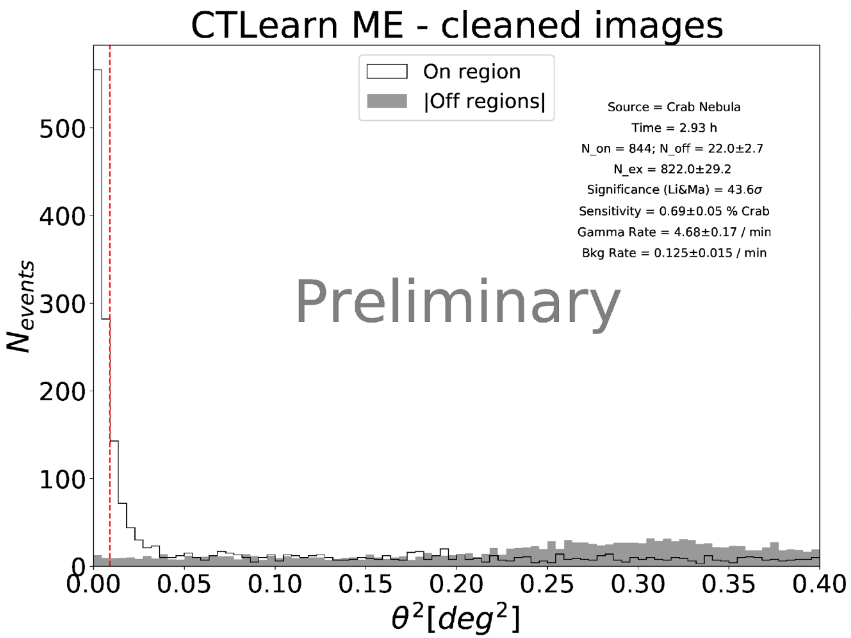




T. Miener et al. 2021 (ADASS XXXI)

CTLearn: application to real data





Analysis	γ rate [/min]	bkg rate [/min]	Sen. [% Crab]	Sig. (Li&Ma)
MARS – ME	4.54 ± 0.16	0.119 ± 0.015	0.70 ± 0.05	43.0σ
CTLearn – ME (raw)	3.45 ± 0.14	0.133 ± 0.018	0.97 ± 0.08	36.5σ
CTLearn – ME (cleaned)	4.68 ± 0.17	0.125 ± 0.015	0.69 ± 0.05	43.6σ
MARS – LE	16.49 ± 0.35	3.861 ± 0.086	1.09 ± 0.03	61.1σ
CTLearn – LE (raw)	11.70 ± 0.32	3.832 ± 0.114	1.53 ± 0.05	47.5σ
CTLearn – LE (cleaned)	16.24 ± 0.35	3.872 ± 0.086	1.11 ± 0.03	60.4σ

Analysis	N_{on}	N_{off}	N_{ex}
MARS – ME	819	21.0 ± 2.6	798.0 ± 28.7
CTLearn – ME (raw)	629	23.3 ± 3.1	605.7 ± 25.3
CTLearn – ME (cleaned)	844	22.0 ± 2.7	822.0 ± 29.2
MARS – LE	3579	679.0 ± 15.0	2900.0 ± 61.7
CTLearn – LE (raw)	2730	673.7 ± 20.0	2056.3 ± 56.0
CTLearn – LE (cleaned)	3536	680.7 ± 15.1	2855.3 ± 61.3

Summary of all performed analyses of the same Crab Nebula sample

T. Miener et al. 2021 (ADASS XXXI)

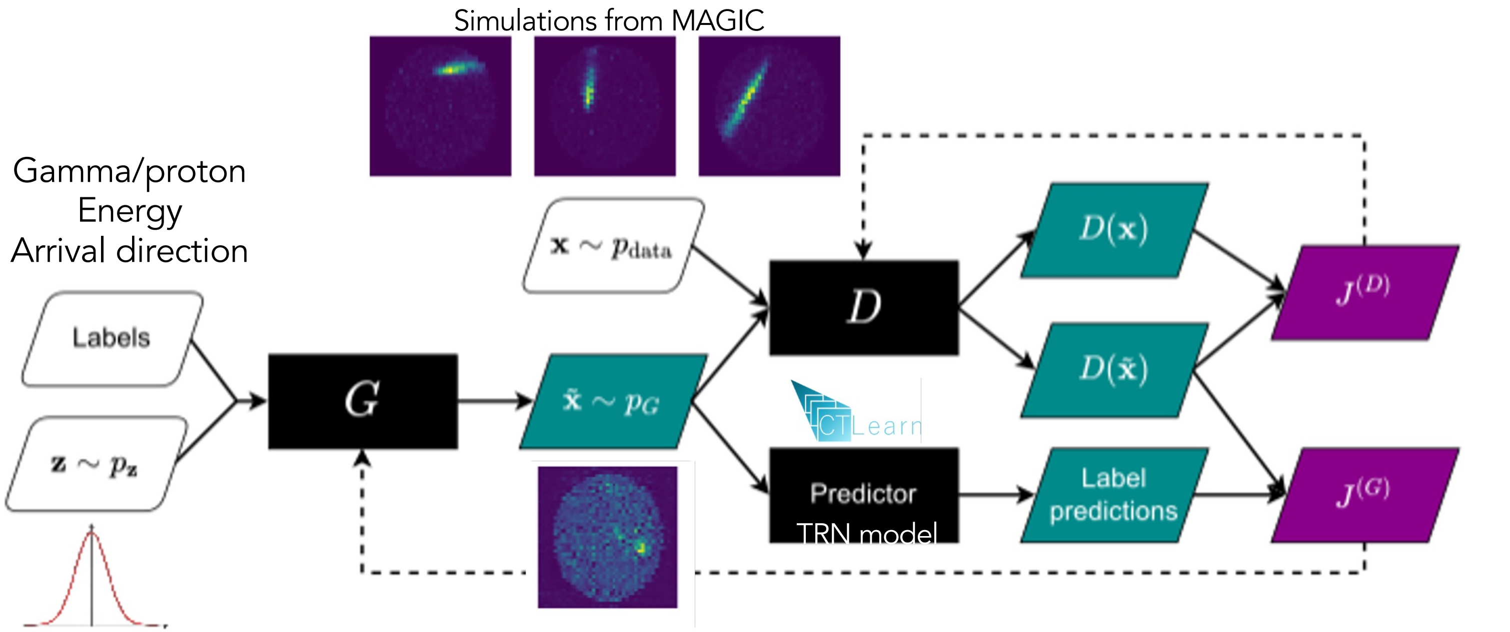
Next step -> find the best performing model for event reconstruction

The curse of dimensionality haunts us here too!

- Hyperparameter space for deep learning architecture design
  - Number of CNN layers
  - Kernel size
  - Activation function
  - Dropout rate
  - Number of FC layers
  - Batch size
  - Learning rate
  - Optimizer
  - …

- Optimization strategies
  - Grid searches
  - Random searches
  - Bayesian optimization
  - Evolutionary algorithms
  - Reinforcement learning
  - …

- Auxiliary conditional generative adversarial networks (AC-GANs)

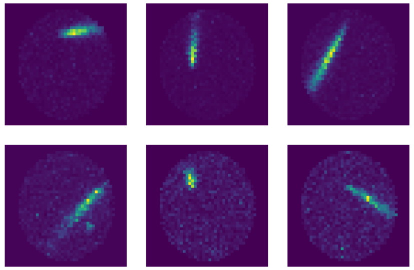


S. García-Heredia et al. (degree thesis)

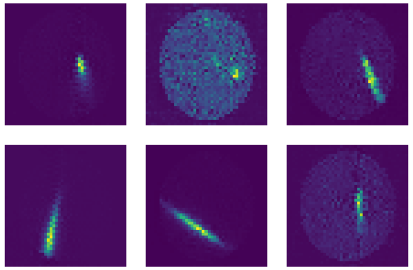
Generating IACT events with GANs

GAMMA RAYS

Simulated

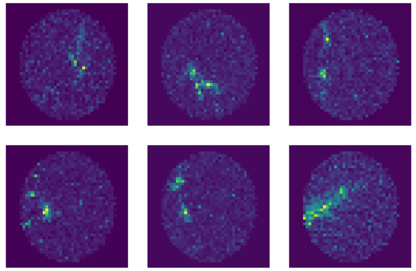


Generated

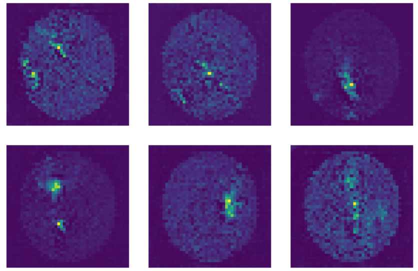


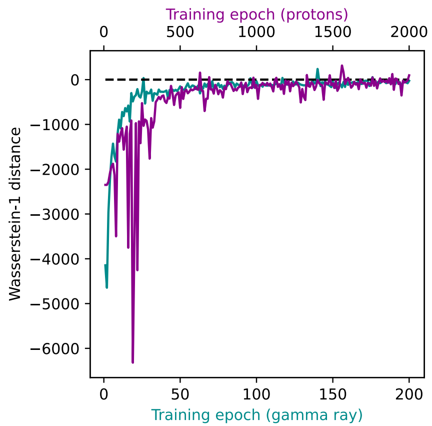
PROTONS

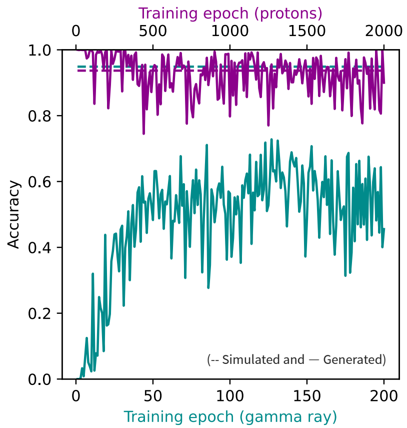
Simulated

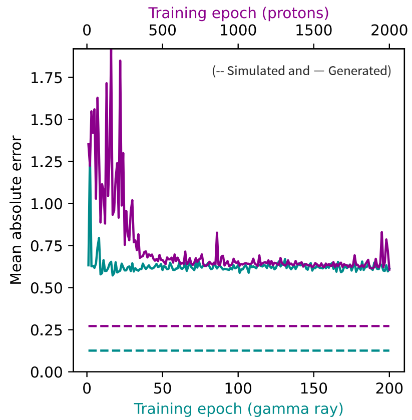


Generated





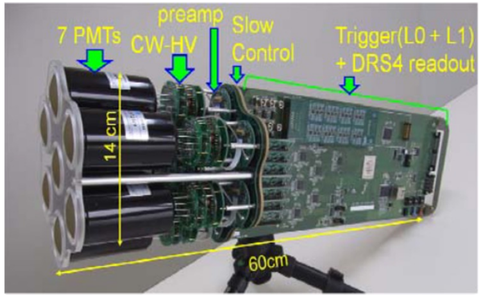


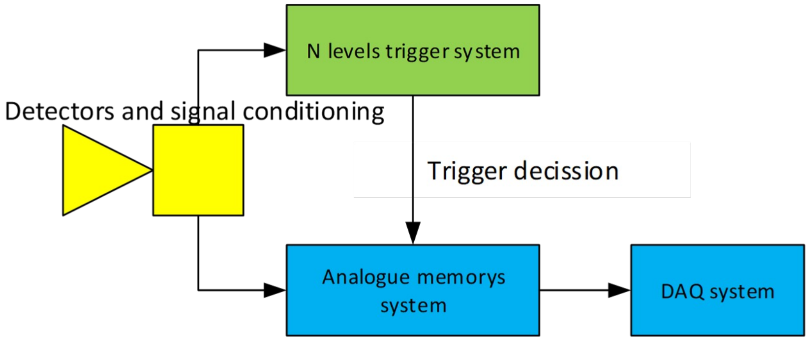


S. García-Heredia et al. (degree thesis)

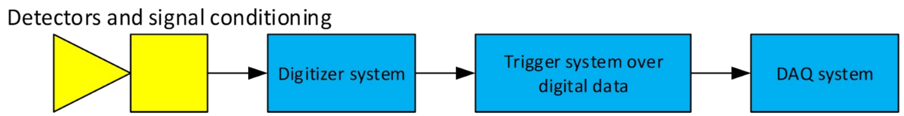
AI Trigger

- Current trigger – Signal filter



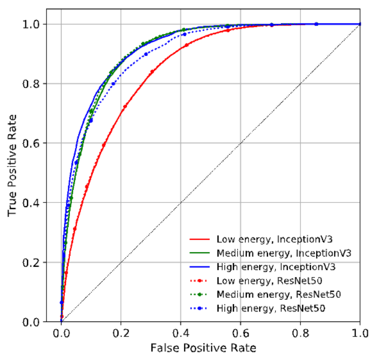


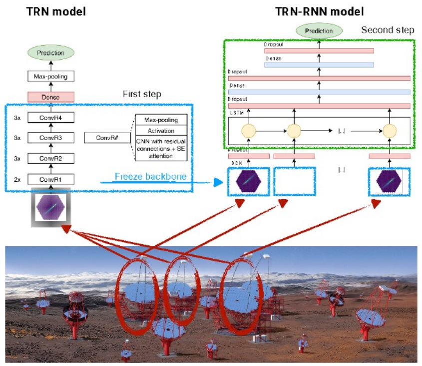
- Future steps - AI trigger – AI integration within the trigger logic – Fully digital
- CNN + FPGA technology
  - (1) Signal and Background noise separation
  - (2) Gamma/hadron classification
- Work in progress

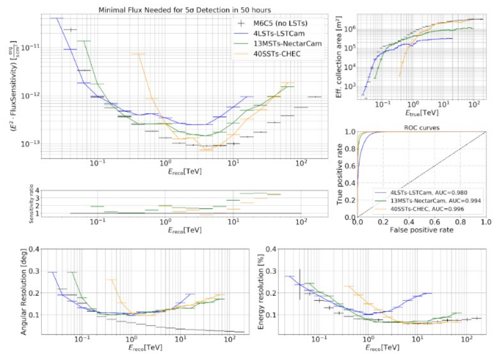


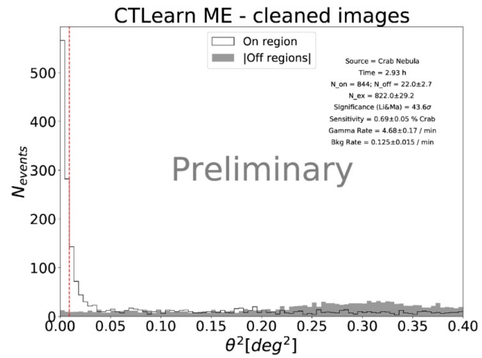
Conclusions & Outlook

- Current-generation IACTs have enhanced their performances through ML
- Next-gen (even current-gen!) IACT may profit from latest developments in ML
- Ongoing efforts to exploit deep learning as an event reconstruction method for IACTs
  - Full-event reconstruction over simulated IACT events demonstrated
  - Application to real observations works!
  - Working on optimizing architectures & multi-task learning
  - Using AC-GANs as pseudosimulators
  - Tackling the real-data problem









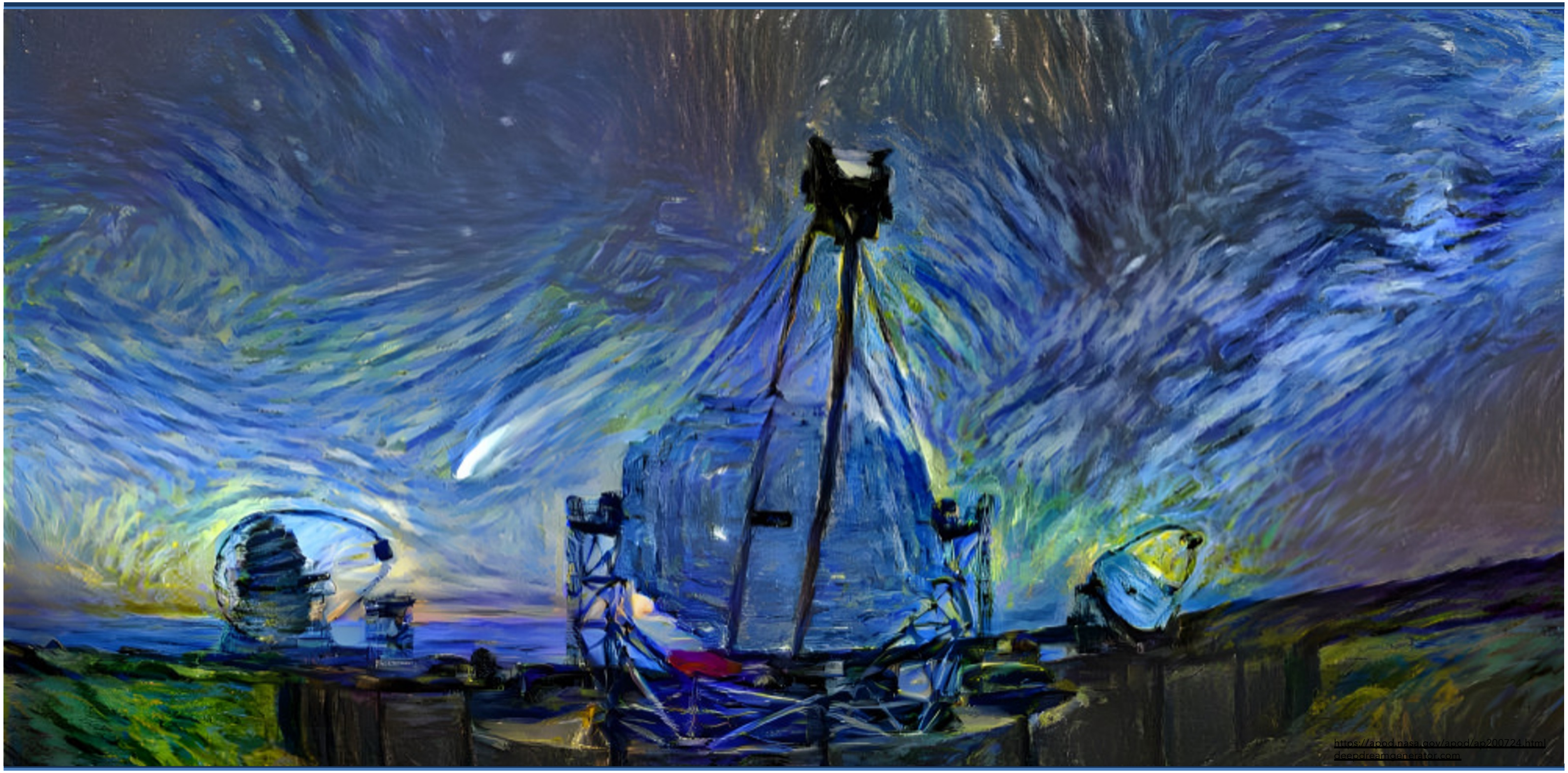
THANK YOU

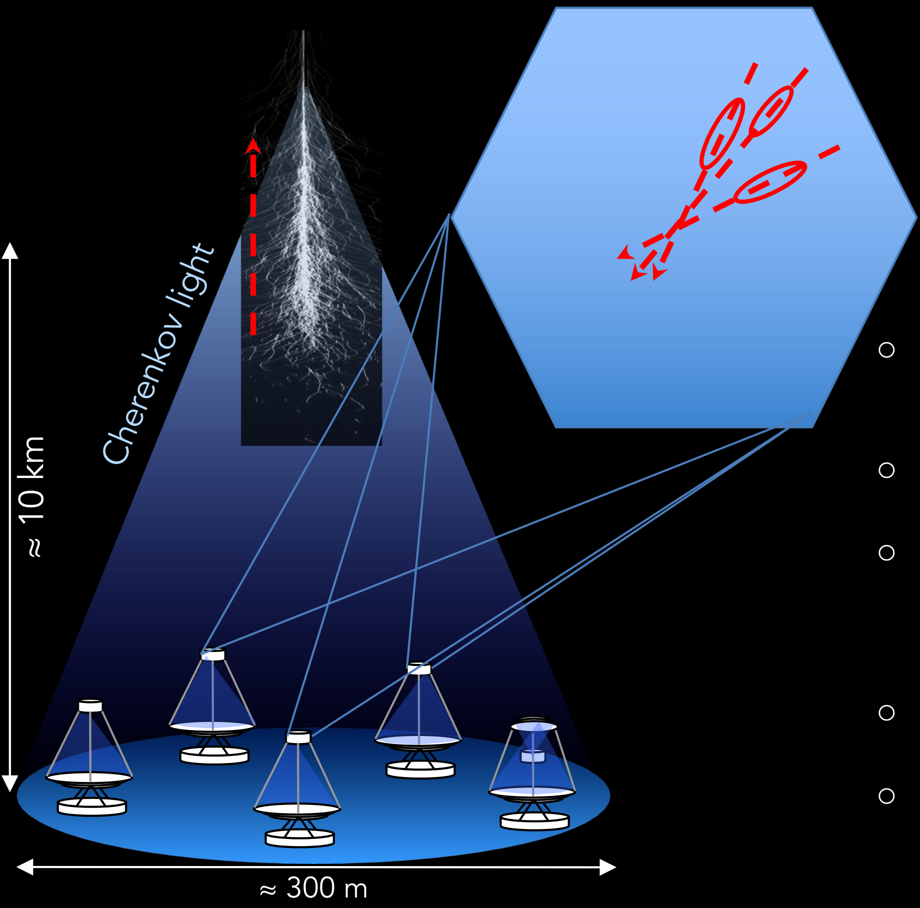
rcervino@ucm.es

Acknowledgment

The research here presented has been partially supported by the former Spanish Ministry of Economy, Industry, and Competitiveness / ERDF grants FPA2015-73913-JIN and FPA2017-82729-C6-3-R, the Spanish Ministry of Science and Innovation grant PID2019-104114RB-C32, PID2022-138172NB-C42, PDC2023-145839-I00, NSF awards PHY-1229205, 1229792, and 1607491, the Community of Madrid grant C. M. (2023) PEJ-2023-AI, and the European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures funded by the European Union's Horizon 2020 research and innovation program under Grant Agreement no. 824064. The authors acknowledge support from Google LLC through the Google Summer of Code program and NVIDIA Corporation with the donation of a Titan X Pascal GPU used for part of this research.

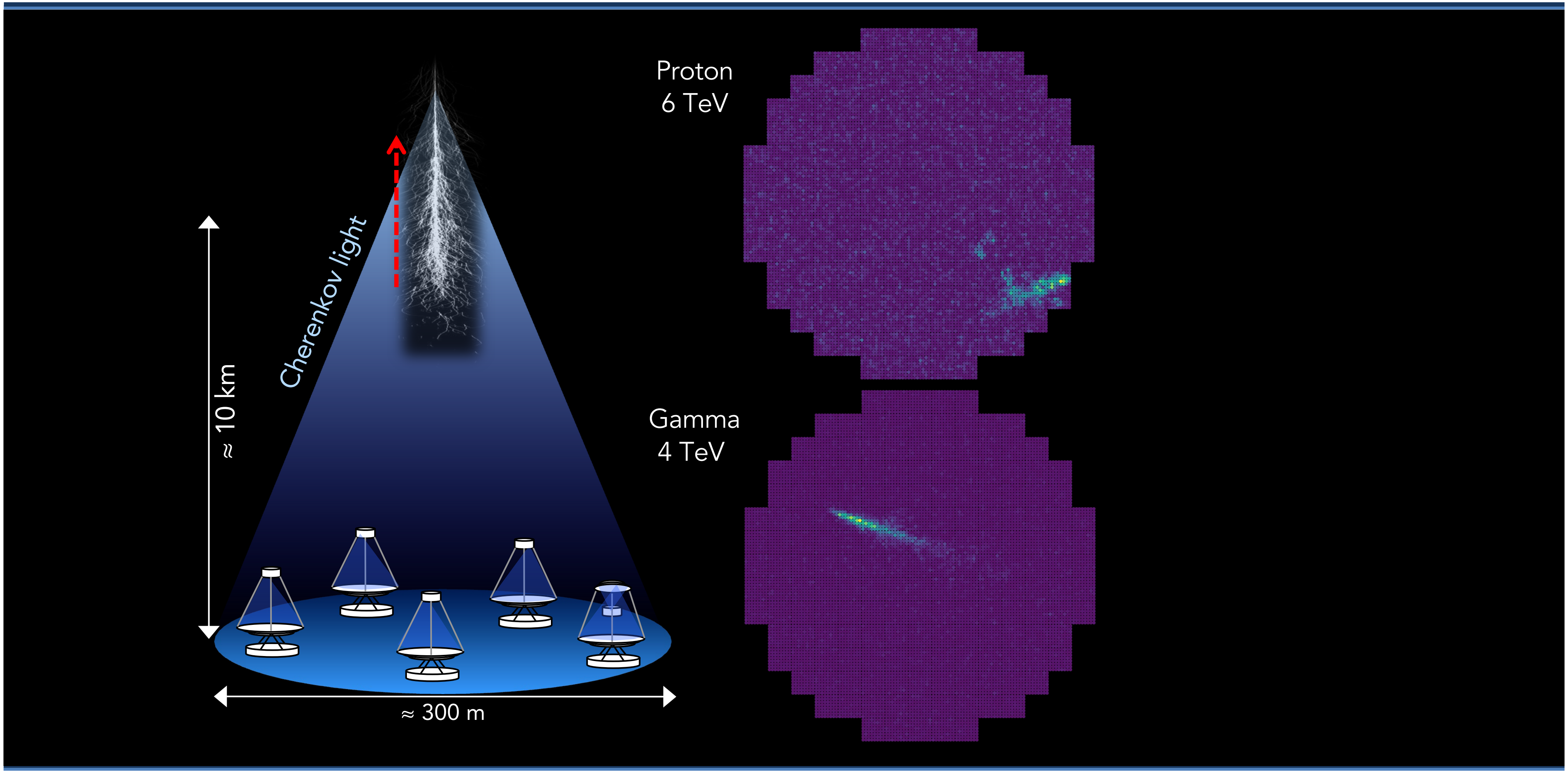
Backup


https://apod.nasa.gov/apod/ap200724.html
deepdreamgenerator.com

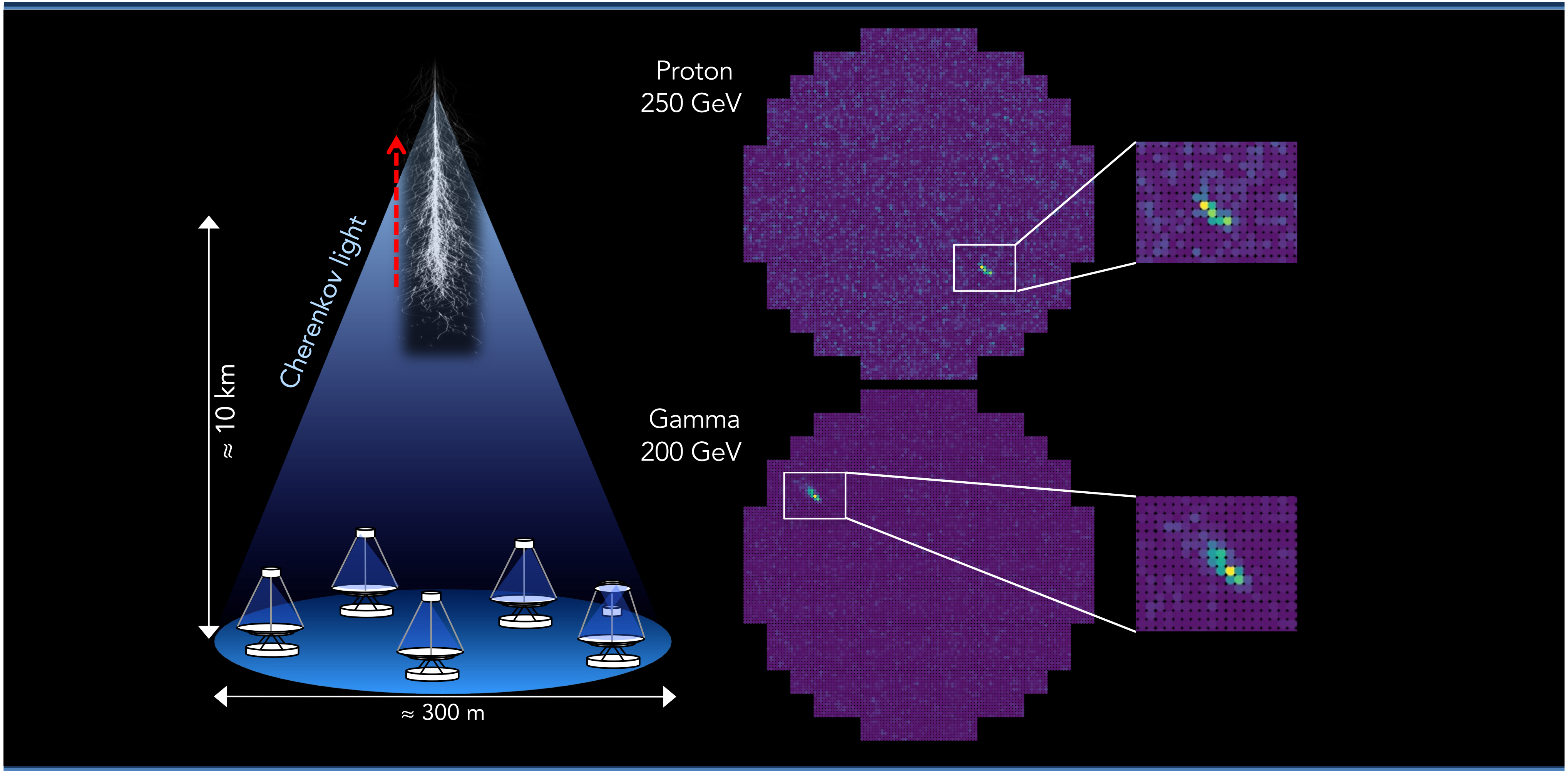


- Detection of extended air showers using the atmosphere as a calorimeter
- Huge γ–ray collection area (~10^5 m^2)
- Large background from charged CR
  - Partly irreducible (e^-/e^+ , single-EM, with current methods)
- Energy window: tens GeV - tens TeV
- Event reconstruction from image:
  - Type of primary event
  - Primary energy estimation
  - Primary arrival direction

Imaging atmospheric Cherenkov technique

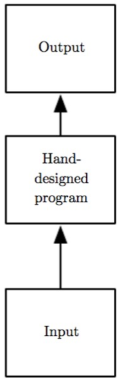


Imaging atmospheric Cherenkov technique

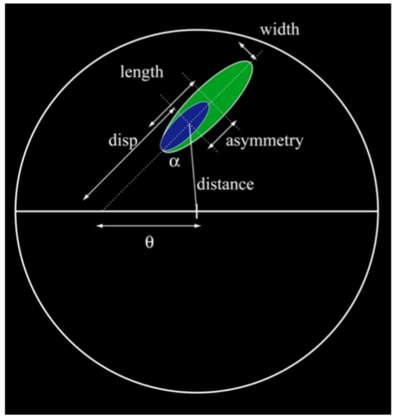


Output: event type, energy, arrival direction

- Event type: box cuts
- Event energy: parametrization
- Event direction: parametrization



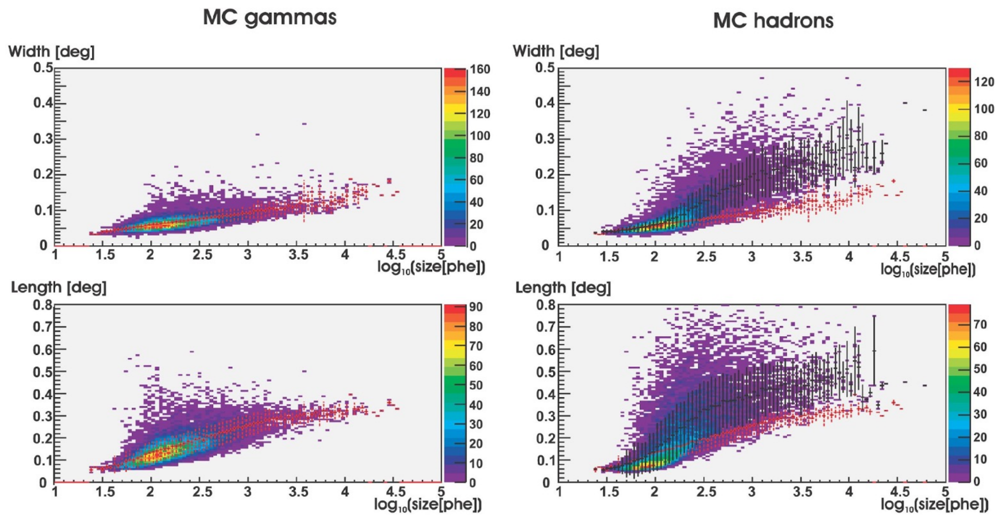
Rule-based systems



Input: observed events

event reconstruction

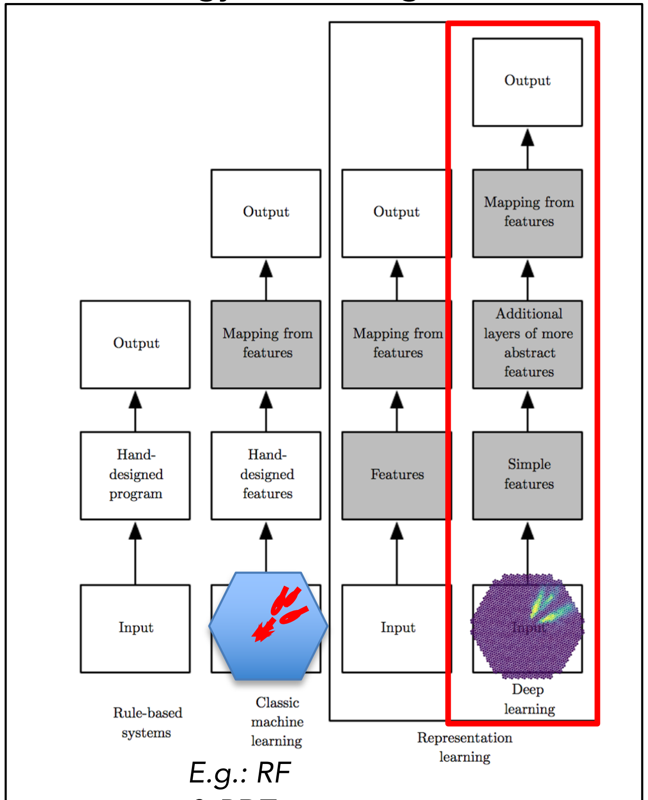
- Based on image parametrization (Hillas parameters)



- Current generation of IACTs: classic ML
  - ML method:
    - Random Forest (RF)
  - Instrument calibration with real data not possible
  - Strong dependency on Montecarlo simulations

Enhancing IACT performance with deep learning?

Output: event type, energy, incoming direction

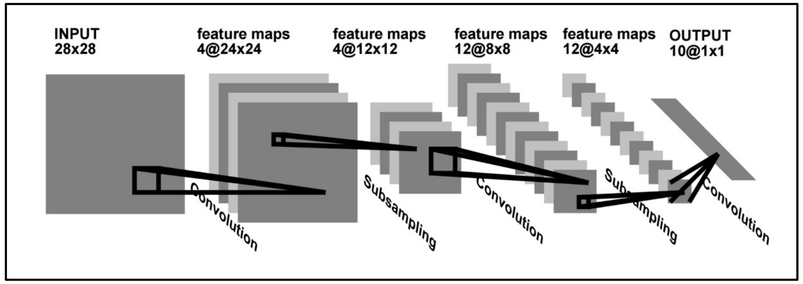


E.g.: RF & BDT

Input: observed events

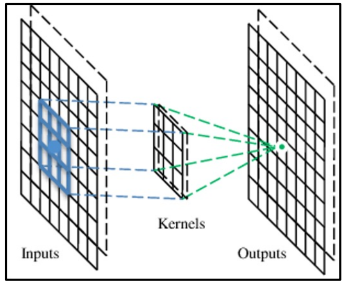
event reconstruction

Convolutional Neural Network (CNN)



LeCunn et al.

Convolution



Guo et al.

- DL capable of ***extracting*** and mapping image features automatically with unprecedented classification accuracy. Hyper-active CS research field constantly improving
- Many HEP/Astro experiments already exploring/utilizing the technique (LIGO, LHC, MicroBooNe, NOVa, etc…)

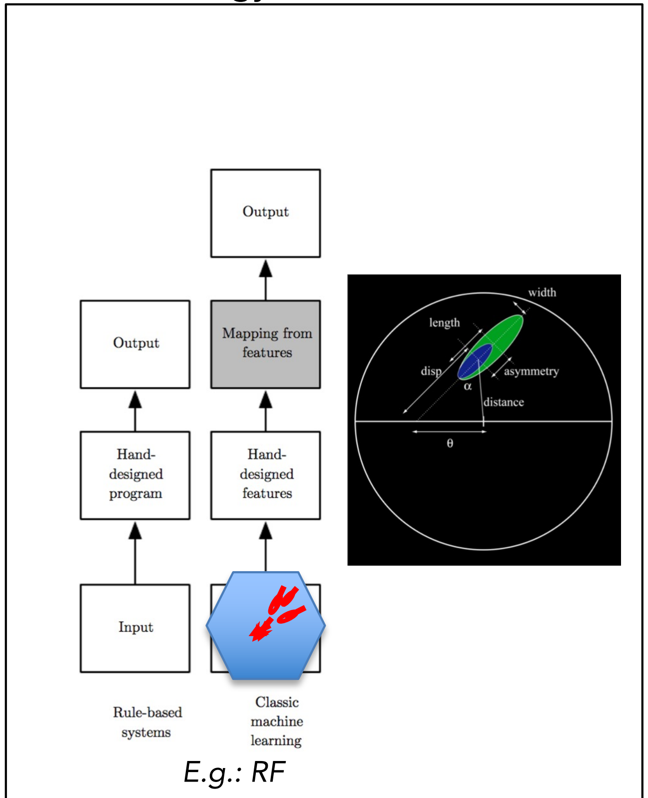
Method:

- Use deep learning to reconstruct IACT events from non-parameterized images
  - Performance enhancement -> better sensitivity

But there are risk…

- MC reliability (e.g. network selecting some features from your MC not present in real data)

Output: event type, energy, arrival direction



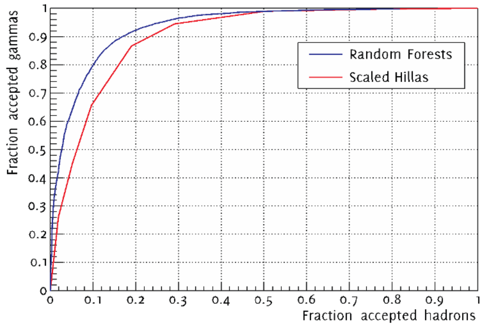
E.g.: RF & BDT

Input: observed events

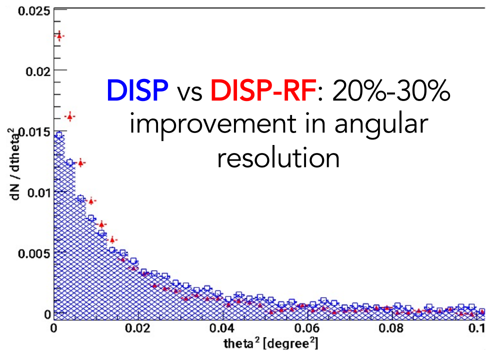
event reconstruction

- Current generation of IACTs: classic ML

- ML method:
  - Random Forest (RF)
- Applied to:
  - Background rejection
  - Arrival direction

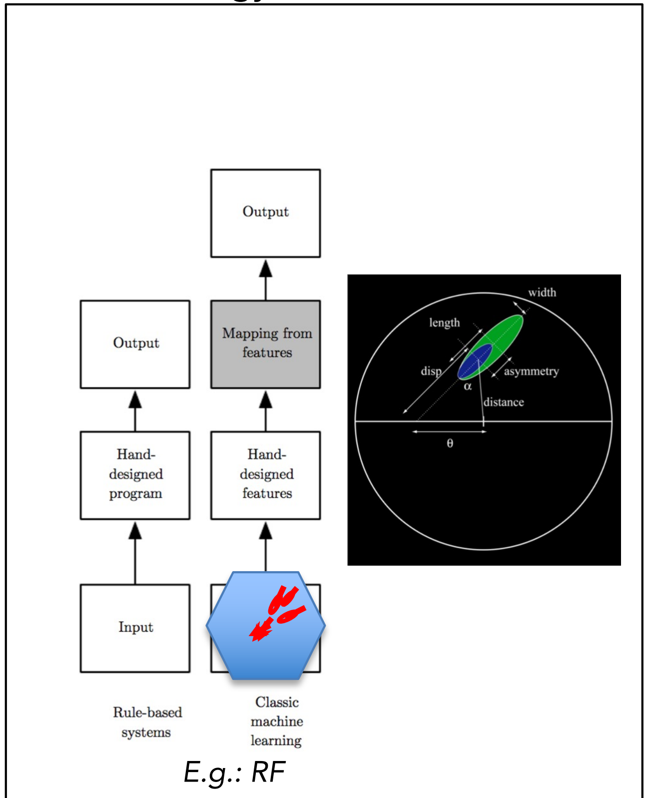


Albert et al., NIM-A 588:424-432 (2008)



Aleksic et al., A&A 524 A77 (2010)

Output: event type, energy, arrival direction



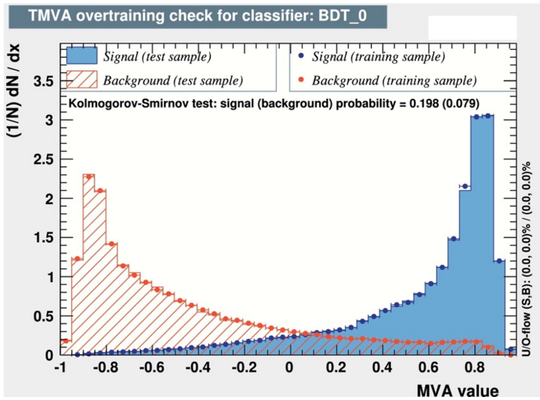
E.g.: RF & BDT

Input: observed events

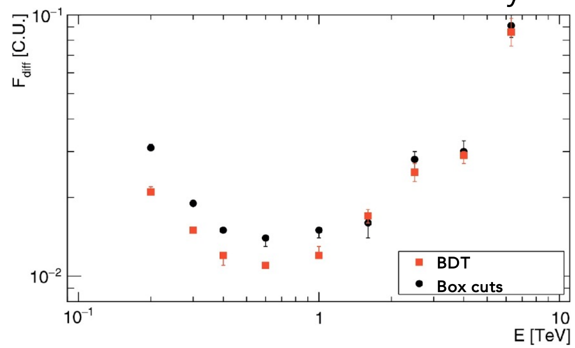
event reconstruction

- Current generation of IACTs: classic ML

- ML method:
  - Boosted Decision Trees (BDT)
- Applied to:
  - Background rejection

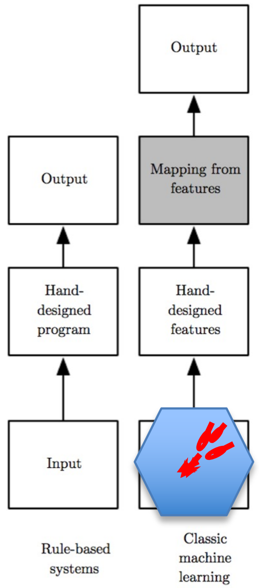


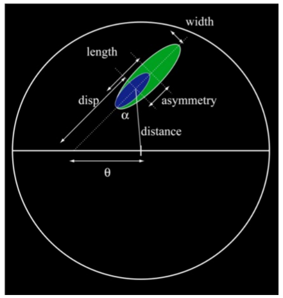
VERITAS diff. sensitivity



Krause et al., APP V89 P1-9 (2017)

Output: event type, energy, arrival direction



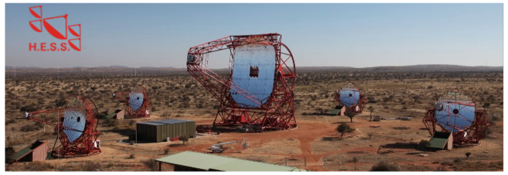


E.g.: RF & BDT

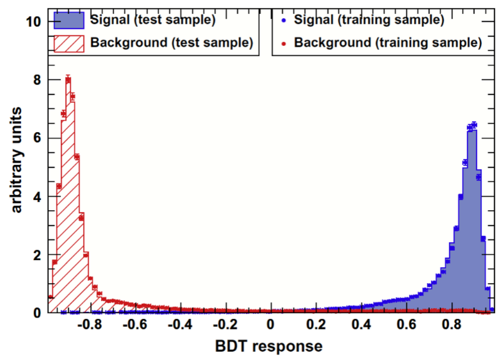
Input: observed events

event reconstruction

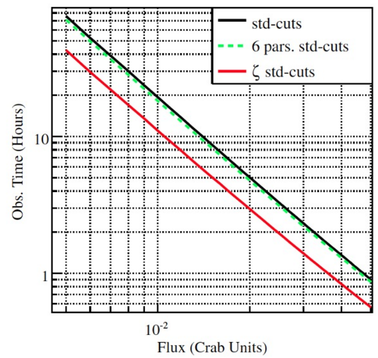
- Current generation of IACTs: classic ML



- ML method:
  - Boosted Decision Trees (BDT)
- Applied to:
  - Background rejection



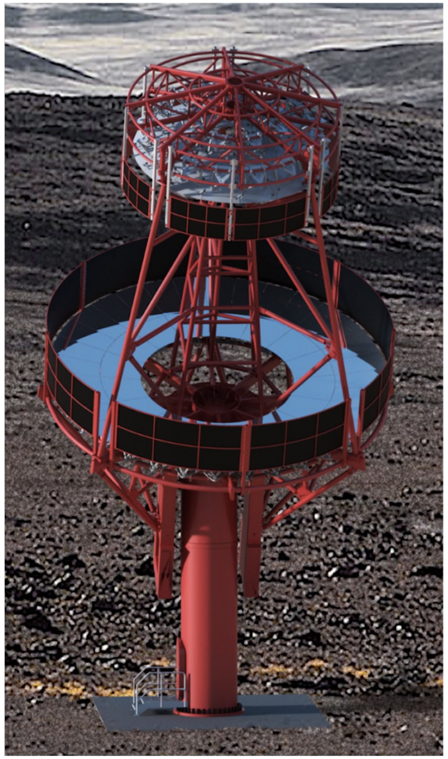
Becherini et al., APP V34-12 P858-870 (2011)



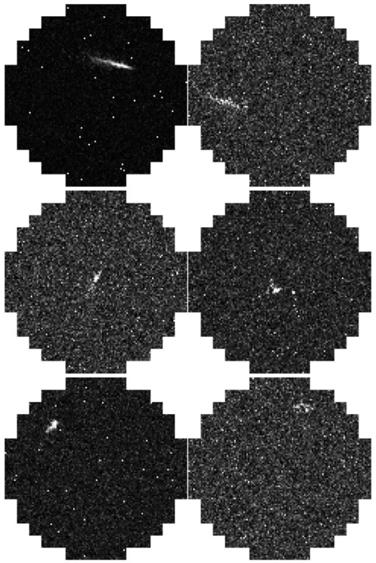
Ohm et al., APP V31-5 P383-391 (2009)

(Results for H.E.S.S. I only)

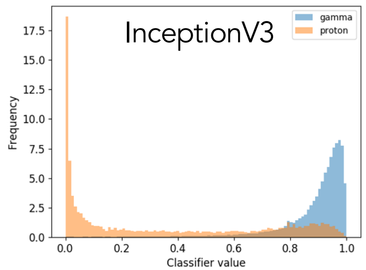
Proof of concept: gamma/hadron classification in SC-MST

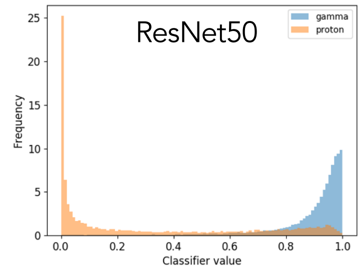


- Single telescope
- Square pixels
- Only signal charge (no timing)
- Single task: classification



Medium energies
(0.3 TeV < E < 1 TeV)





AUC

Model/Energy	Low E.	Med. E.	High E.
InceptionV3	84.7%	91.1%	92.0%
ResNet50	84.8%	91.4%	90.2%

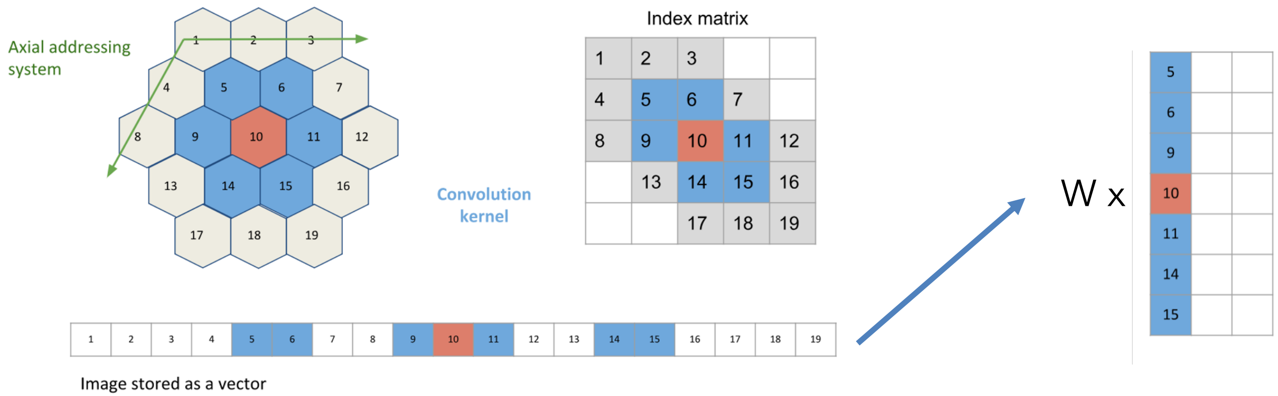
Nieto et al., PoS(ICRC2017)809

Tackling the hexagonal-pixel challenge

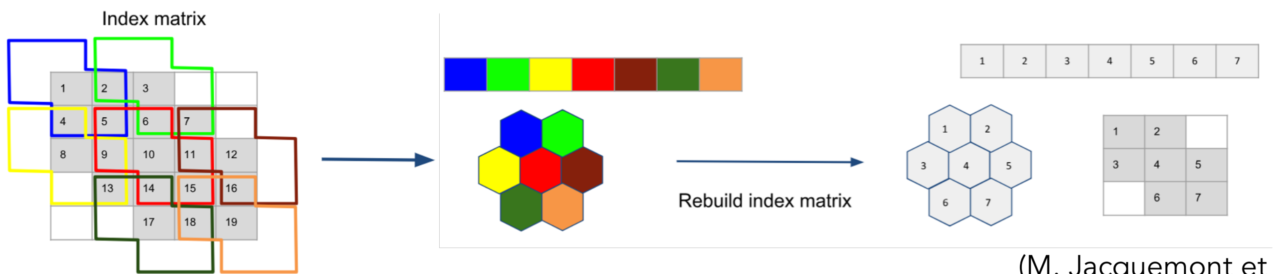
T. Vuillaume,
M. Jaquemont, et al.

https://github.com/IndexedConv

- Hexagonal convolution
  - Convolution



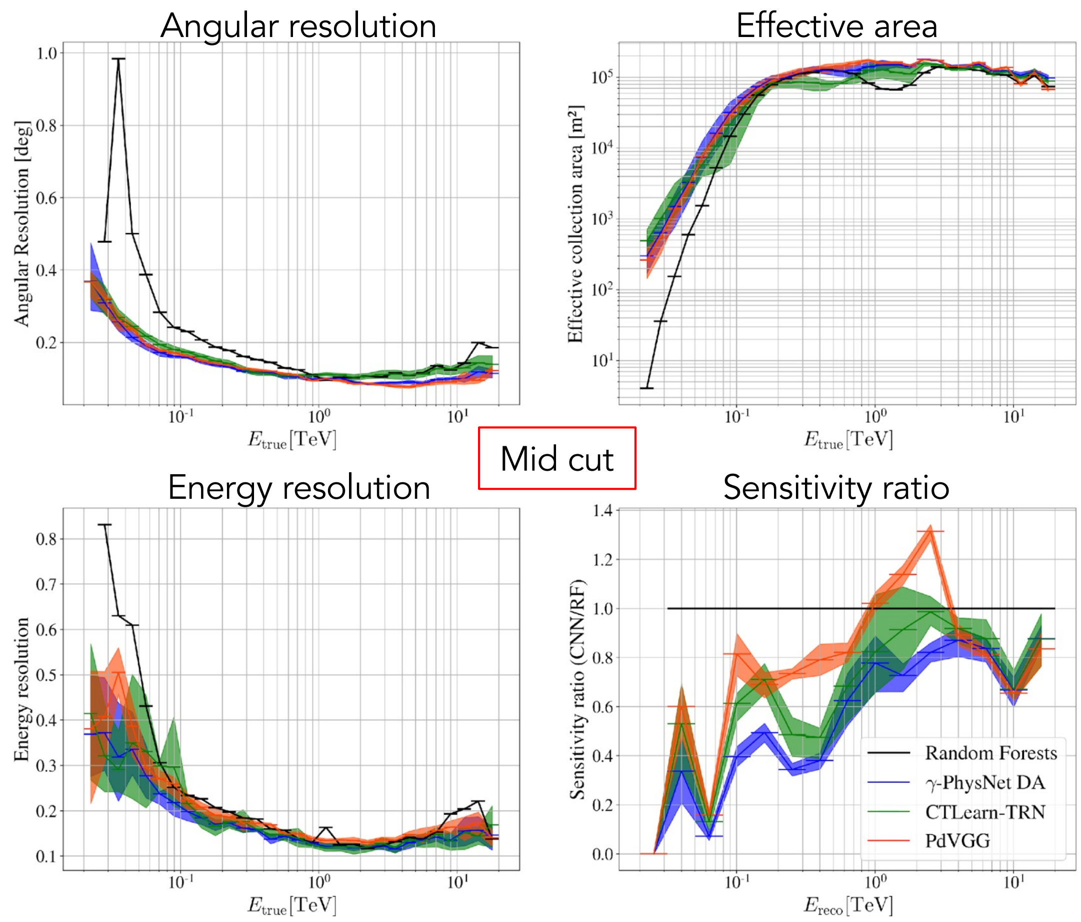
  - Pooling



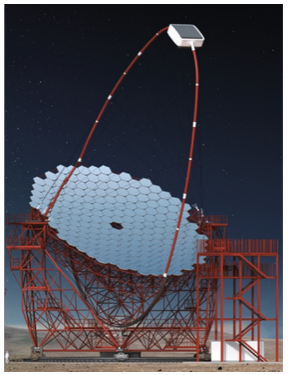
(M. Jacquemont et al. 2019)

CTLearn: crosschecking results

- Crosschecking three different implementations
- Same datasets, same cuts
- Different models
- Comparison against standard analysis (RF)



P. Grespan et al. PoS(ICRC2021) 771



CTLearn Optimizer

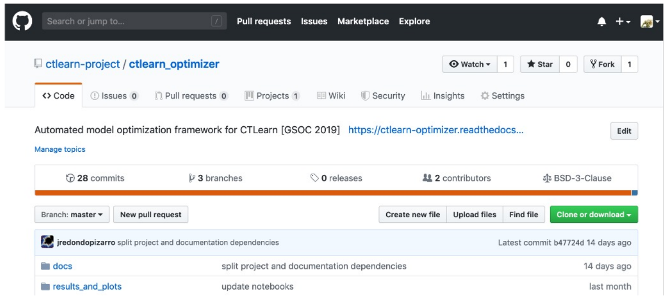
- Framework for hyperparameter optimization of CTLearn models
  (Although can be adapted to any config-file based DCN framework)

- Based on Tune: a scalable hyperparameter tuning library

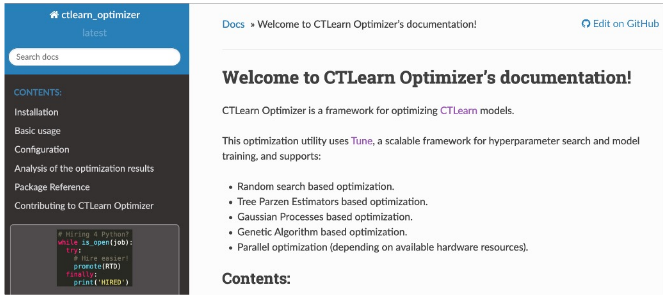
- Supported optimization strategies:
  - Random search
  - Tree Parzen Estimators
  - Gaussian Processes
  - Genetic Algorithms
  - Parallel optimization (depending on available hardware)

Bayesian optimization

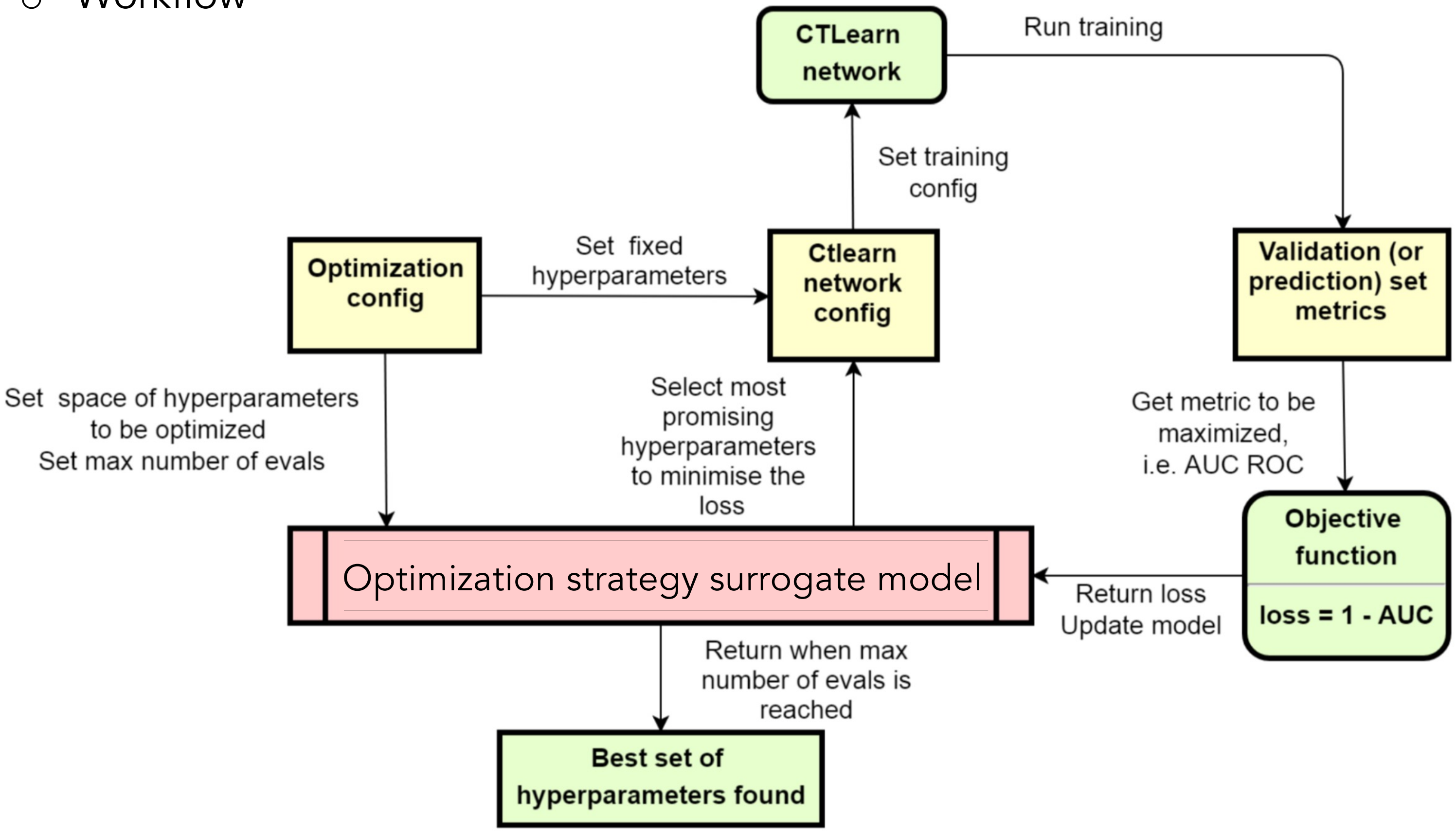
github.com/ctlearn-project/ctlearn_optimizer



ctlearn-optimizer.readthedocs.io



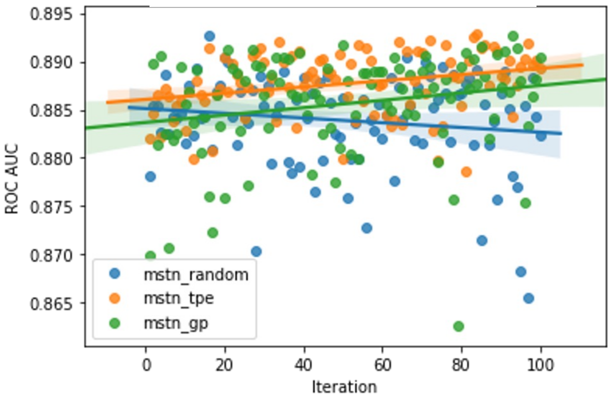
- Workflow


CTLearn network
Run training
Set training config
Optimization config
Set fixed hyperparameters
Ctlearn network config
Validation (or prediction) set metrics
Set space of hyperparameters to be optimized
Set max number of evals
Select most promising hyperparameters to minimise the loss
Get metric to be maximized, i.e. AUC ROC
Objective function
loss = 1 - AUC
Optimization strategy surrogate model
Return loss
Update model
Return when max number of evals is reached
Best set of hyperparameters found

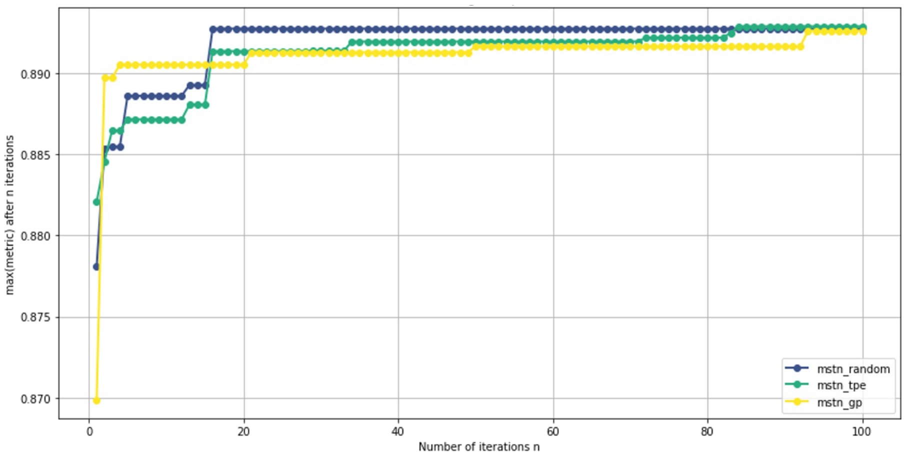
CTLearn Optimizer

- Visualization

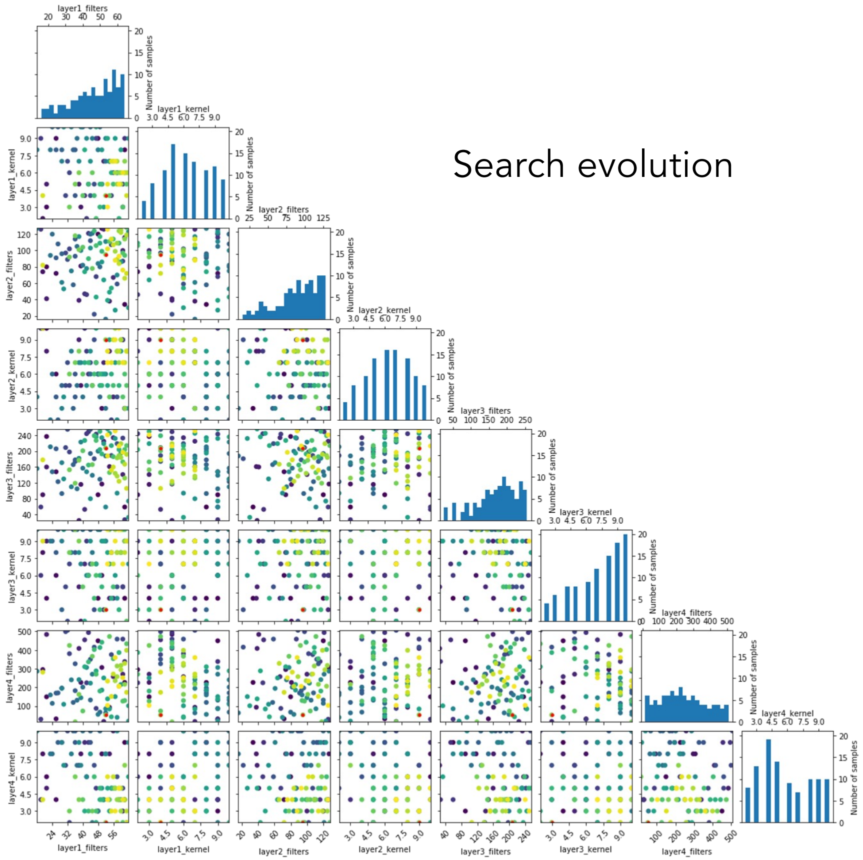
Evolution of the metric



Convergence of the metric

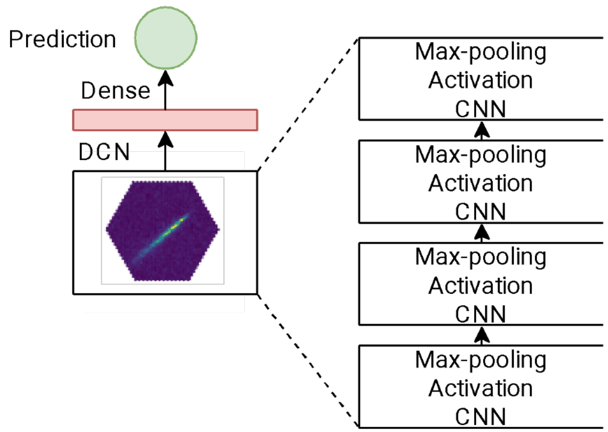


Search evolution



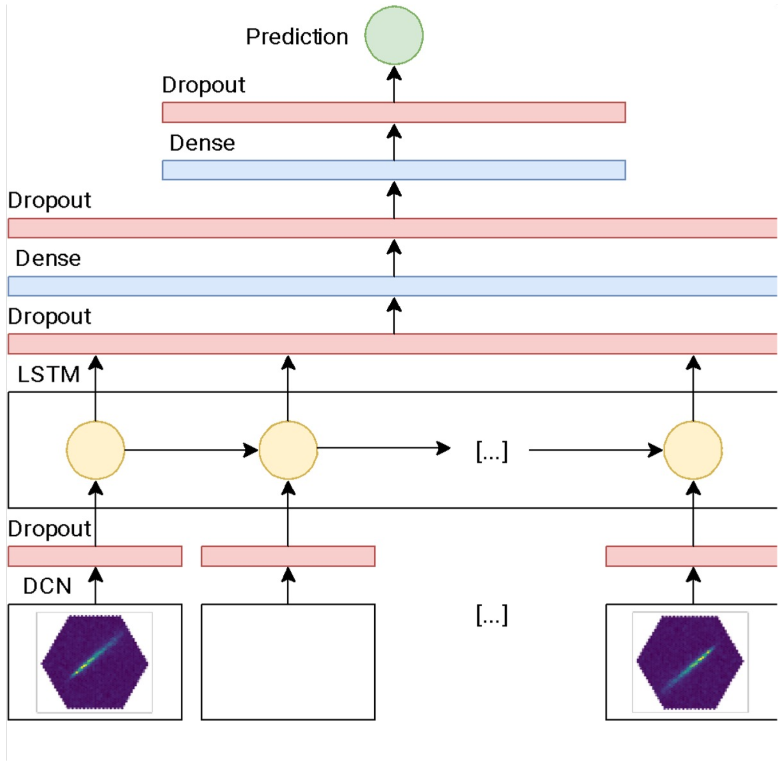
CTLearn Optimizer: some results

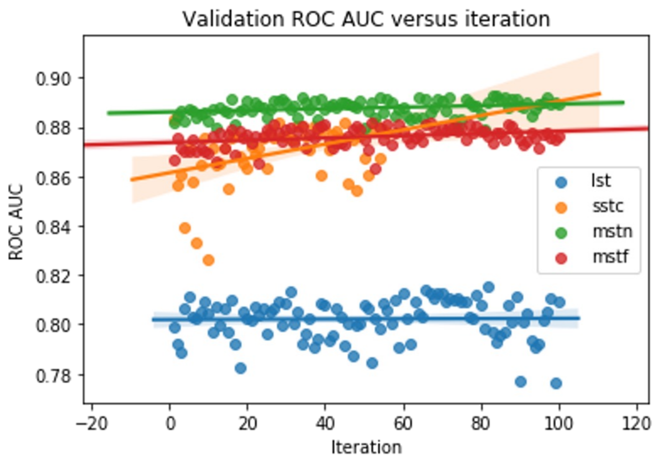
Single-tel model

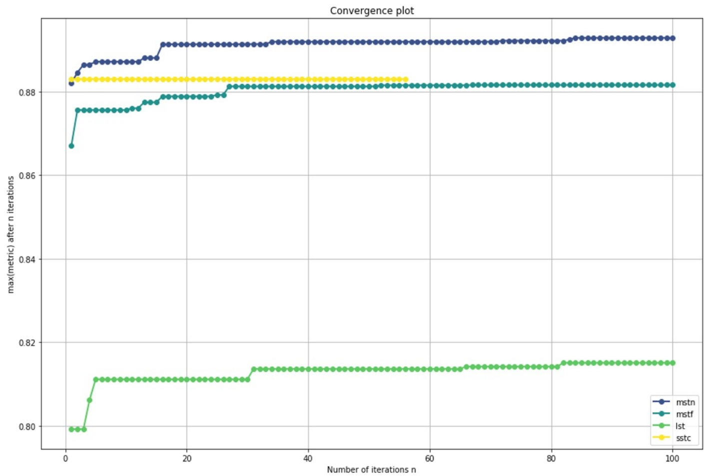


Number of kernels
Size of kernels
Number of CNN layers

CNN-RNN model





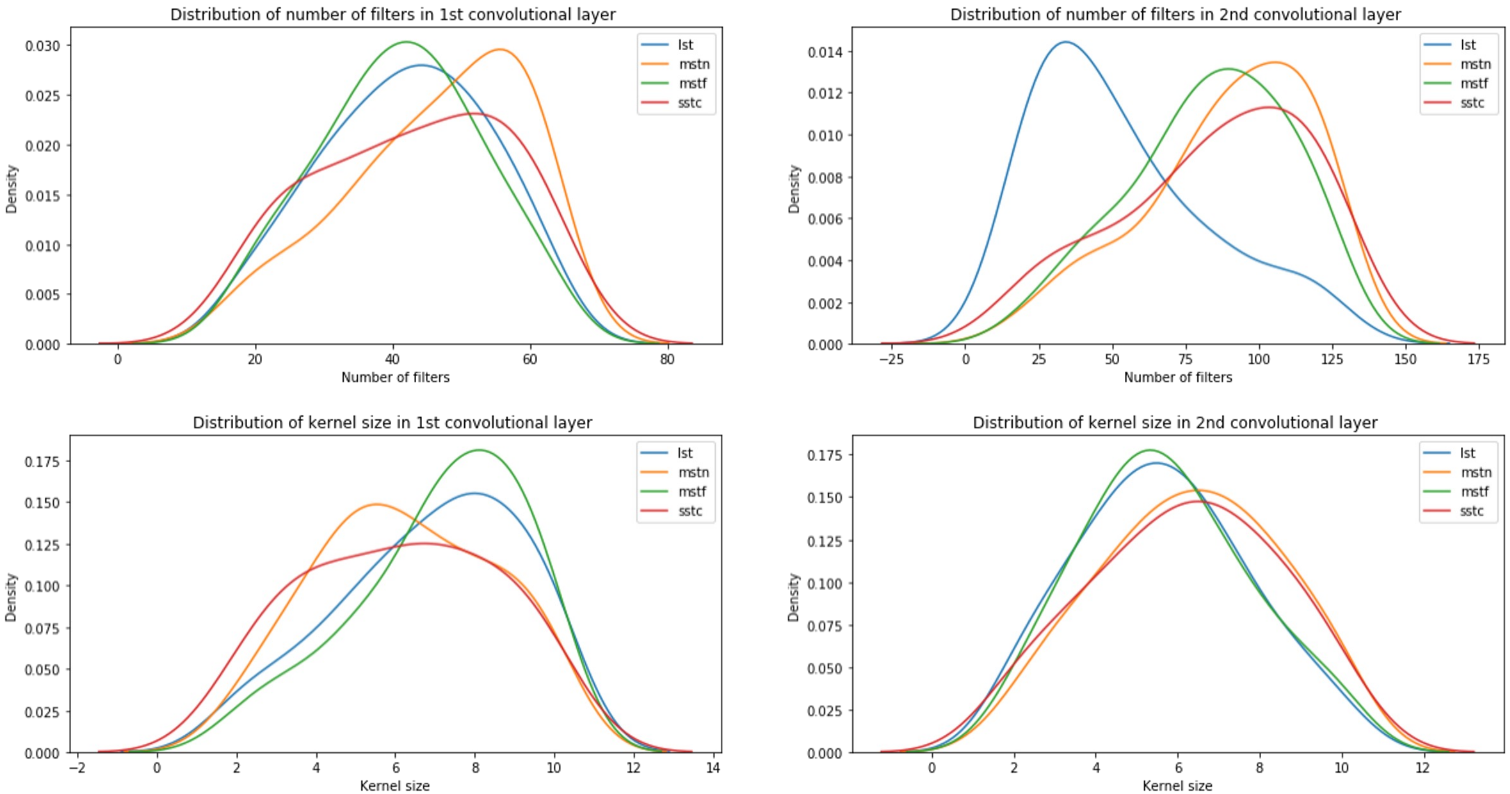


Hyperparameters	Telescope Type	Validation Accuracy	Validation AUC	Training Time
Base	LST	70.38%	0.7887	0h 41m 22s
Optimized	LST	72.45%	0.8150	0h 39m 14s
Base	SSTC	73.90%	0.8118	0h 42m 4s
Optimized	SSTC	79.87%	0.8830	1h 16m 4s
Base	MSTN	78.04%	0.8659	0h 58m 10s
Optimized	MSTN	80.11%	0.8929	0h 52m 48s
Base	MSTF	74.60%	0.8360	0h 55m 0s
Optimized	MSTF	79.01%	0.8816	0h 48m 37s

Telescope Type	Metric	Improvement
LST	Validation Accuracy	2.07%
LST	Validation AUC	2.63%
SSTC	Validation Accuracy	5.97%
SSTC	Validation AUC	7.12%
MSTN	Validation Accuracy	2.07%
MSTN	Validation AUC	2.70%
MSTF	Validation Accuracy	4.41%
MSTF	Validation AUC	4.56%

CTLearn Optimizer: some results

- Single_tel & TPE search



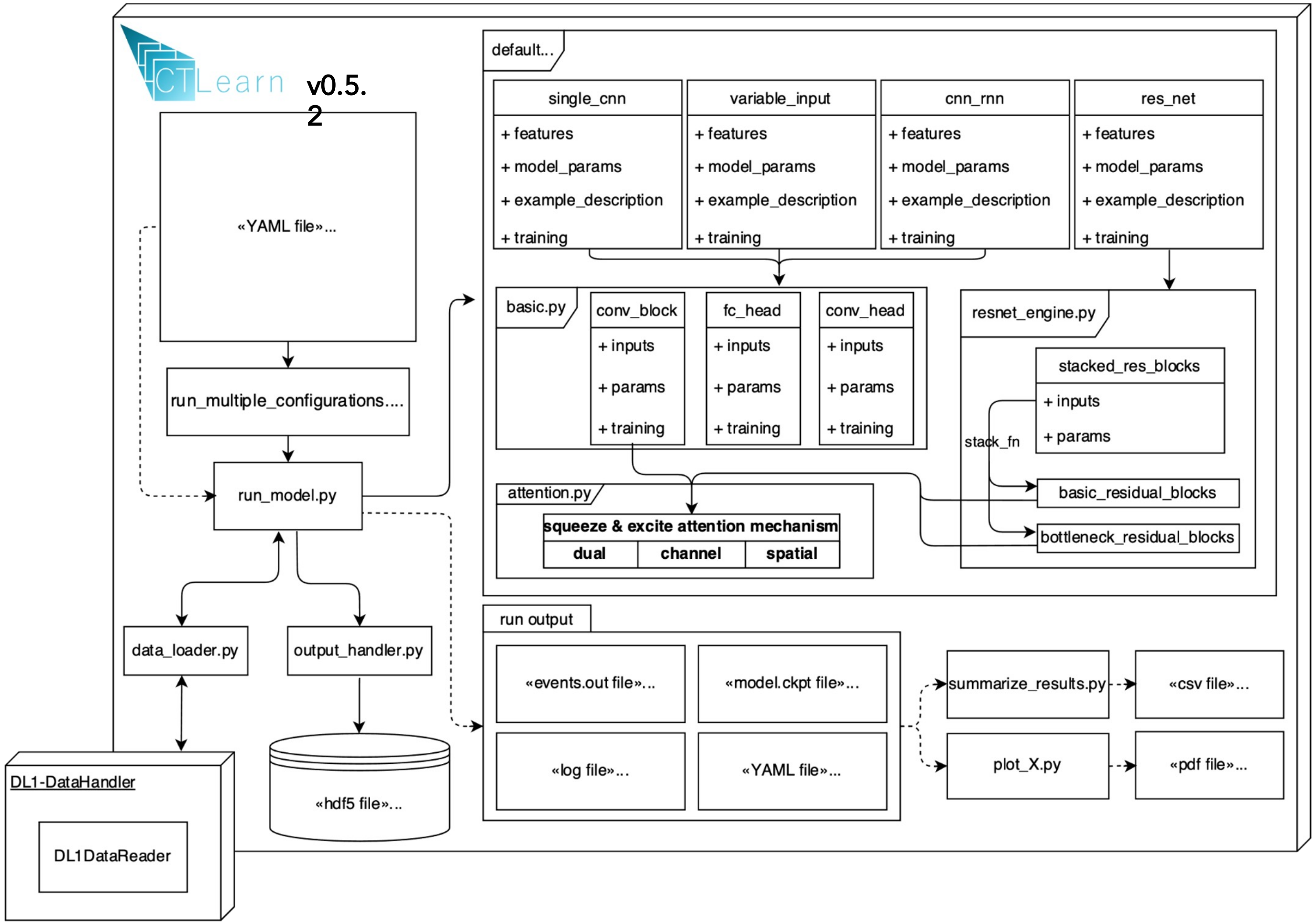
Optimized hyperparameters seem to be telescope-type dependent

CTLearn Optimizer: some results

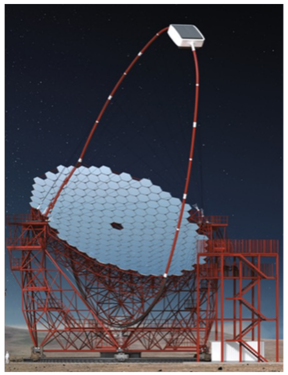
- Single_tel & TPE search: transfer to CNN-RNN

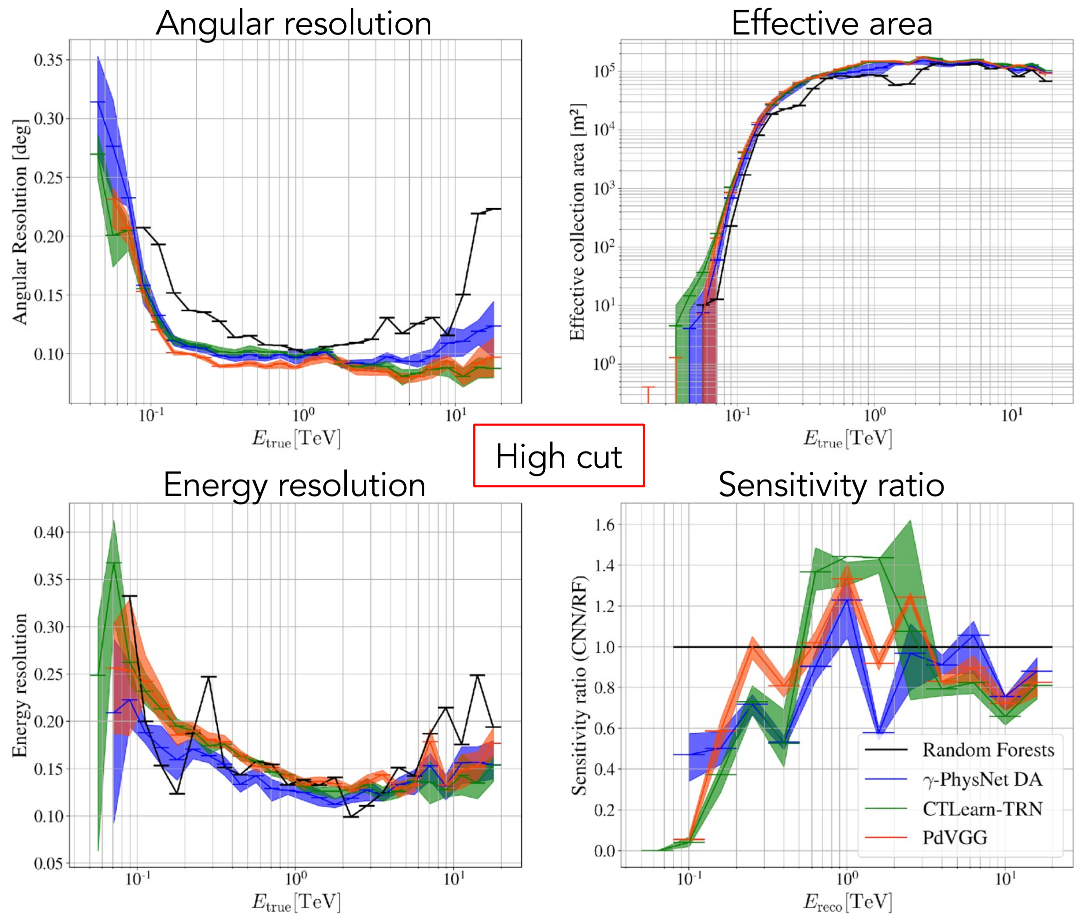
Hyperparameters	Telescope Type	Validation Accuracy	Validation AUC	Training Time
Base	LST	73.43%	0.8285	0h 41m 22s
Optimized	LST	74.96%	0.8422	0h 46m 53s
Base	SSTC	80.64%	0.9072	1h 51m 5s
Optimized	SSTC	83.49%	0.9217	3h 31m 43s
Base	MSTN	83.10%	0.9169	2h 15m 52s
Optimized	MSTN	84.20%	0.9313	6h 43m 14s

Telescope Type	Metric	Improvement
LST	Validation Accuracy	1.53%
LST	Validation AUC	1.37%
SSTC	Validation Accuracy	2.85%
SSTC	Validation AUC	1.45%
MSTN	Validation Accuracy	1.10%
MSTN	Validation AUC	1.44%


CTLearn v0.5.2
«YAML file»...
run_multiple_configurations....
run_model.py
data_loader.py
output_handler.py
DL1-DataHandler
DL1DataReader
«hdf5 file»...
default...
single_cnn
variable_input
cnn_rnn
res_net
+ features
+ model_params
+ example_description
+ training
basic.py
conv_block
fc_head
conv_head
+ inputs
+ params
+ training
resnet_engine.py
stacked_res_blocks
stack_fn
basic_residual_blocks
bottleneck_residual_blocks
attention.py
squeeze & excite attention mechanism
dual
channel
spatial
run output
«events.out file»...
«model.ckpt file»...
«log file»...
«YAML file»...
summarize_results.py
plot_X.py
«csv file»...
«pdf file»...

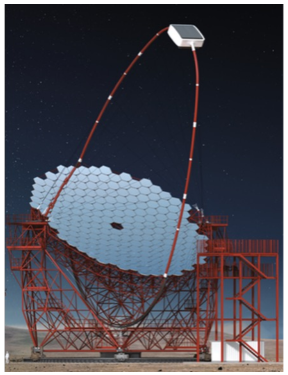
- Crosschecking three different implementations
- Same datasets, same cuts
- Different models
- Comparison against standard analysis (RF)

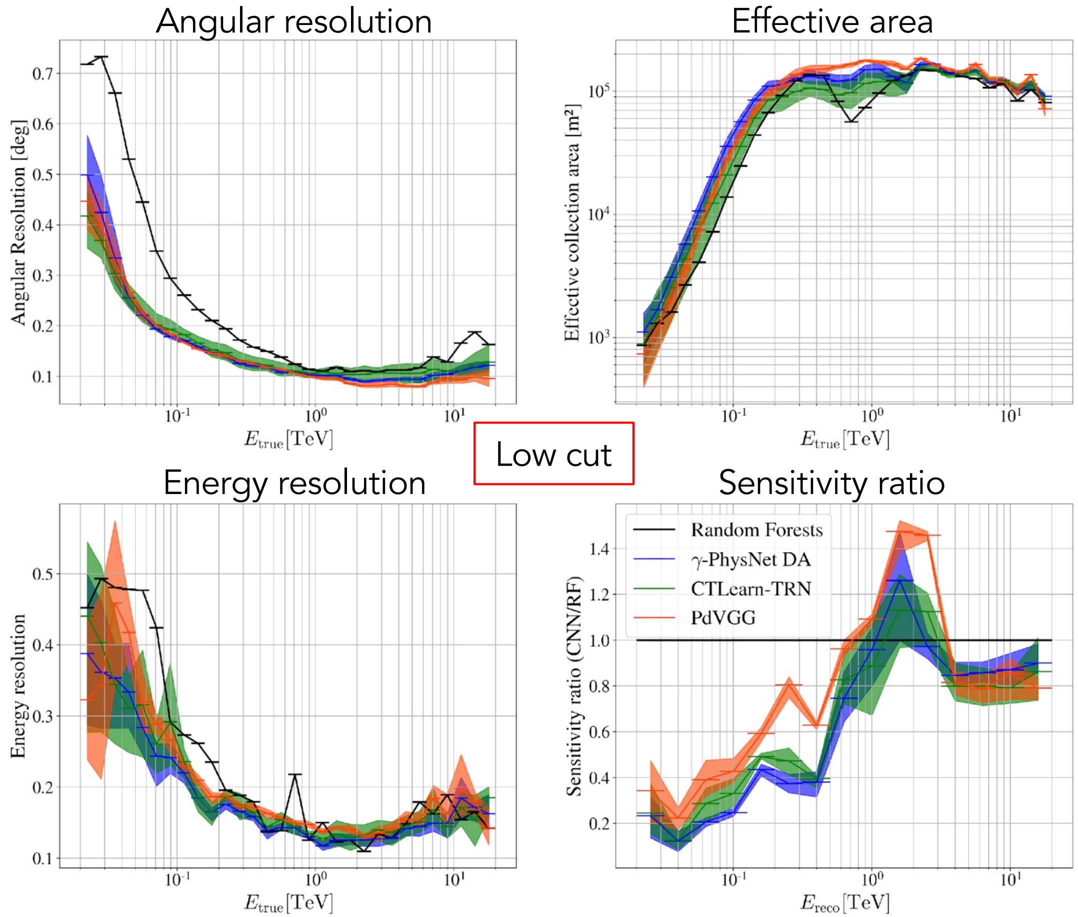




P. Grespan et al. PoS(ICRC2021) 771

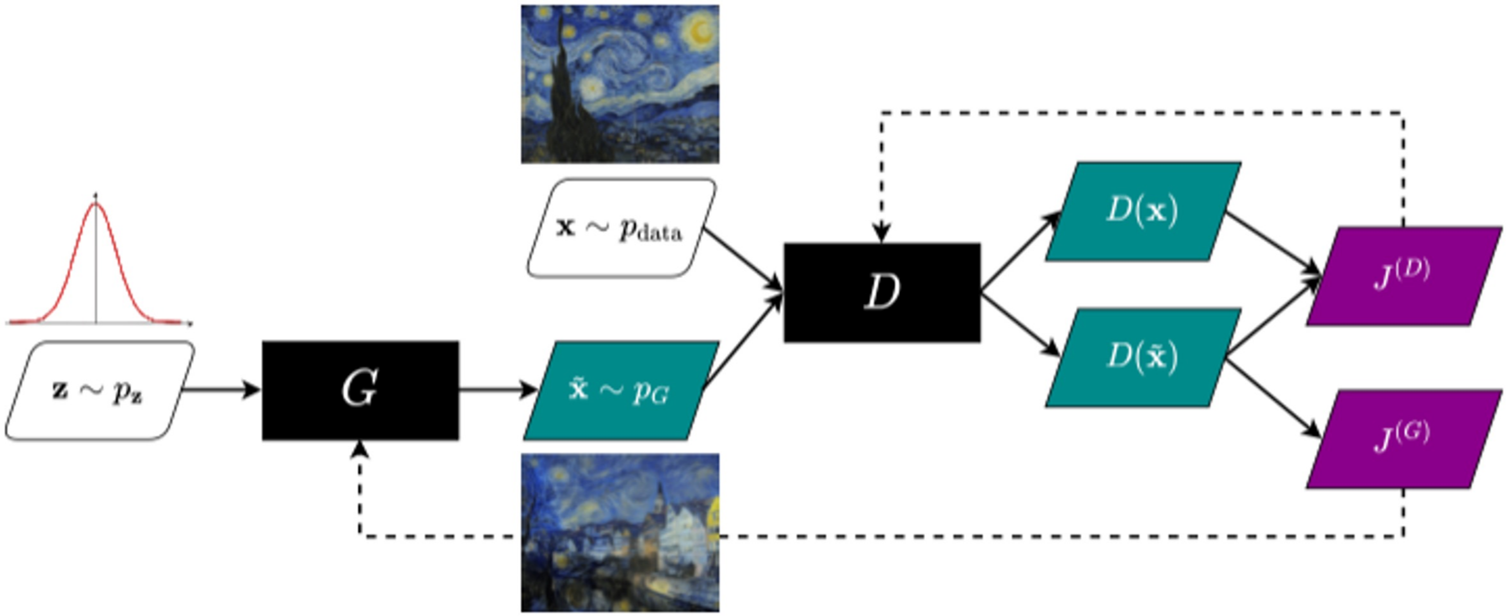
- Crosschecking three different implementations
- Same datasets, same cuts
- Different models
- Comparison against standard analysis (RF)





P. Grespan et al. PoS(ICRC2021) 771

- Generative adversarial networks (GANs)

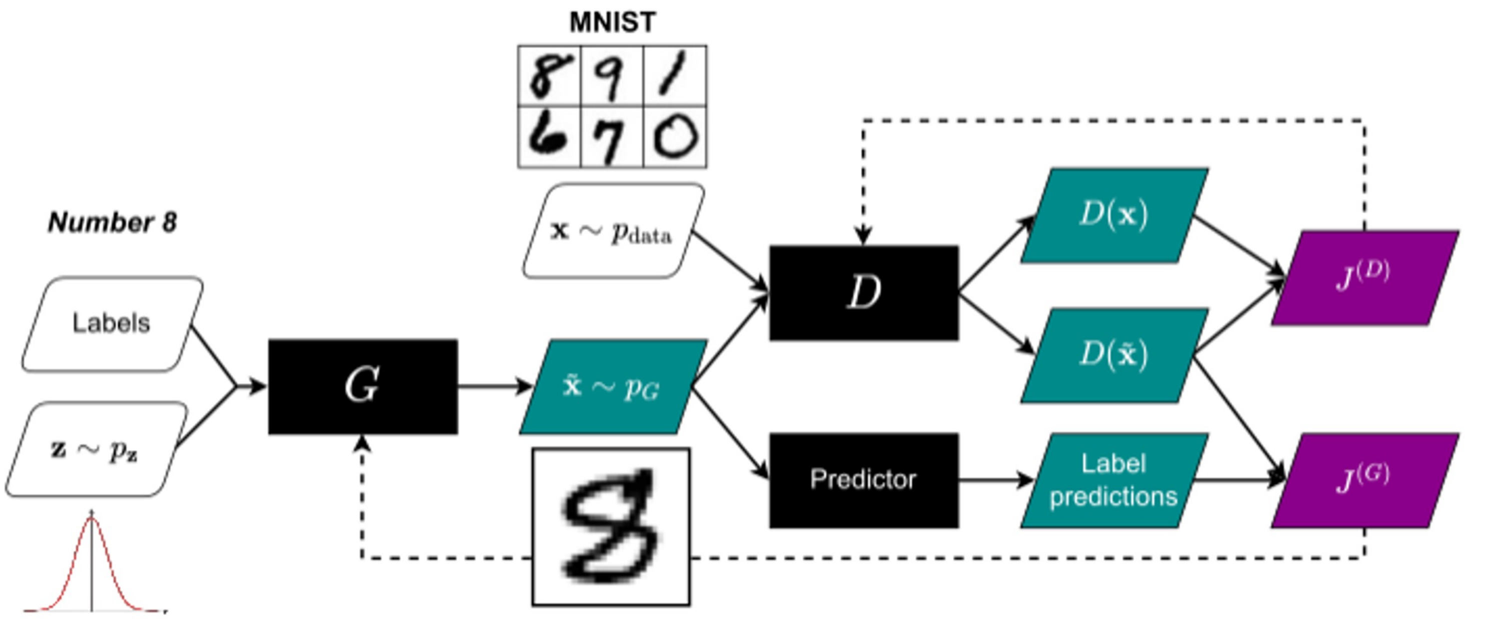


$$J^{(D)} = \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}}[D(G(\mathbf{z}))] - \mathbb{E}_{\mathbf{x}\sim p_{\text{datos}}}[D(\mathbf{x})]$$

$$J^{(G)} = -J^{(D)} = \mathbb{E}_{\mathbf{x}\sim p_{\text{datos}}}[D(\mathbf{x})] - \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}}[D(G(\mathbf{z}))] \rightarrow -\mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}}[D(G(\mathbf{z}))]$$

S. García-Heredia et al.

- Auxiliary conditional generative adversarial networks (AC-GANs)

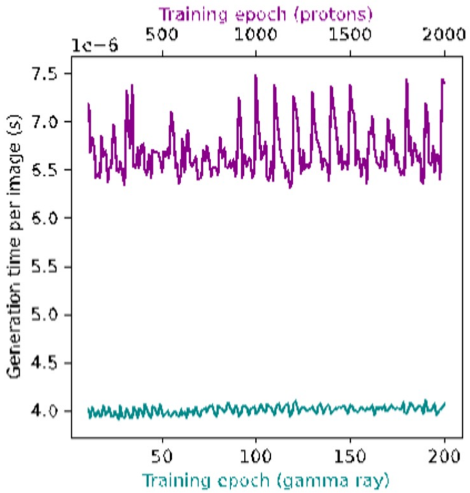


S. García-Heredia et al.

- Generation time

GANS

Using a GPU Nvidia 3090:

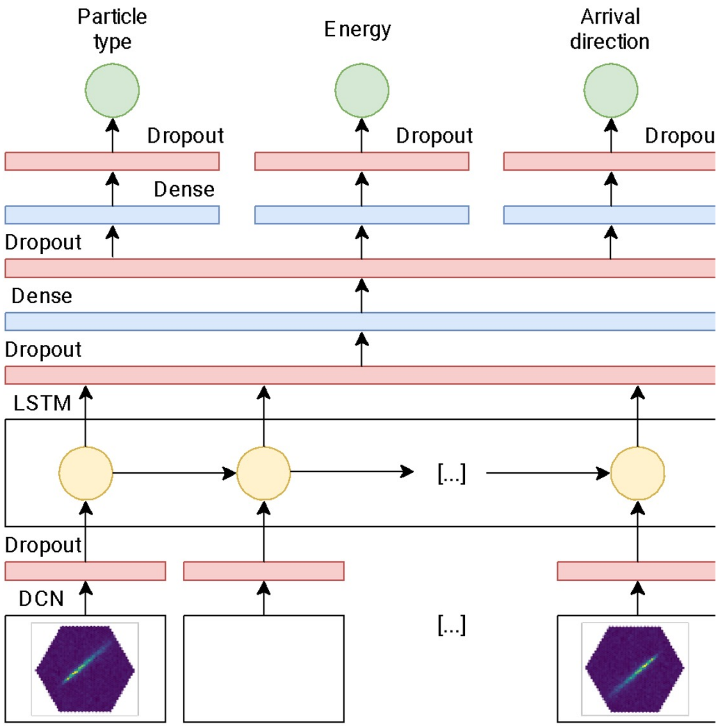


SIMULATIONS

- □ 1 s/ event
- Each event consists of one image for each detector
- Depends on what is being simulated and the computational capacity.

S. García-Heredia et al.

- Multi-task learning



- Tackling the real-data problem

Using GANs to bridge the gap between performances on simulations and observations

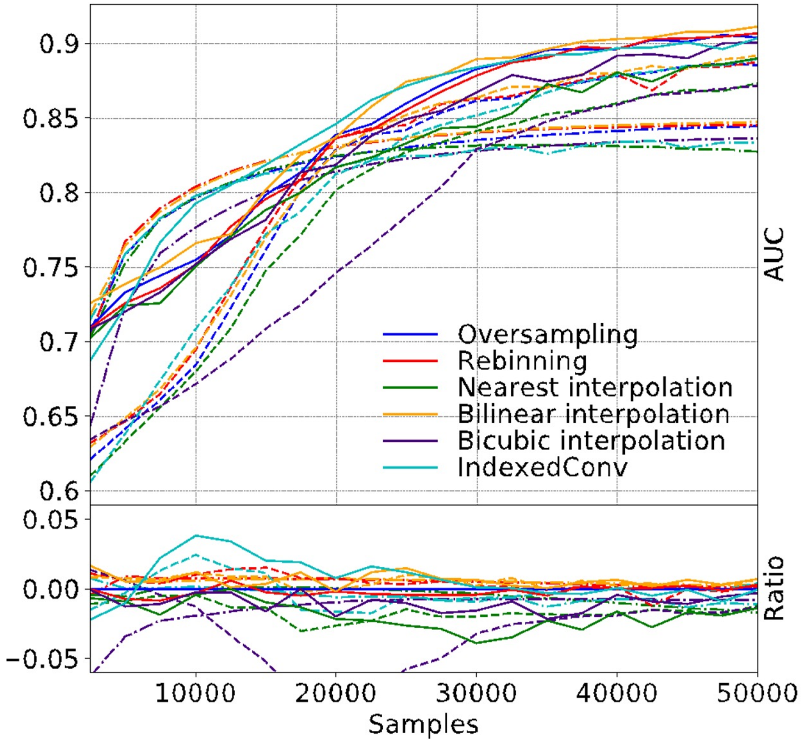
- Model optimization

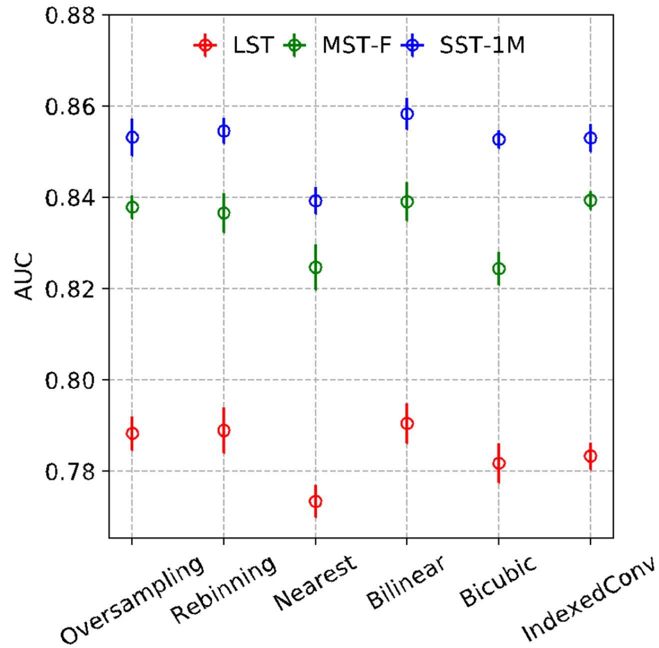
Combine heterogeneous cameras in one model
Implement and test deeper models
Enable optimization on large GPU clusters

- Invert models to explore pseudo-simulators

- …

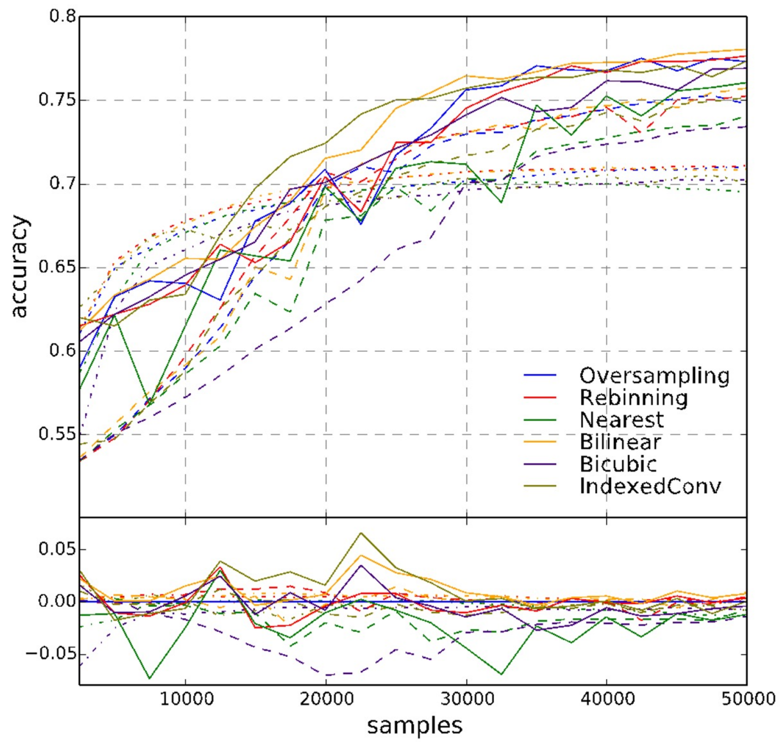
- Event classification task (AUC)

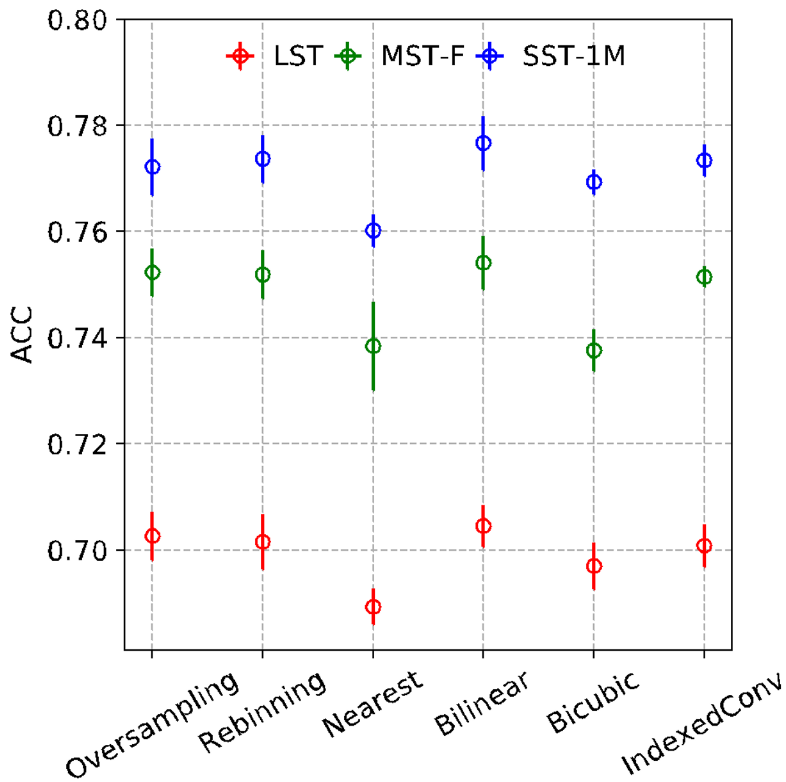




https://arxiv.org/abs/1912.09898

- Event classification task (ACC)





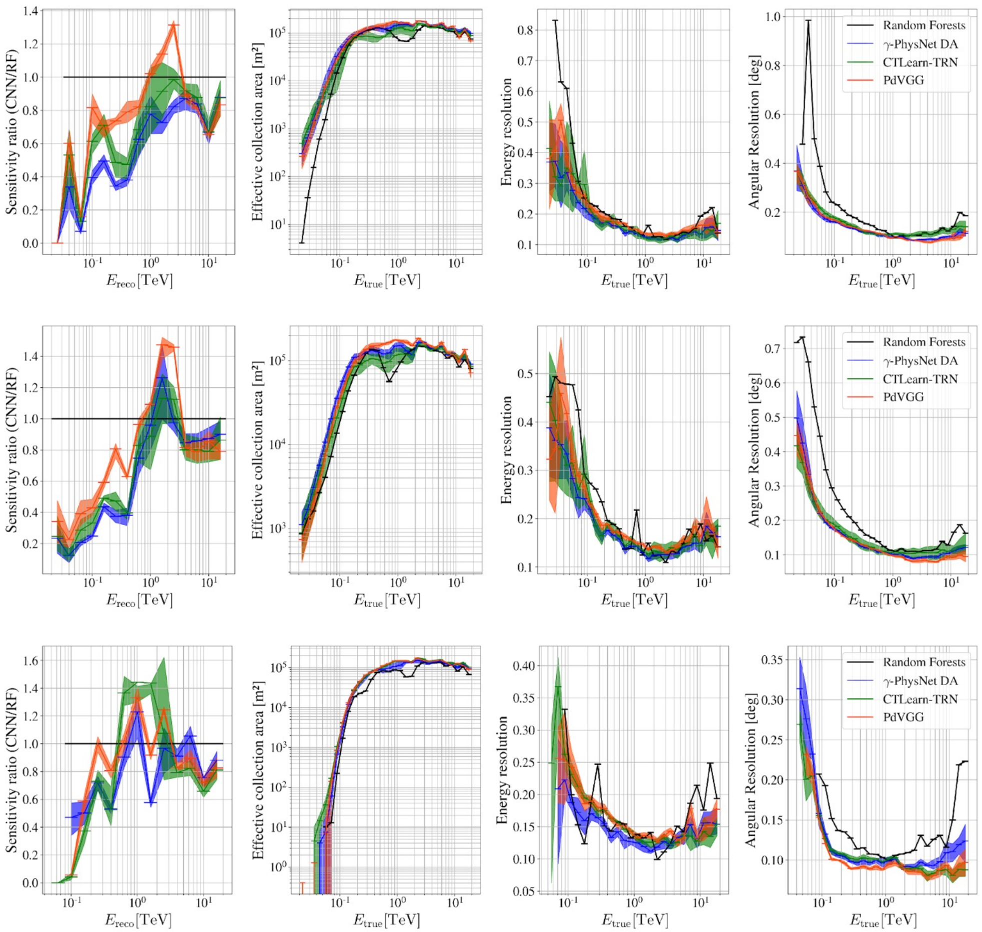
https://arxiv.org/abs/1912.09898

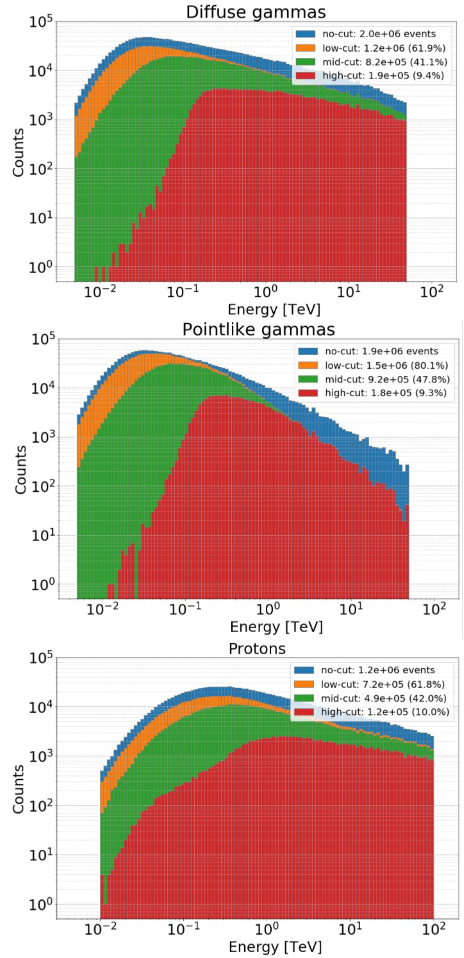
CTLearn: crosschecking results

Low cuts

Mid cuts

High cuts





P. Grespan et al. PoS(ICRC2021) 771