CTAO
Cherenkov Telescope Array Observatory

Low-Level Analysis

CTAO Summer School 2025

2025-06-23

Maximilian Linhoff, DPPS Lead Developer
maximilian.linhoff@cta-observatory.org

This presentation contains videos not exported to the PDF

1. Introduction
2. From Raw Data to Images
3. Image Parametrization
4. Reconstruction
5. Instrument Response Functions

Introduction

Scope of this Lecture

- This lecture will give an overview of the data processing from DL0 to DL3
  - DL0: First level of data written to disk and long-term preserved in CTAO, mostly pixel-wise time series for each telescope
  - DL3: Lowest data level given out to the users of the observatory on a regular basis, reconstructed event lists of gamma-ray candidates and a mathematical description of the measurement process (IRF)

- Many steps already happen before DL0 to process the raw camera signals to a common data format for all telescopes, some examples of this lower-level processing are also presented
- This lecture will explain concepts, designs, algorithms etc. It's not a hands-on tutorial in any particular software
- All example data you see here are CTAO simulations or LST observed data processed with ctapipe https://gitlab.cta-observatory.org/mlinhoff/ctao-school-2025

The task is to "invert" everything that happened to the gamma rays in the atmosphere and our detectors.

Data Levels up to DL2

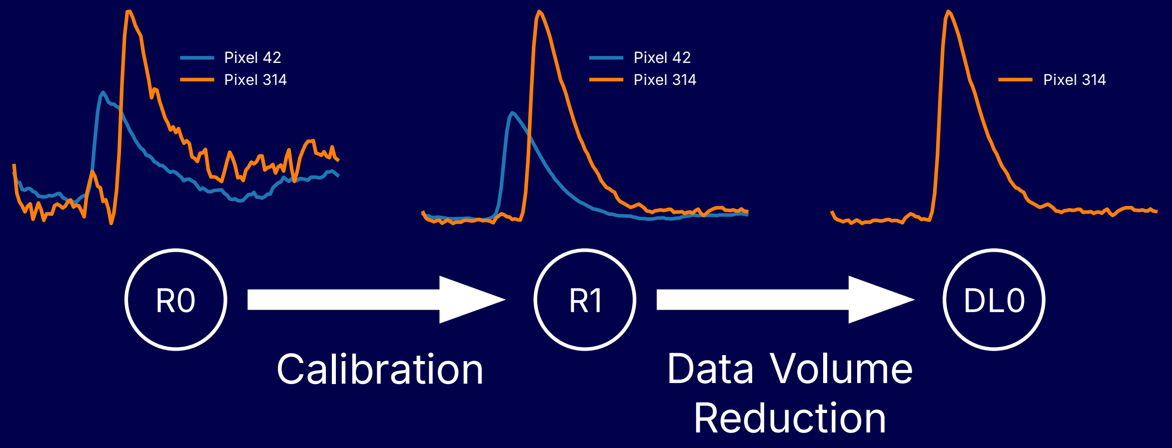


Image Extraction

DL1a

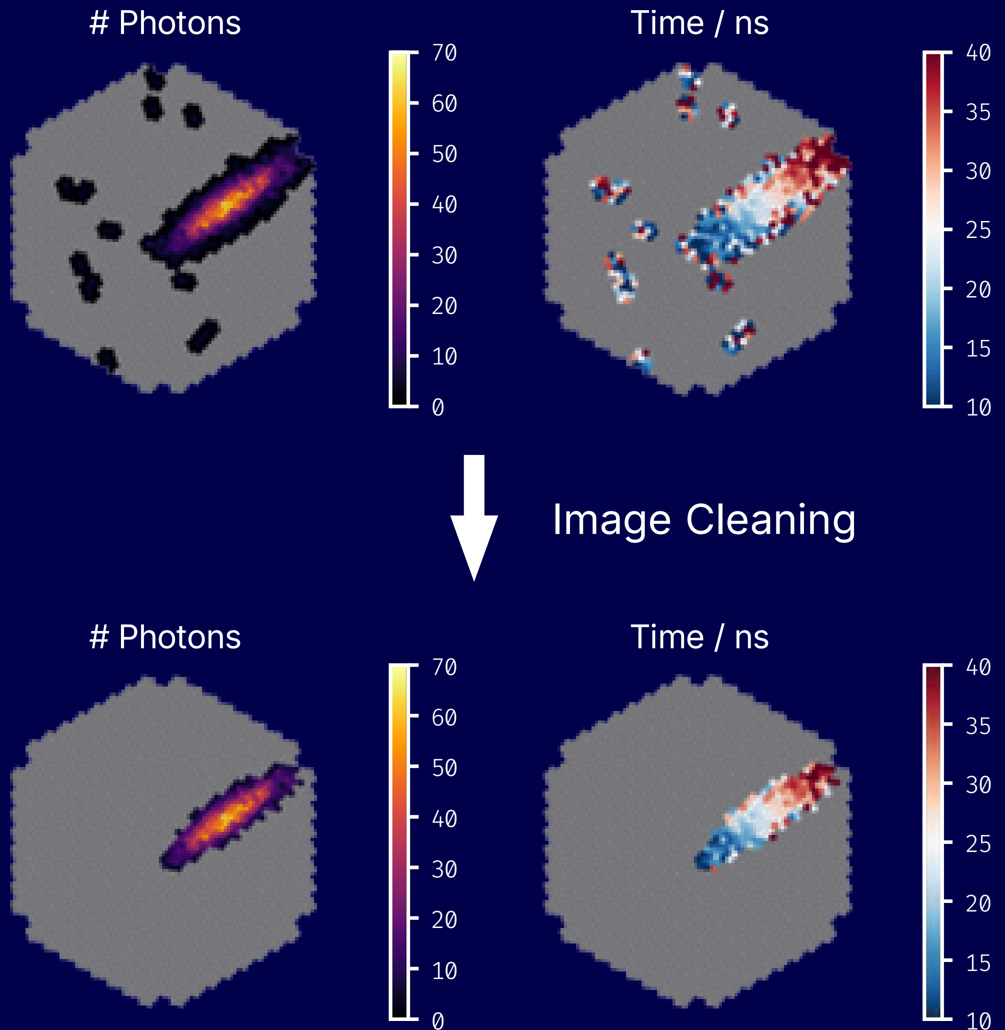


Image Parametrization

DL1b

Reconstruction

DL2

event_id	energy GeV	gammaness	ra deg	dec deg	time mjd
1	1500	0.82	83.6	22.1	59024.63123
2	400	0.73	83.5	21.9	59024.64183
5	680	0.92	83.7	22.0	59024.67093

event_id	tel_id	hillas_width	hillas_length	…	intensity
1	1	0.15	0.52	…	1253.1
1	2	0.07	0.15	…	323.3
1	3	0.23	0.42	…	2012.1
2	1	0.03	0.11	…	321.3
2	4	0.05	0.12	…	263.3
5	2	0.09	0.15	…	512.7
5	3	0.08	0.19	…	637.4

From Raw Data to Images

What does the Raw Data look like?

- PMT or SiPM signals are readout at high frequencies:
  LST: 1 GSample/s = 1 measurement per nanosecond
- A local trigger looks for “interesting” data
- Telescope triggers are sent to the array trigger, which decides if data should be stored or not
- In case of a trigger, the cameras digitize the pixel data in a window around the trigger time.
- Some cameras apply two different amplification gains to the analog signal to increase dynamic range (e. g. LST & NectarCam)

What does the Raw Data look like?

```
DL0TelescopeTrigger(
    tel_id=1
    trigger_id=1
    trigger_type=3
    trigger_time_s=1738704882
    trigger_time_qns=7648132
    readout_requested=False
    data_available=True
    hardware_stereo_trigger_mask=1)
```

What does the Raw Data look like?

```
DL0SubarrayEvent(
    event_id=1
    trigger_type=1
    sb_id=2000000066
    obs_id=2000000200
    event_time_s=1738704882
    event_time_qns=8420956
    trigger_ids=array([2], dtype=uint64)
    tel_ids=array([1], dtype=uint16))
```

What does the Raw Data look like?

```
DL0TelescopeEvent(
    event_id=1
    tel_id=1
    event_type=32
    event_time_s=1738704882
    event_time_qns=8420956
    pixel_status=array([5, 5, ..., 5, 5], shape=(1834,), dtype=uint8)
    first_cell_id=array([2260,    0, ...,    0,    0], shape=(2096,), dtype=uint16)
    num_channels=1
    num_samples=36
    num_pixels_survived=1834
    calibration_monitoring_id=0
    waveform=array([  0,   0, ...,   0, 480], shape=(66024,), dtype=uint16)
    pedestal_intensity=array([], dtype=float64)
    num_modules=262,
)
```

Raw Data Calibration

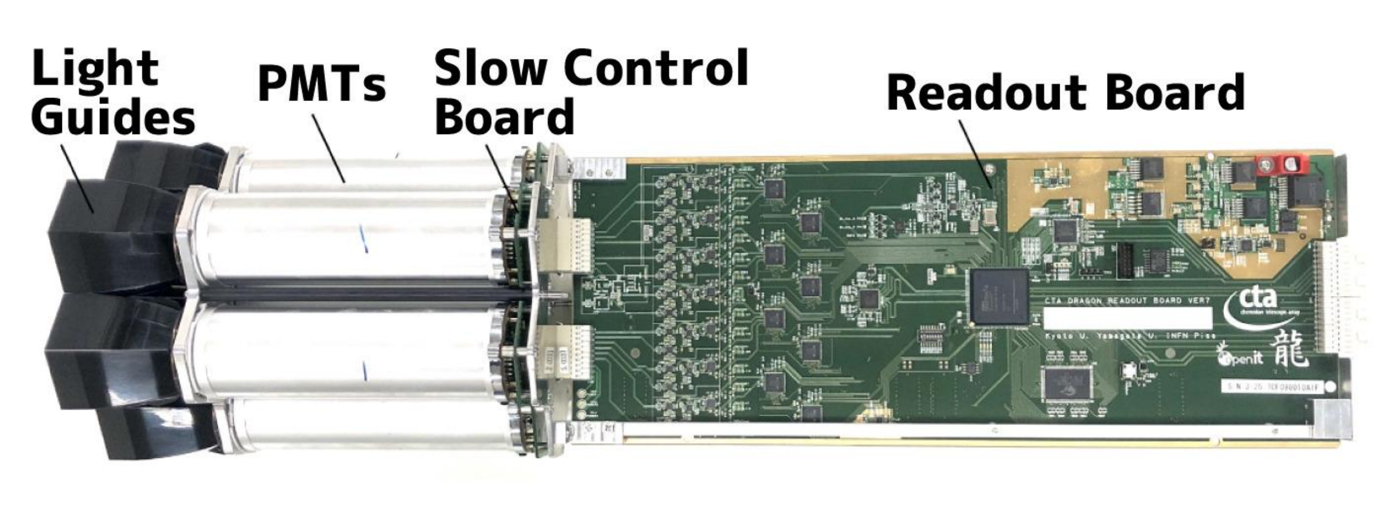
- The cameras are delivering pixel data in ADC counts
- Camera servers perform a first low-level calibration to amplitudes in photoelectrons
- Cameras are calibrated using different kinds of calibration events

Dark Pedestal	Randomly triggered event without HV switched on, closed shutter
Single-PE	Randomly triggered event with HV switched on, closed shutter
Sky Pedestal	Randomly triggered event with HV switched on, open shutter
Flatfield	Calibration-box laser in the dish illuminates the camera with a light pulse

- During observations, sky pedestal and flatfield events are taken at regular intervals, “interleaved” with the physics trigger events

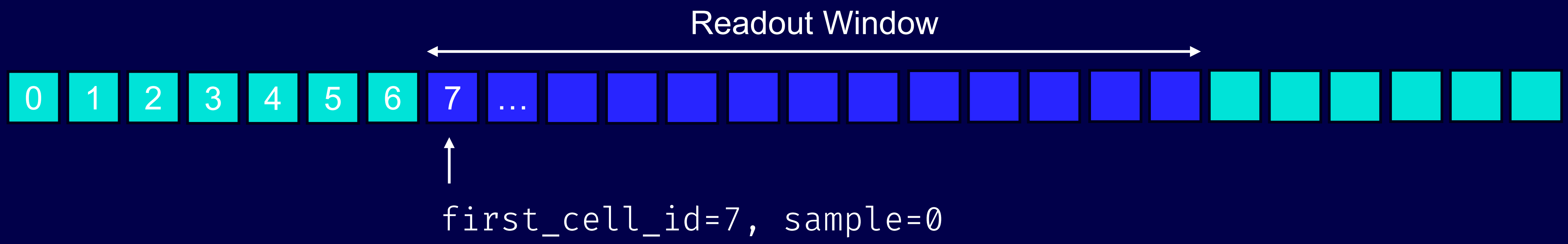
LST DRS4 Calibrations

- The LST Camera uses DRS4 chips to digitize the pixel data
- This chip adds several "artifacts" to the data, which need to be corrected
- Needs several tables of calibration coefficients
- Computed from dark pedestal events
- Are updated regularly and immediately after hardware changes



doi:10.1016/j.nima.2025.170229

LST DRS4 Calibrations



- A DRS4 chip as 8 channels of 1024 capacitor cells
- LST cascades 4 channels to have a larger time buffer
  → 1 pixel/gain combination uses 4096 cells
- A window of 40 samples is read out around the trigger time
- Position of the window on the buffer is recorded as first_cell_id

LST DRS4 Calibrations

Five different effects need to be calibrated:

- Cells affected by strong noise
  - The first three and the last sample in the readout window show increased electronic noise
  - No efficient way to correct ⇒ just discard these samples

- Baseline Correction
  - each cell as an individual value for “no signal”
  - $2 \cdot 1855 \cdot 4096$ calibration coefficients

- Spike Subtraction
  - For certain predictable samples, the amplitude is increased for 3 consecutive cells.
  - Mean height of these spikes needs to be subtracted. $2 \cdot 1855 \cdot 3$ coefficients.

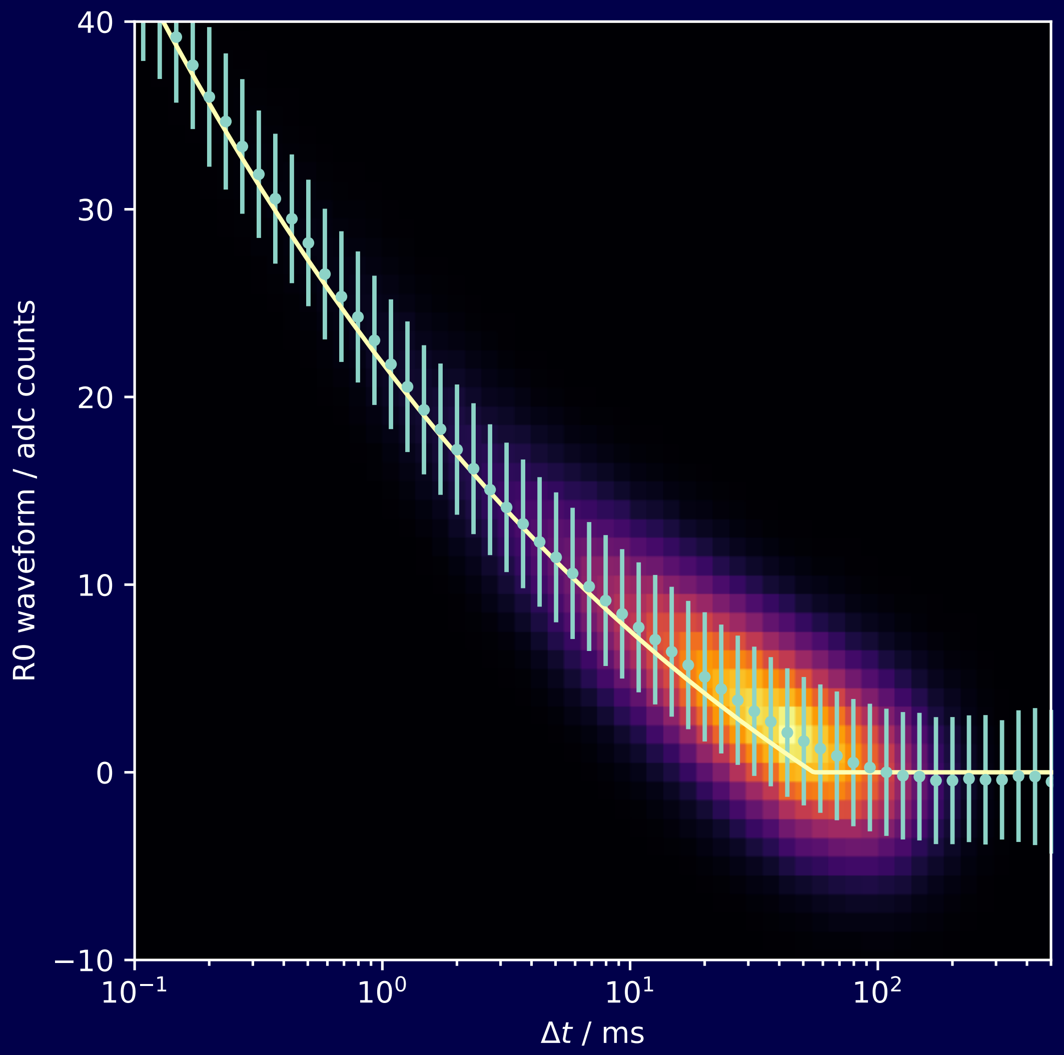
LST DRS4 Calibrations

Five different effects need to be calibrated:

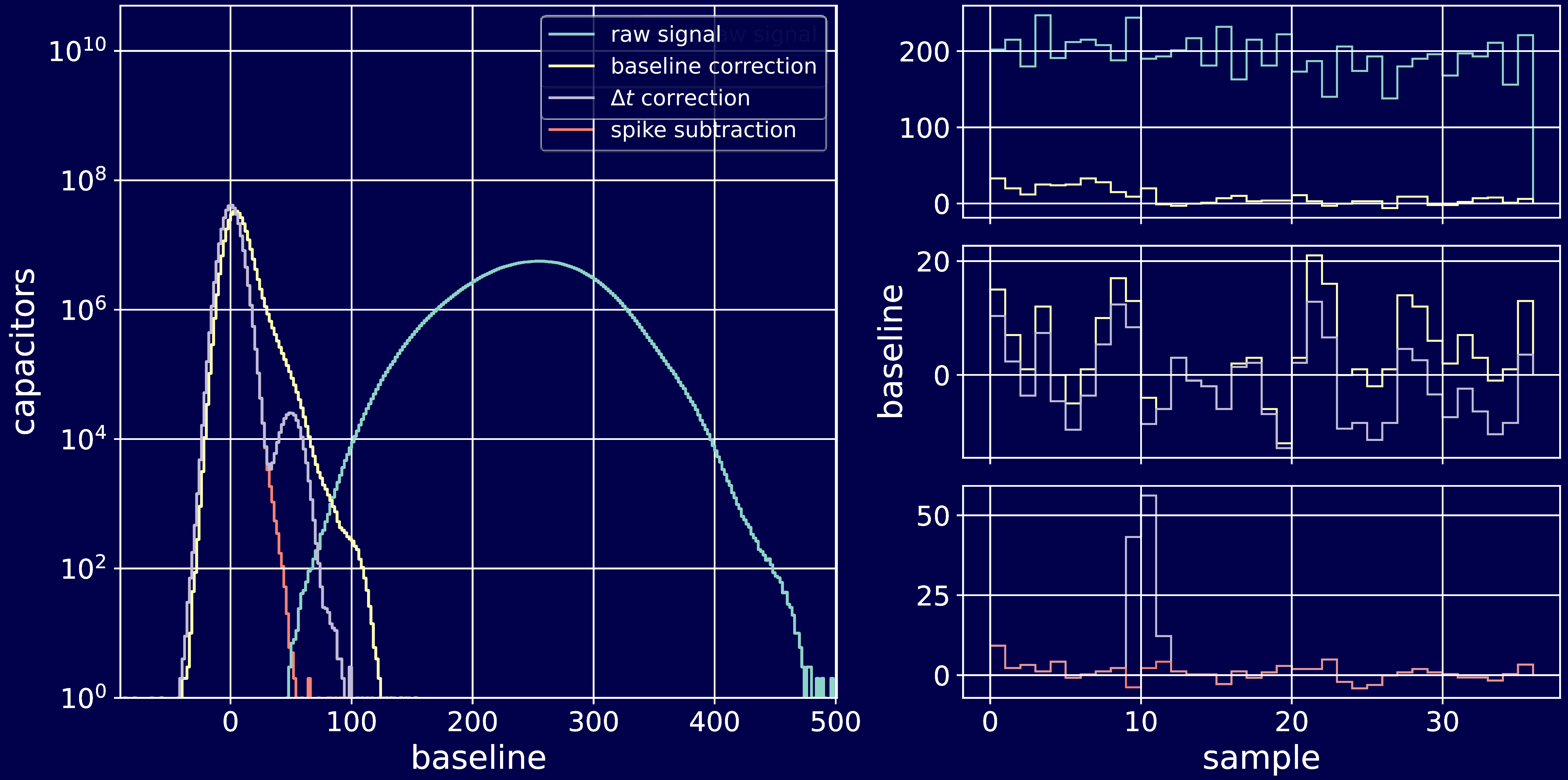
- DT-Correction
  - The amplitude of the cell also depends on the time since it was last read out
  - Same formula can be used for the same kind of DRS4 chip:

$$f(\Delta t) = \begin{cases} a \cdot \left(\frac{\Delta t}{t_0}\right)^b - a, & \Delta t < t_0 \\ 0, & \Delta t \geq t_0 \end{cases}$$

- Time Correction
  - The readout time of each cell is not perfectly uniform → time shift
  - This time shift is not corrected at the waveform level but later in the analysis



LST DRS4 Calibrations



Calibration to Photoelectrons

- Computed from flatfield and pedestal events
- Method for PMT cameras like LST & NectarCam:

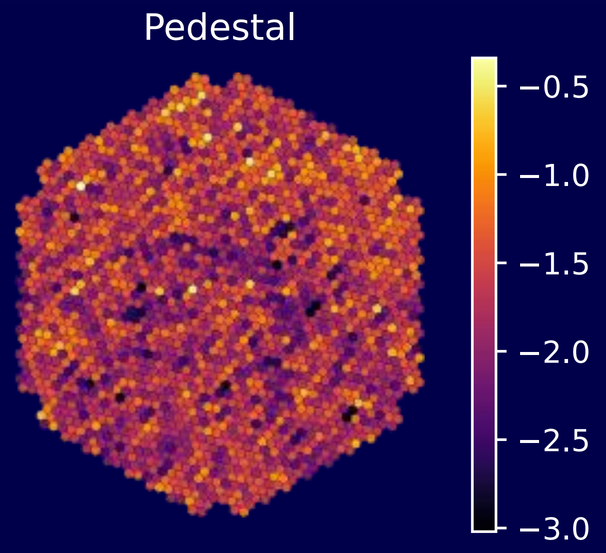
$$N_{p.e.} = \frac{(\bar{Q} - \bar{P})^2}{\sigma_Q^2 - \sigma_P^2} \cdot F^2 \Rightarrow C = \frac{\overline{N_{p.e.}}}{\bar{Q} - \bar{P}}$$

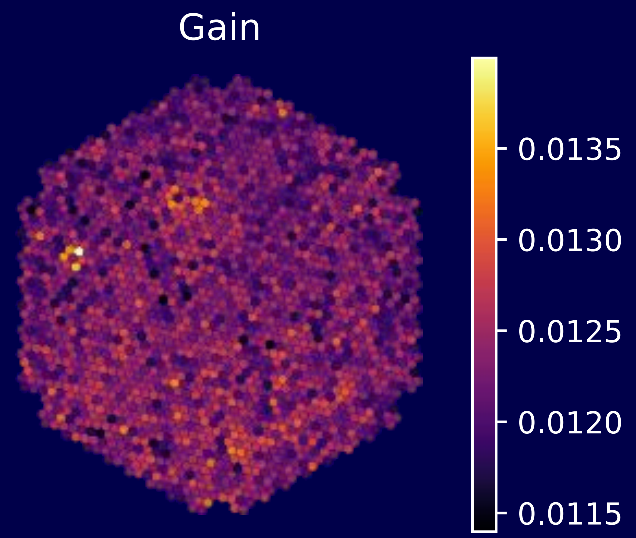
- Calibration factors also need be corrected for an additional noise term dependent on signal strength
- Needs flat field events with different amplitudes → Filter Wheel Scans
- Then simply applied as

$$S_i = (R_i - P_i) \cdot C_i$$

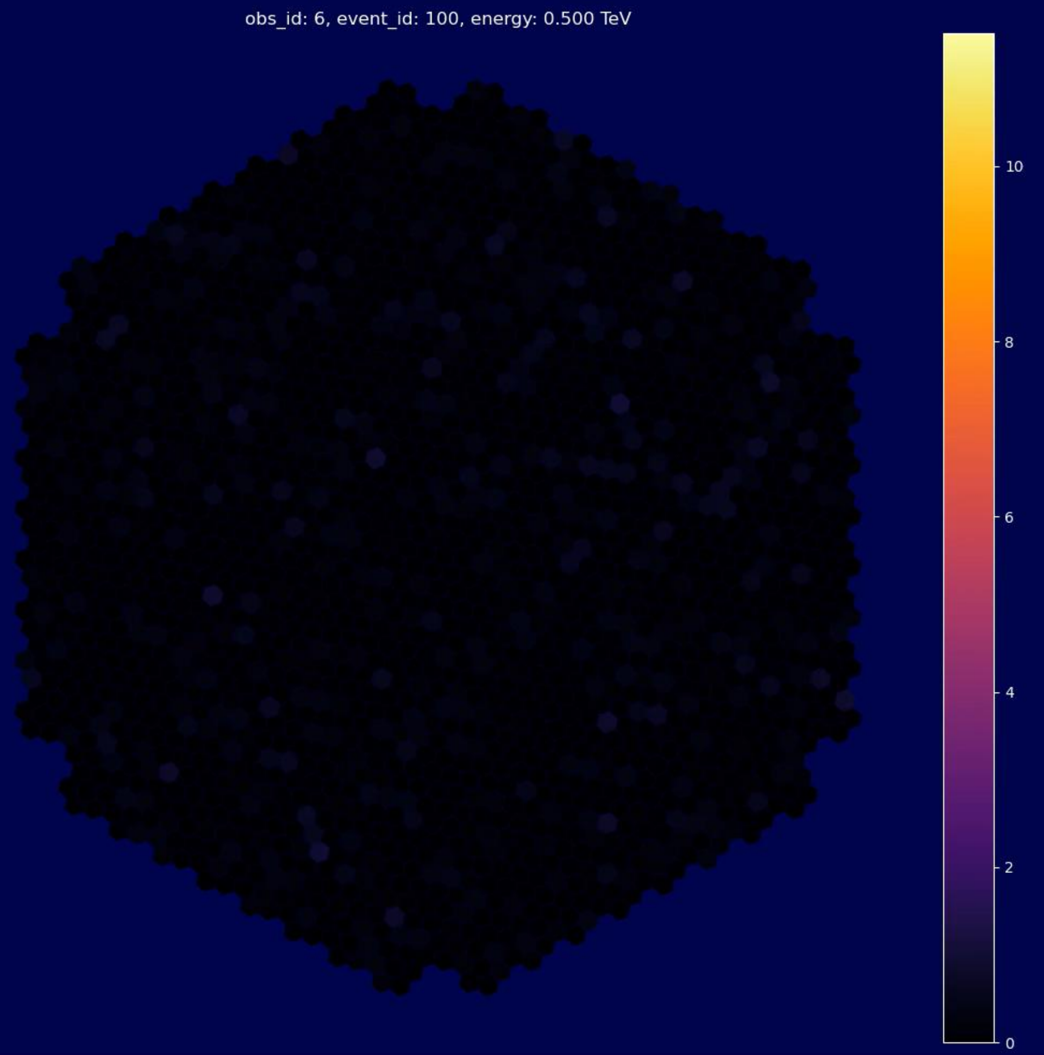
- Relative time shifts between pixels are also computed from flatfield events

“Camera Calibration of the CTA-LST Prototype” doi:10.22323/1.395.0720

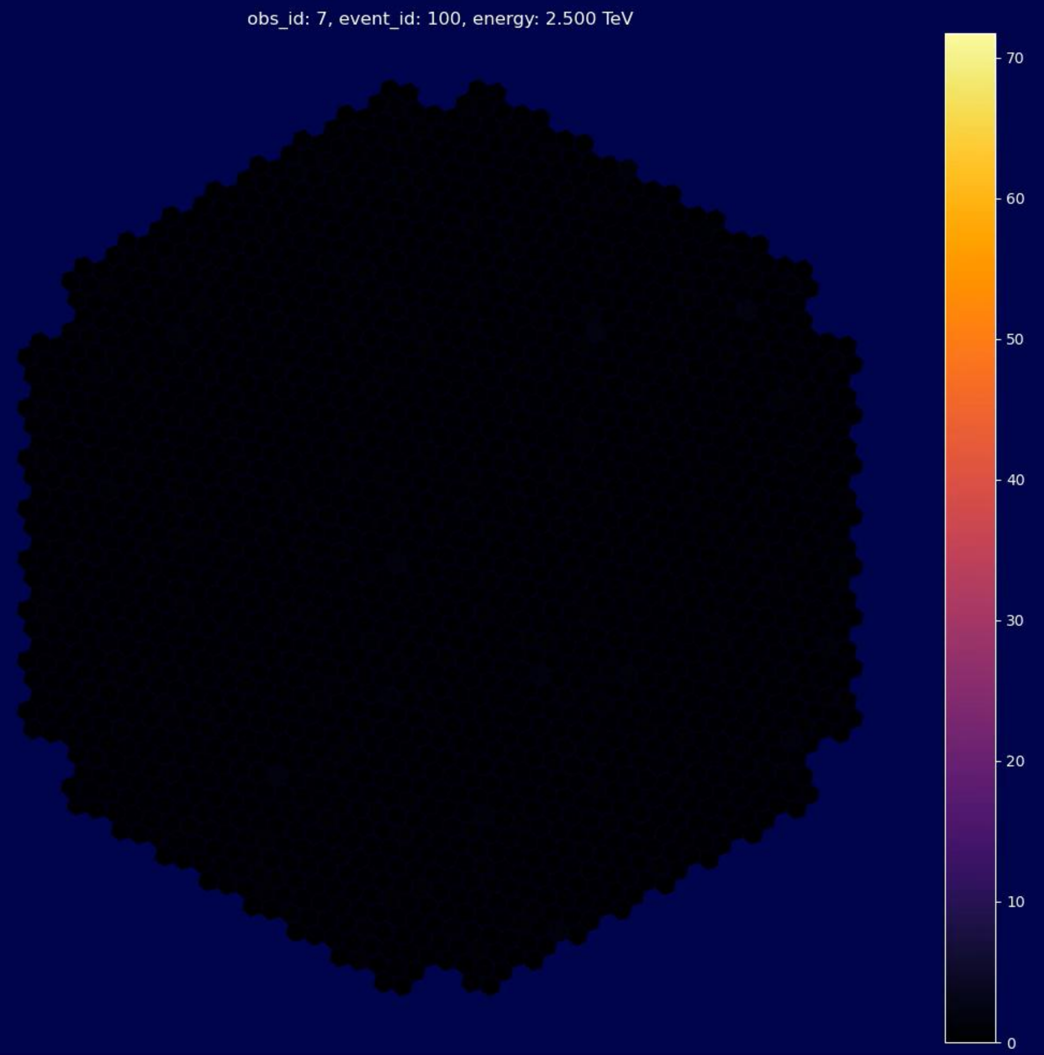




Gammas



Protons



obs_id: 6, event_id: 100, energy: 0.500 TeV

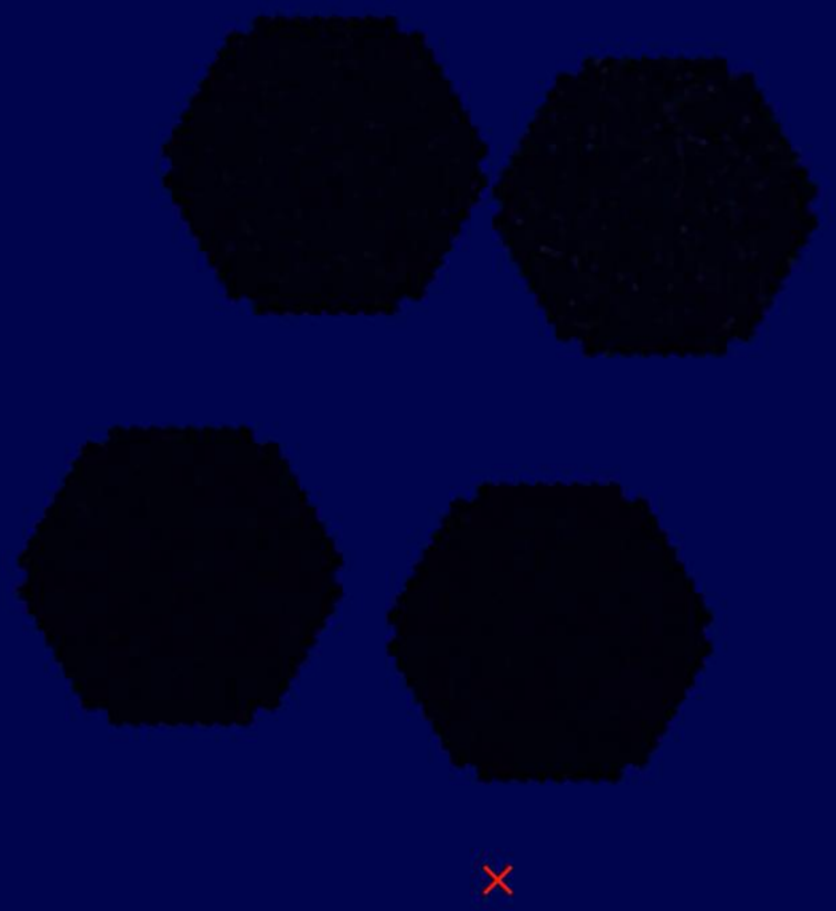
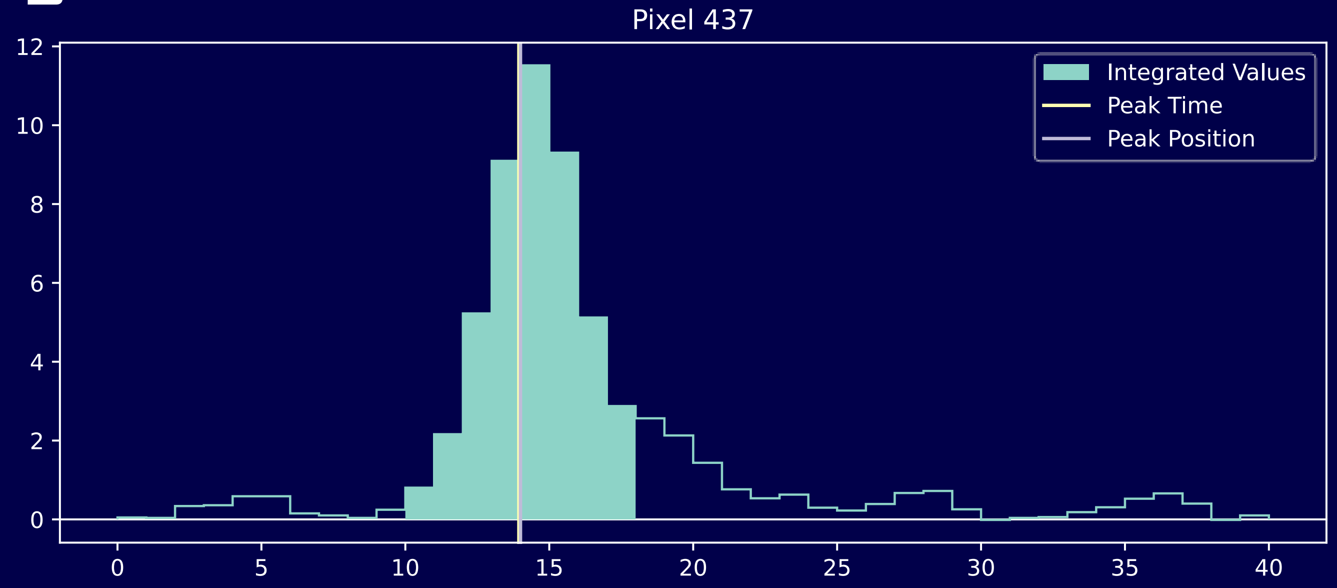


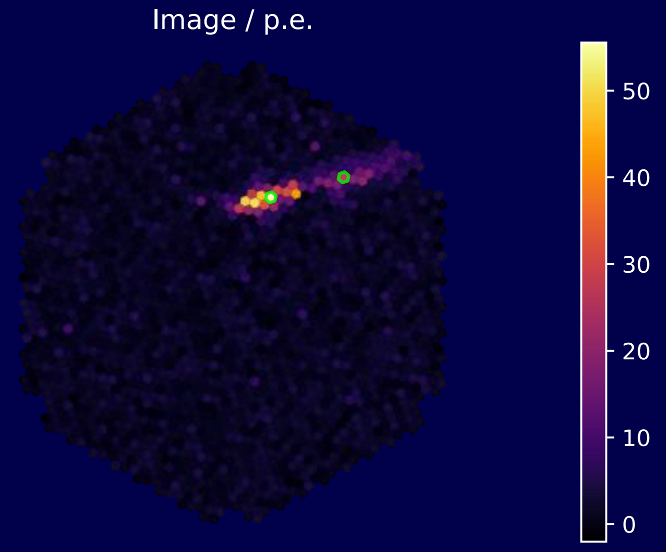
Image Extraction

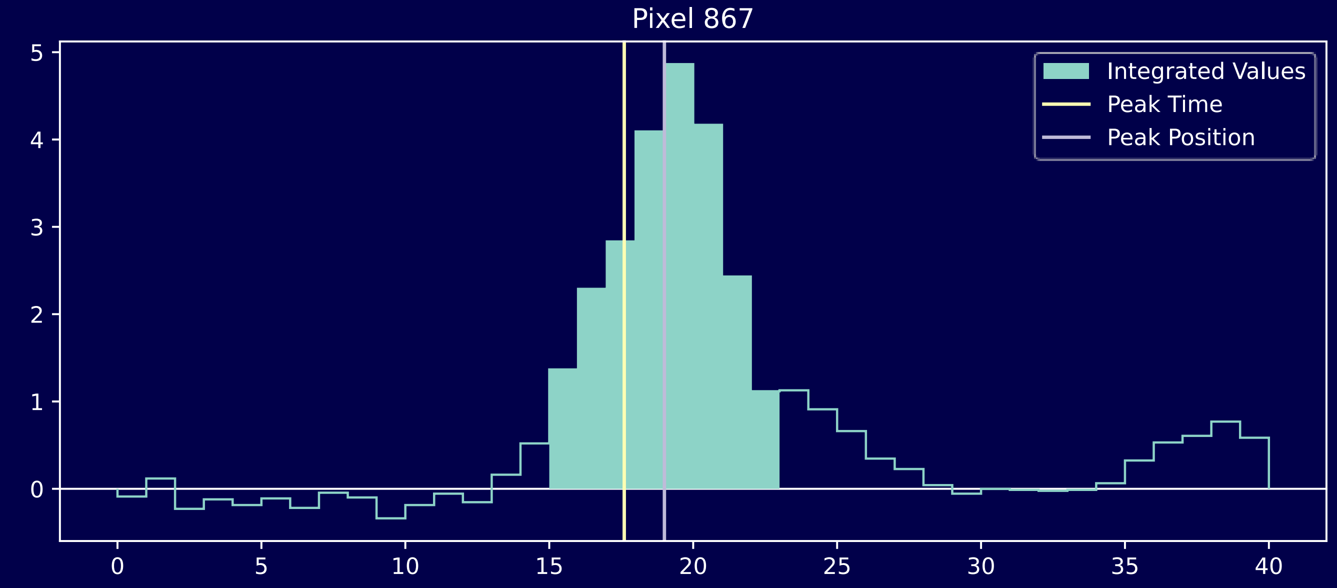
- Second large data reduction, reduce waveform to just two values for each pixel:
    - Number of photoelectrons
    - Average arrival time (peak time)

- Several algorithms available in ctapipe
- Most work by finding a peak, integrating around it and computing a weighted average for the peak time
- To minimize effect of noise, peak finding can combine waveforms of neighboring pixels or the whole camera

Image Extraction







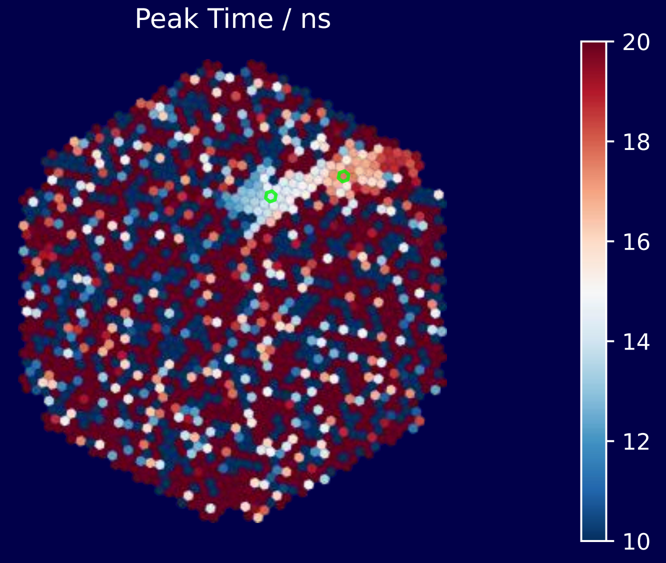


Image Extraction

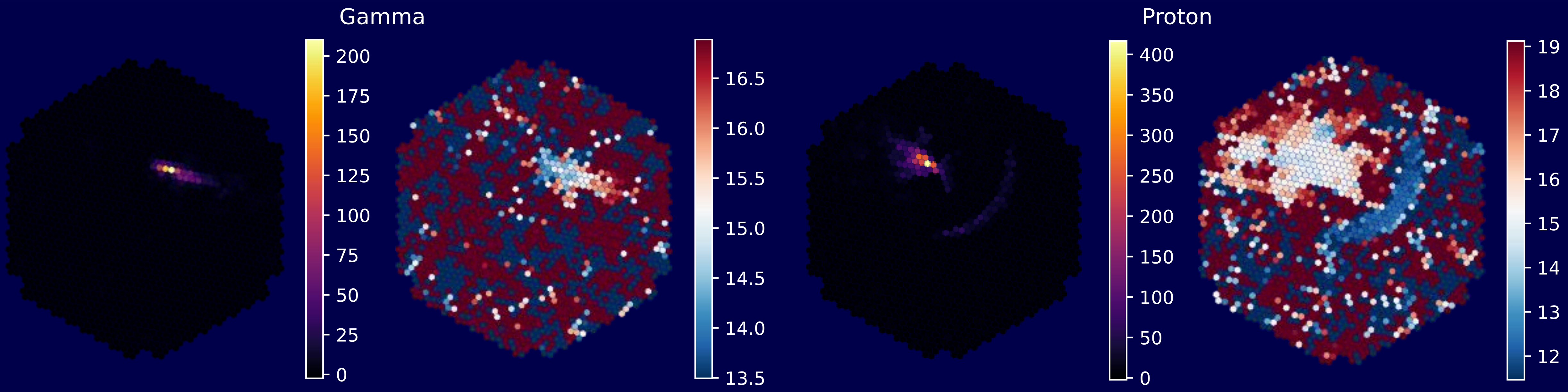


Image Cleaning

- Most pixels in any given event only contain noise
  ⇒ Identify pixels likely containing Cherenkov light
- Many different algorithms
- Can use intensity, peak time and noise level estimated from pedestal events
- Simple Example: tailcuts clean:
  1. Identify pixels above first threshold
  2. Remove pixels with less than 2 neighbors
  3. Add neighboring pixels above a second, lower threshold

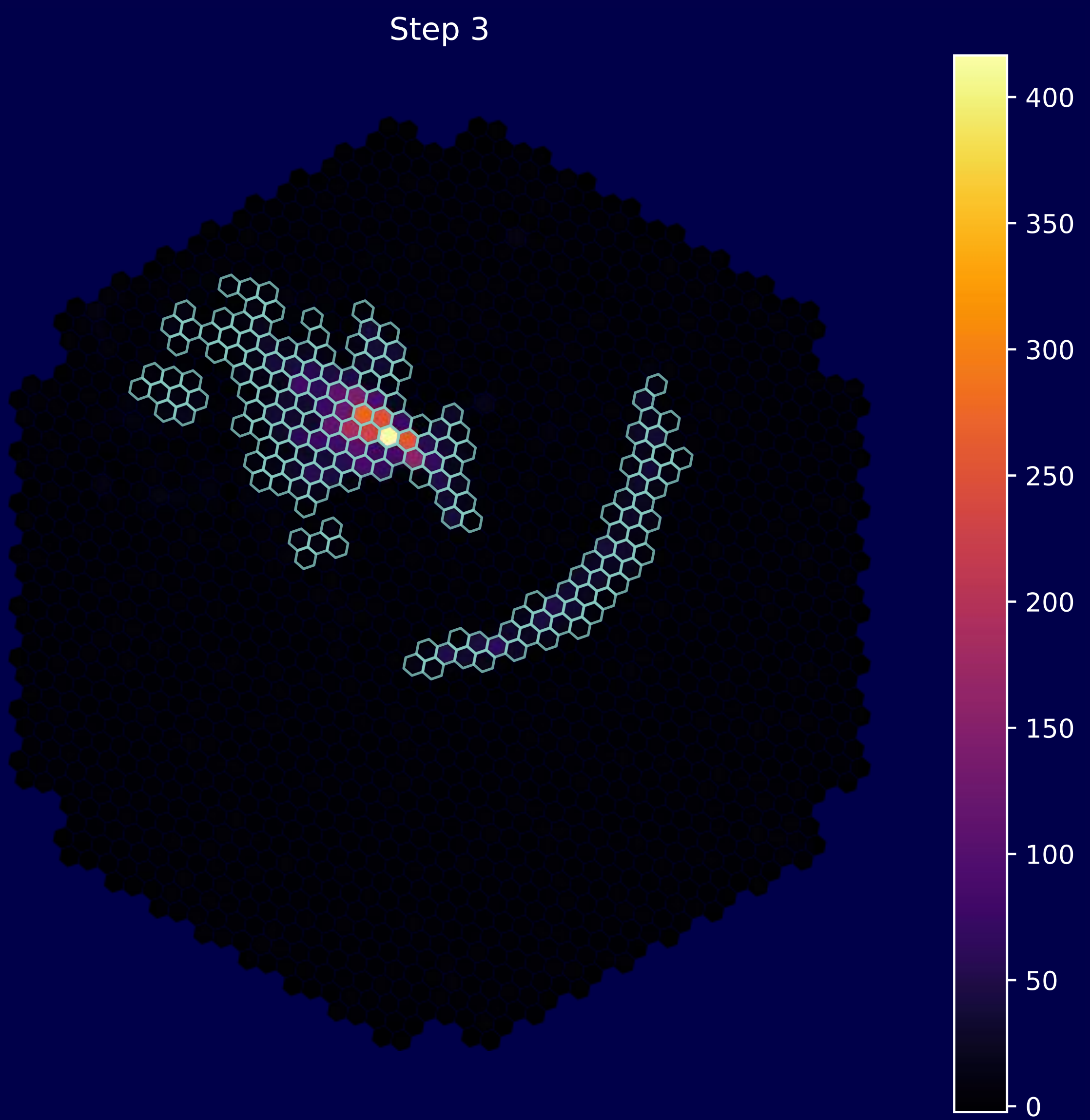


Image Parametrization

Image Parameters

- Next step of the data reduction
- From two values per pixel to tens of parameters per telescope event
- Goal: keep as much information from images in high-level parameters
- Will be used in reconstruction of energy, direction and primary particle type
- Initial set of parameters proposed by Hillas in 1985
- Extended over time

A. M. Hillas, *"Cerenkov Light Images of EAS Produced By Primary Gamma Rays and By Nuclei"*,
19th International Cosmic Ray Conference, 1985

3. Parameters used to describe Cerenkov images of showers

A typical simulated image is shown alongside in Figure 1. The figures give numbers of photoelectrons in each photomultiplier. (2 TeV gamma-ray from source in centre of field. Impact parameter 60m, zenith angle 30°.) The image axis (dashed line) is determined: this line minimises the signal-weighted sum of squares of perpendicular angular distance of the detectors. (As small distant noise signals can distort the small r.m.s. dimensions, *any individual signal below 1% of the total in all 37 tubes is ignored.*)

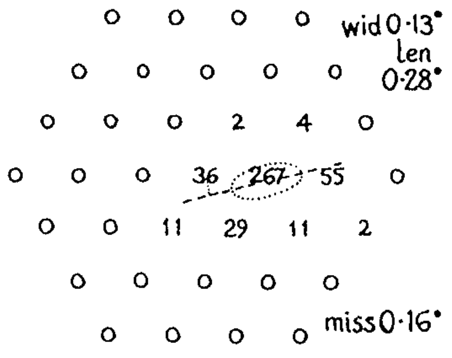
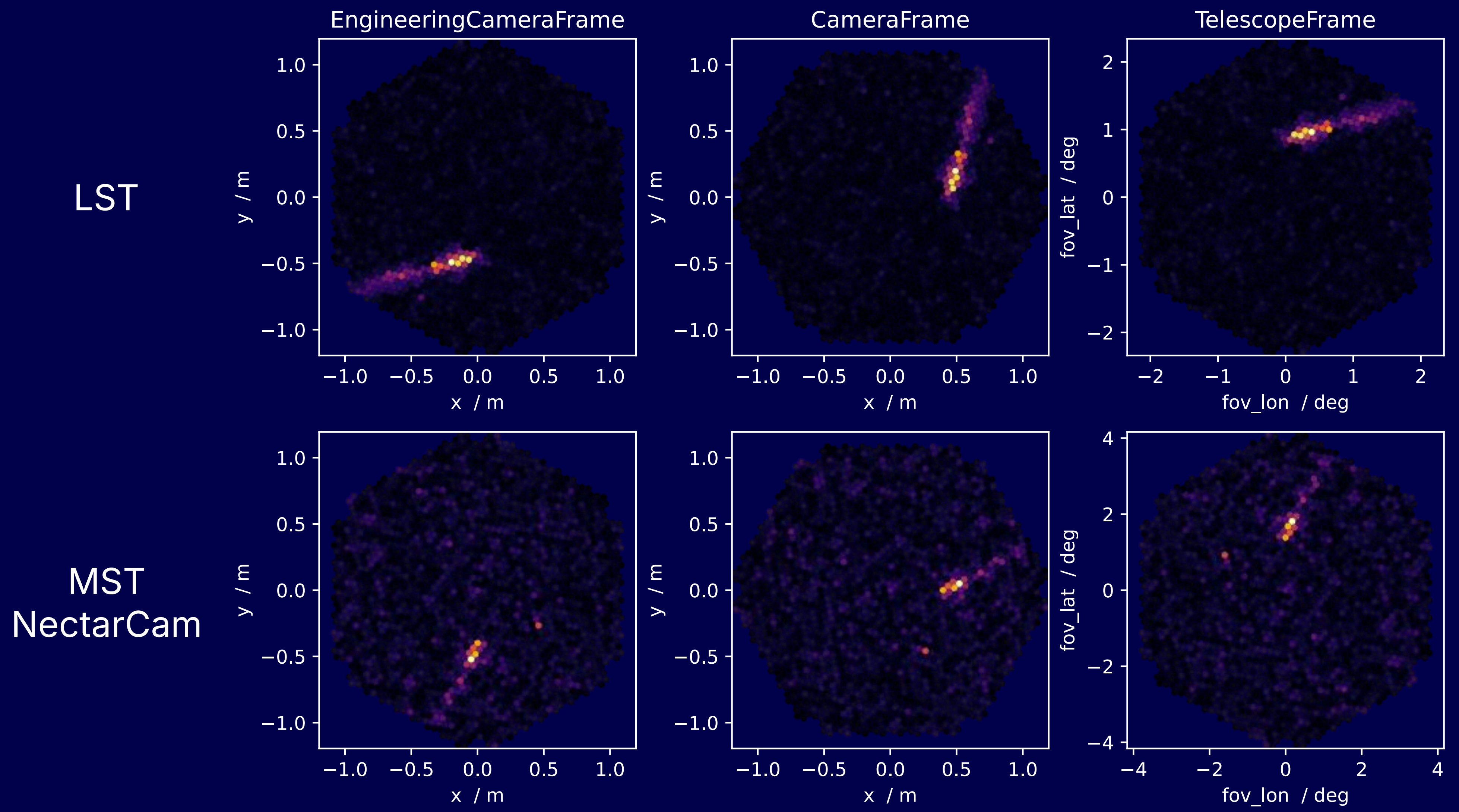


Figure 1.

The r.m.s. spread of light in directions parallel and perpendicular to this axis are referred to as the *LENGTH* and *WIDTH* of the image. *FRAC(2)* (following a suggestion by Weekes) measures the general concentration of light: it is the fraction of light collected by all 37 tubes that is contained in the 2 largest signals. The orientation of the image relative to the (central) source is described by (a) *MISS*, the perpendicular distance of the centre of the field (the source) from the image axis, (b) the *AZIMUTHAL-WIDTH*, the r.m.s. image width relative to a new axis which joins the source to the centroid of the image, and (c) the *DISTANCE* (distance of image centroid from source), to be compared with the distance of the brightest point (tube with largest signal) from the source — thus related to the orientation of the skew images.

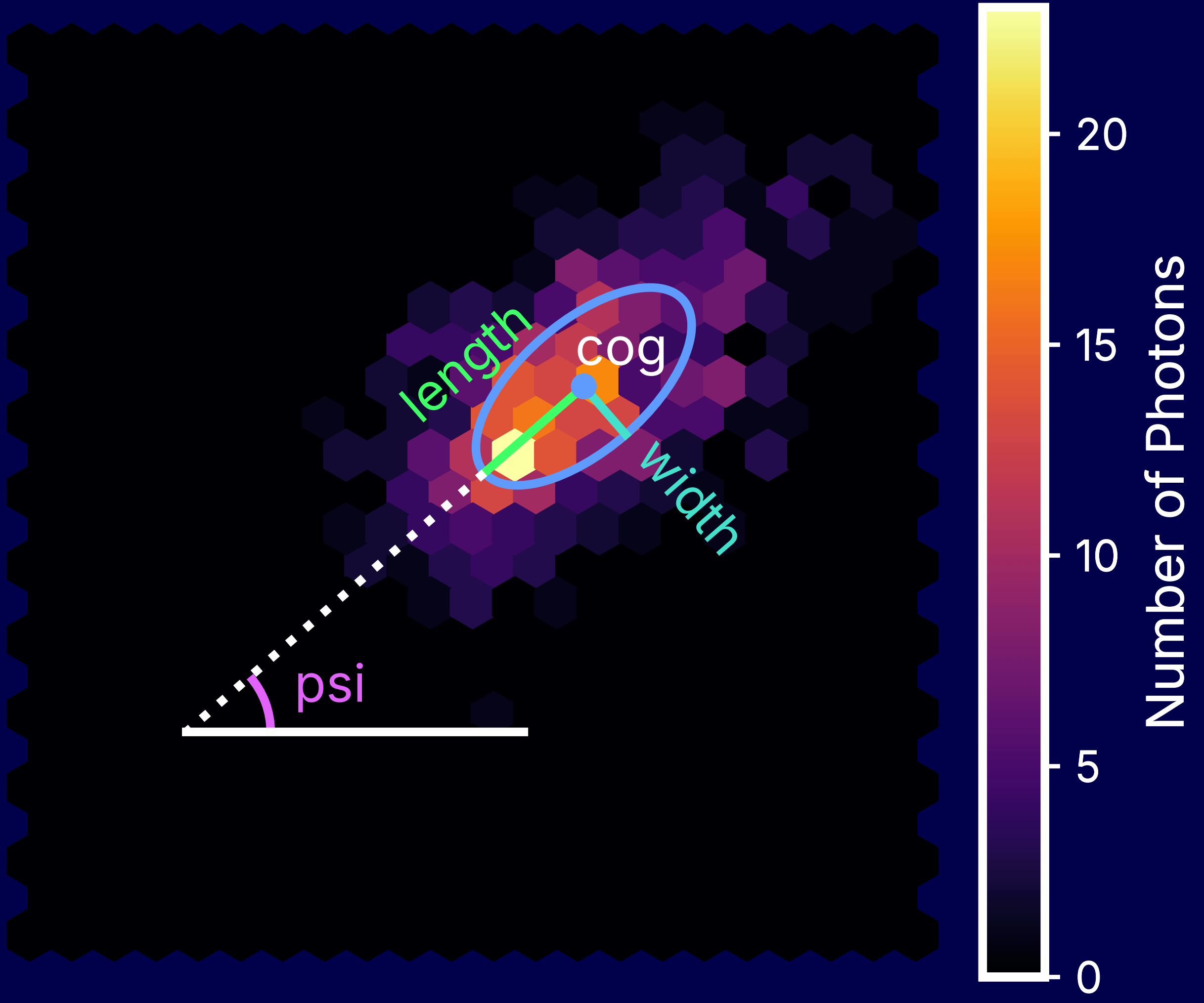
Side note: camera coordinate frames



Hillas Parameters Today

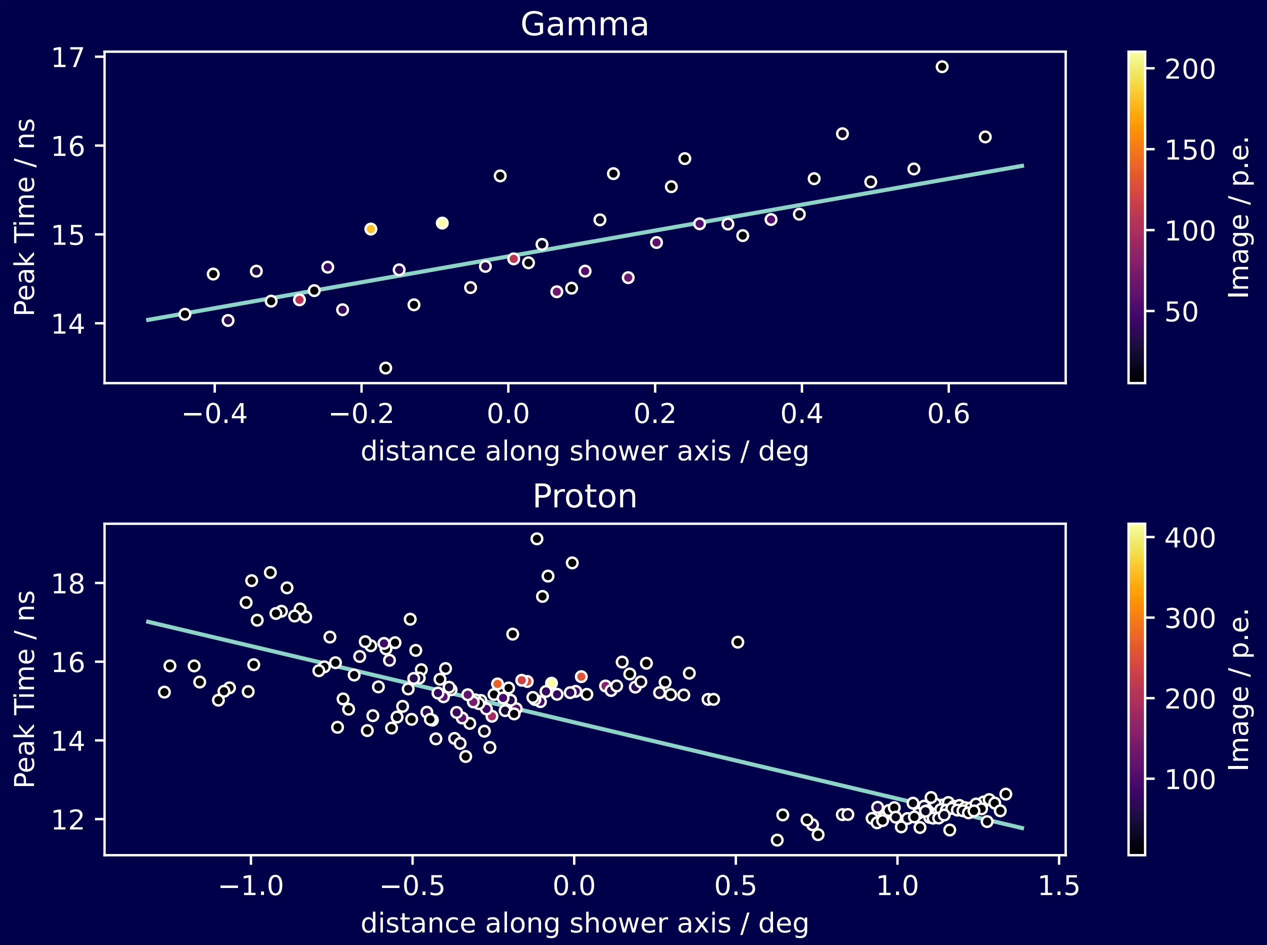
- The computation lined out by Hillas is equivalent to a well-known statistical method: Principal Component Analysis

  1. Compute weighted mean vector
     ⇒ center of gravity
  2. Compute covariance matrix
  3. Diagonalize covariance matrix by solving eigenvalue problem:
     - Eigenvalues give length and width;
     - Eigenvectors give orientation of major axis (psi)
- The total number of photoelectrons is called size or intensity
- Additionally, also compute higher level *moments* skewness and kurtosis along the shower axis
- Uncertainties can also be computed



Timing Parameters

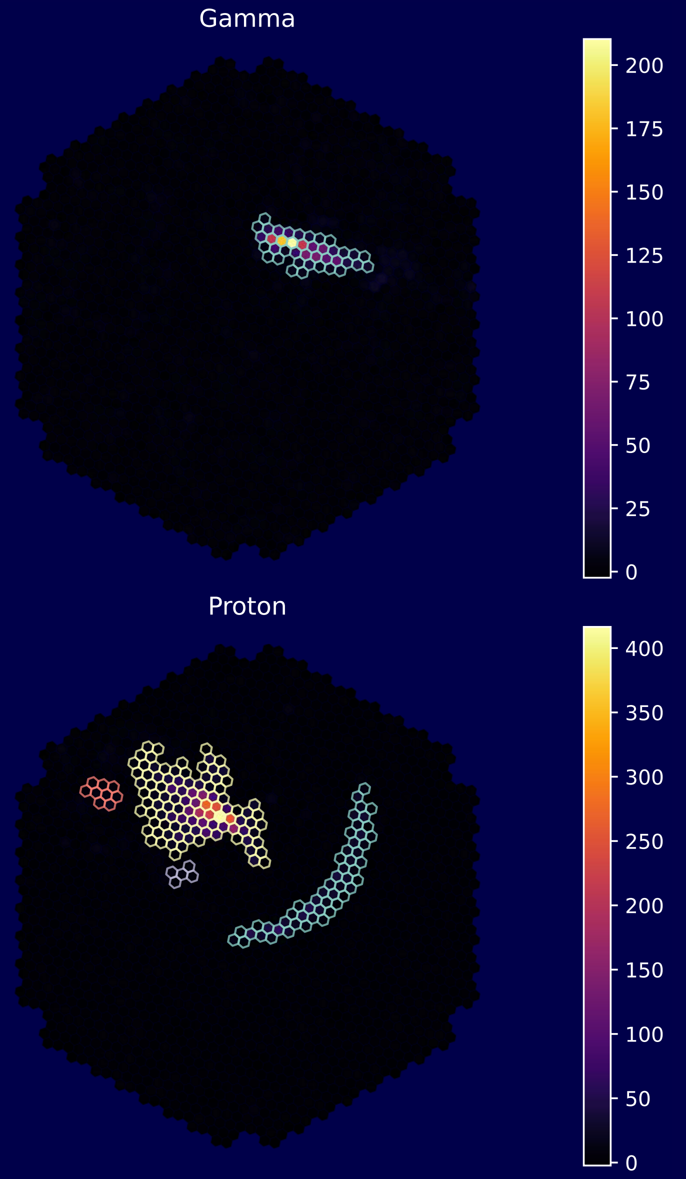
- After computing the Hillas Parameters, we can perform a fit of the peak times along the shower axis
- Slope is highly correlated with impact distance
- Sign of the slope is very useful for monoscopic direction reconstruction
- Good agreement with linear fit only for electromagnetic cascades, hadrons much more erratic



Morphology & Concentration

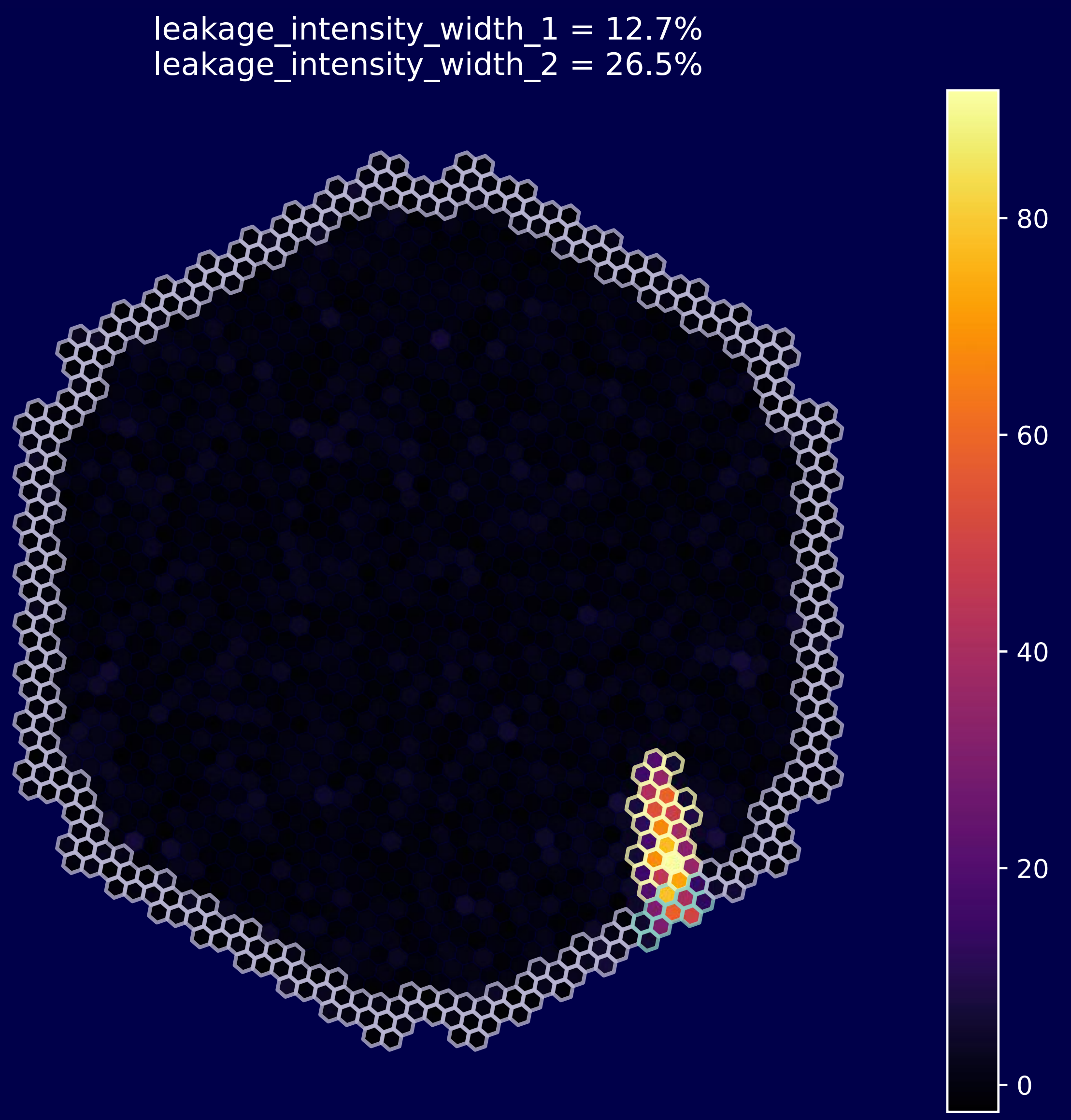
- For gamma-ray showers, we expect only a single blob of light
- Hadronic cascades have higher probability of sub-showers and muon rings
- Compute the number of “islands”, groups of connected pixels after cleaning
- Algorithm: breadth-first search of neighboring pixels
- Number of pixels after cleaning, number of all, small, and large islands
- Concentration features: ratios of intensity
    - inside of the Hillas ellipse
    - the brightest pixel
    - the three pixels closest to the cog

  to the total intensity



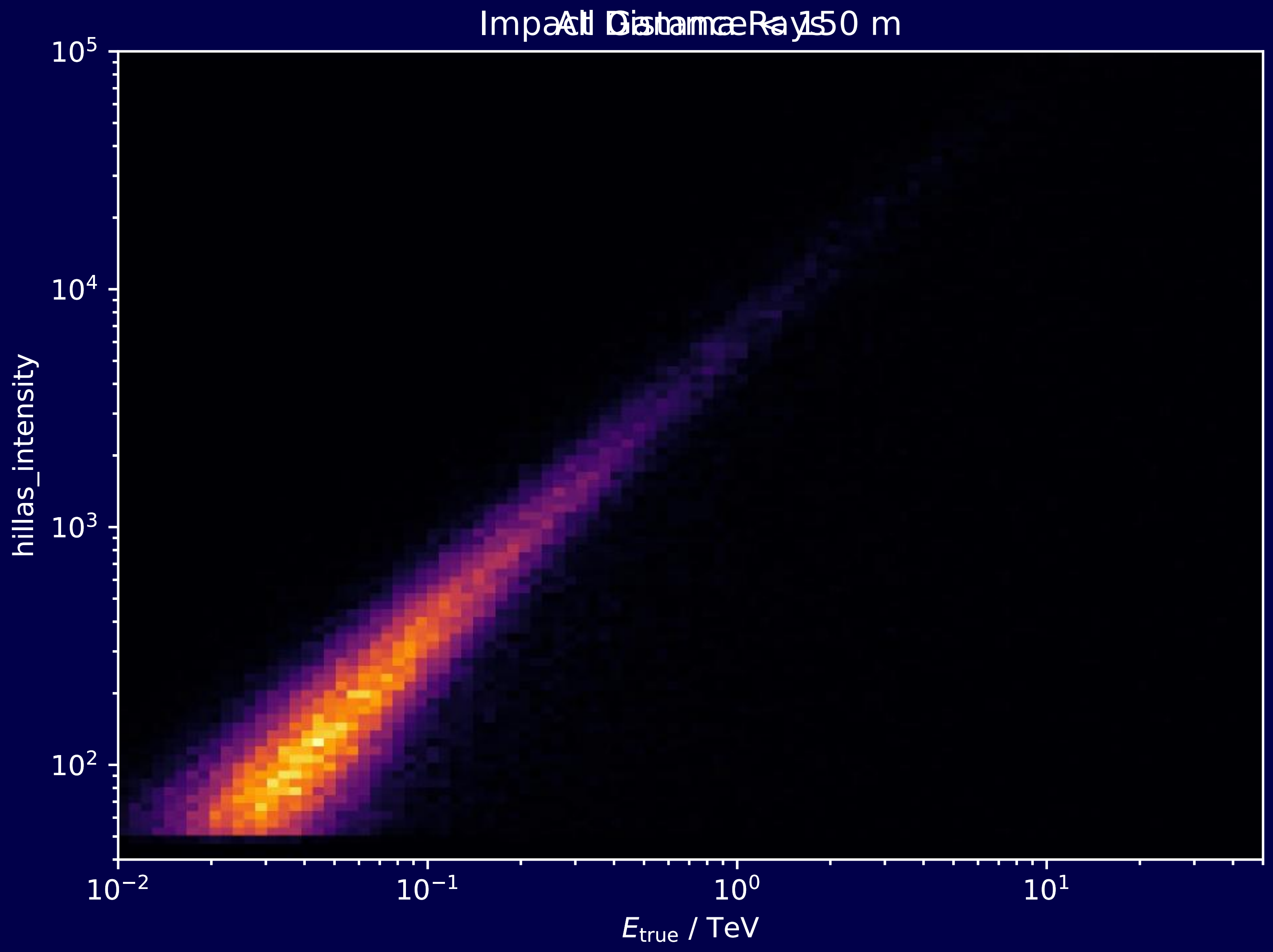
Leakage

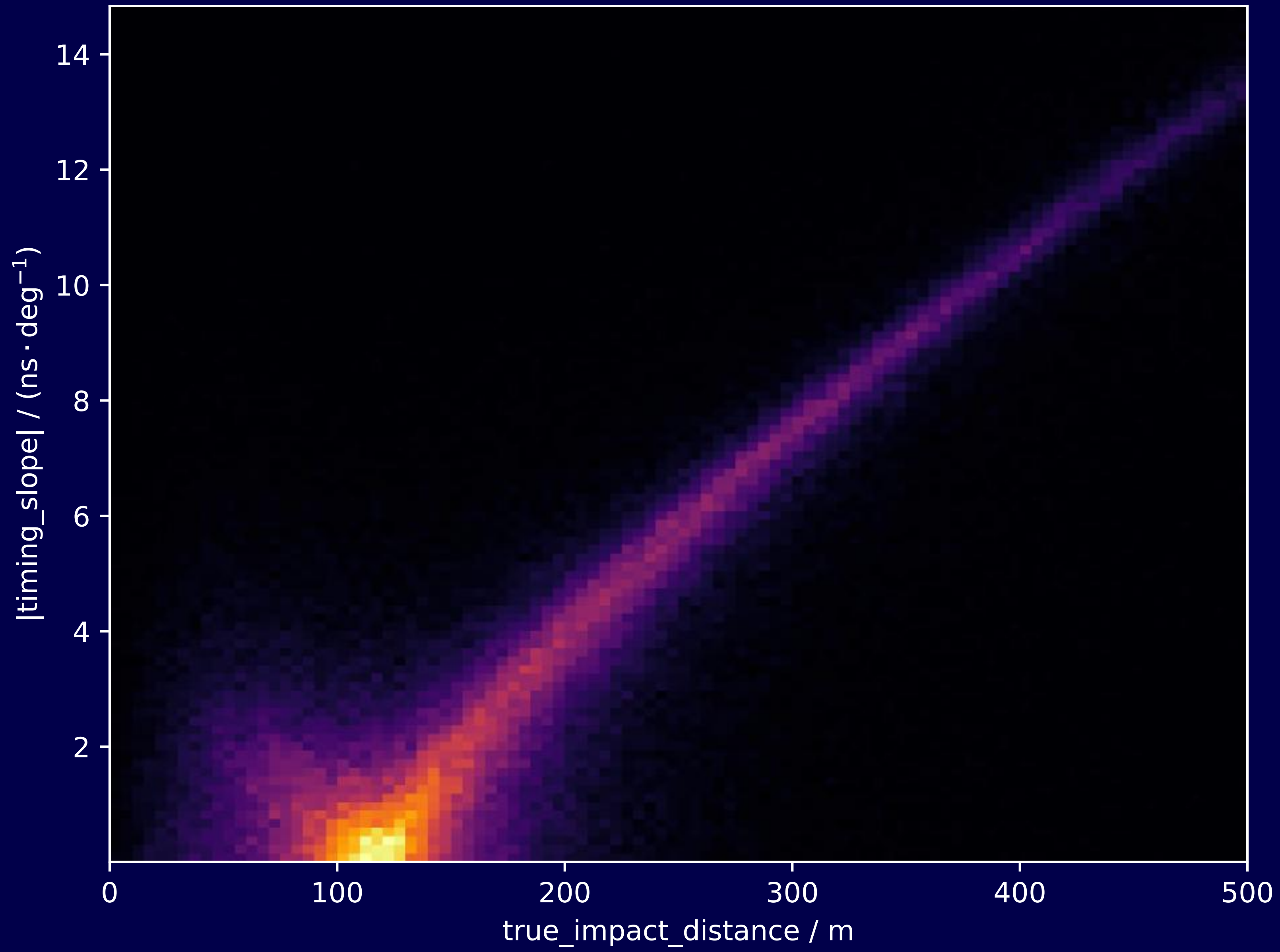
- Ratios of photoelectrons or pixels after cleaning in the outer edge of the camera
- To describe how well an event is *contained* in the telescope's FoV
- Mainly important for energy estimation and discarding non-contained events



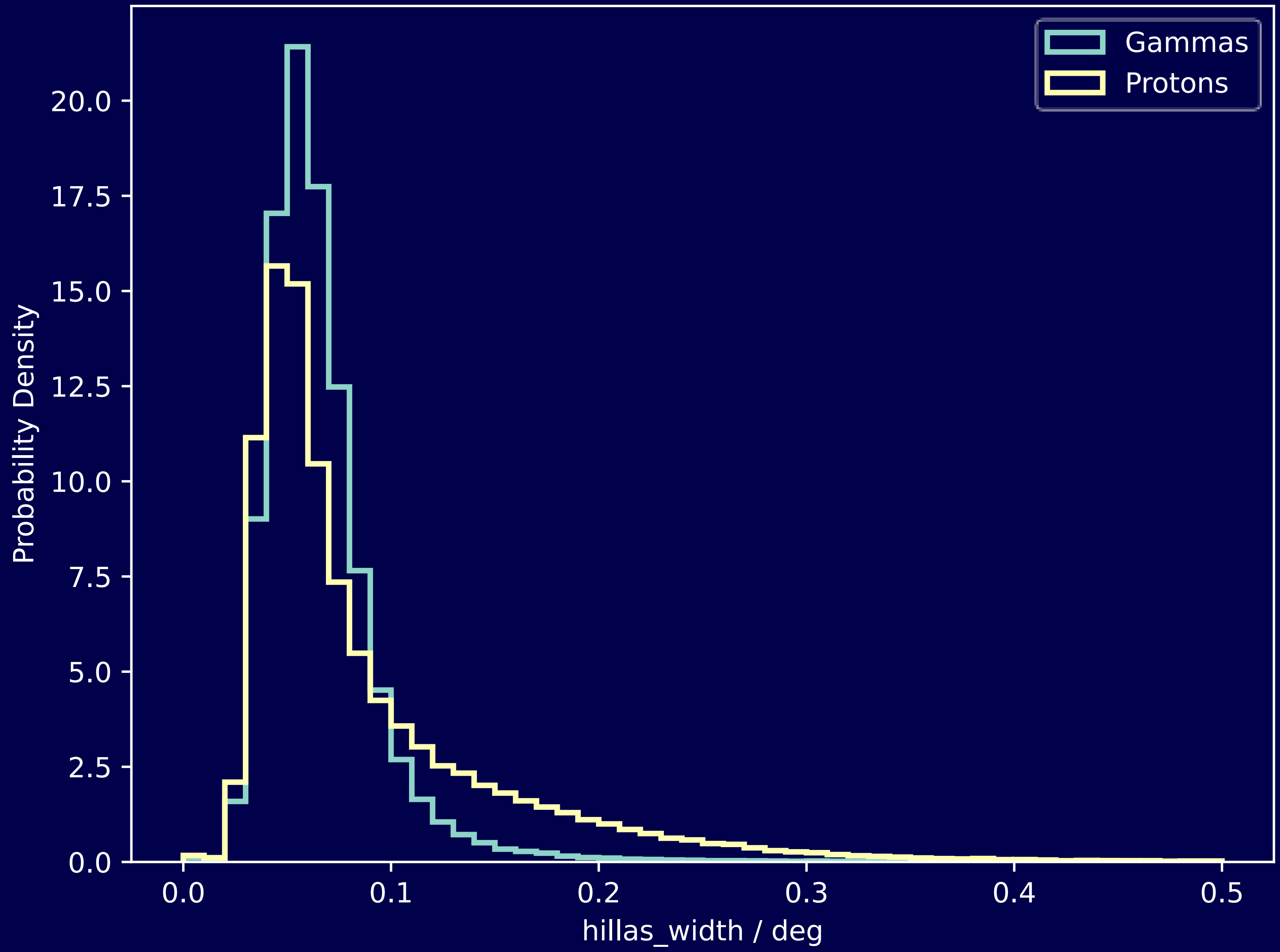
Descriptive Statistics

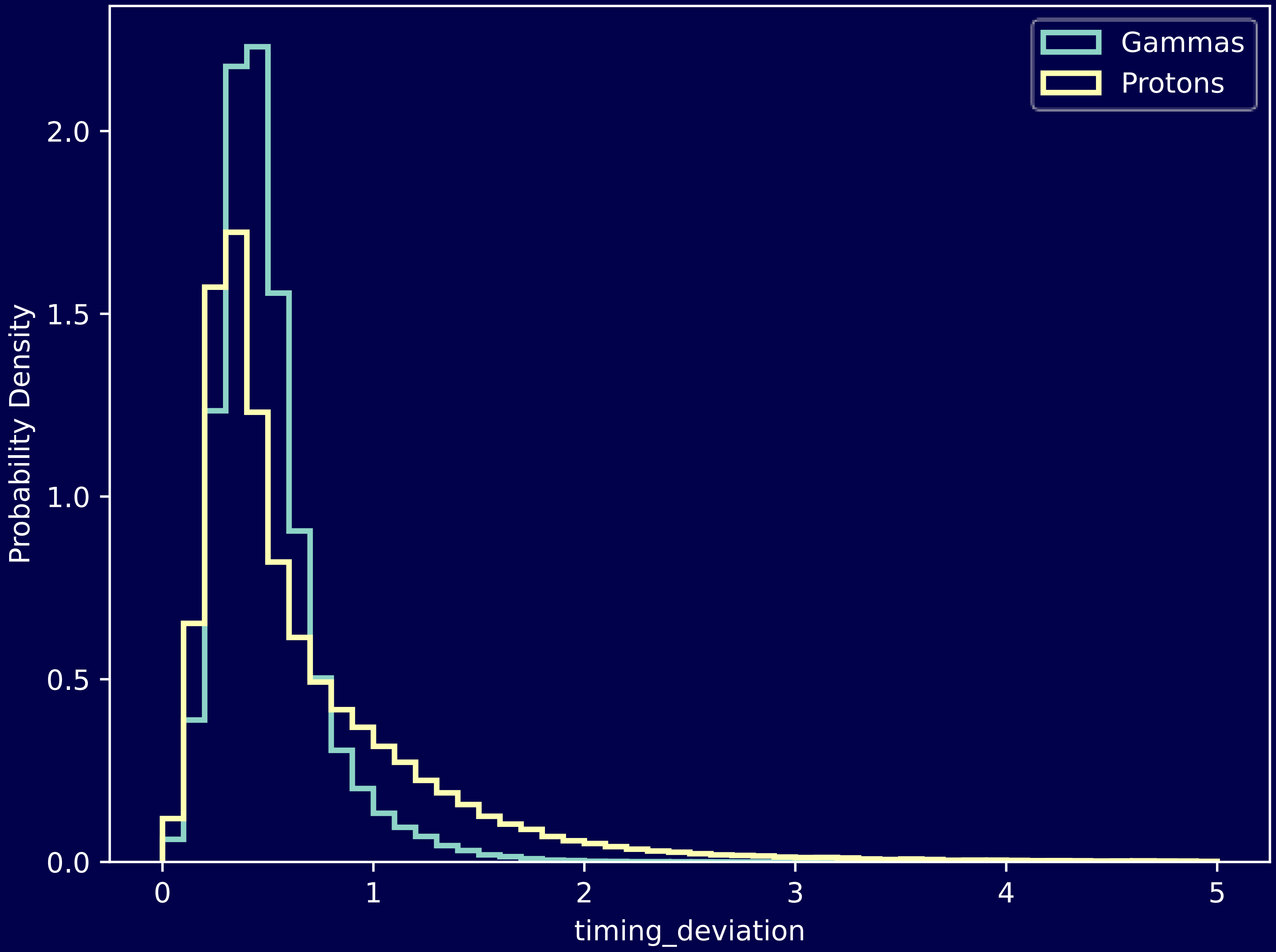
- We also collect descriptive statistics of pixel values, intensity and peak times:
    - Min
    - Max
    - Mean
    - Std
    - Skewness
    - Kurtosis


All Gamma Rays
hillas_intensity
10^5
10^4
10^3
10^2
10^{-2}
10^{-1}
10^0
10^1
E_{true} / TeV



14
12
10
8
6
4
2
0
100
200
300
400
500
true_impact_distance / m
|timing_slope| / (ns · deg⁻¹)

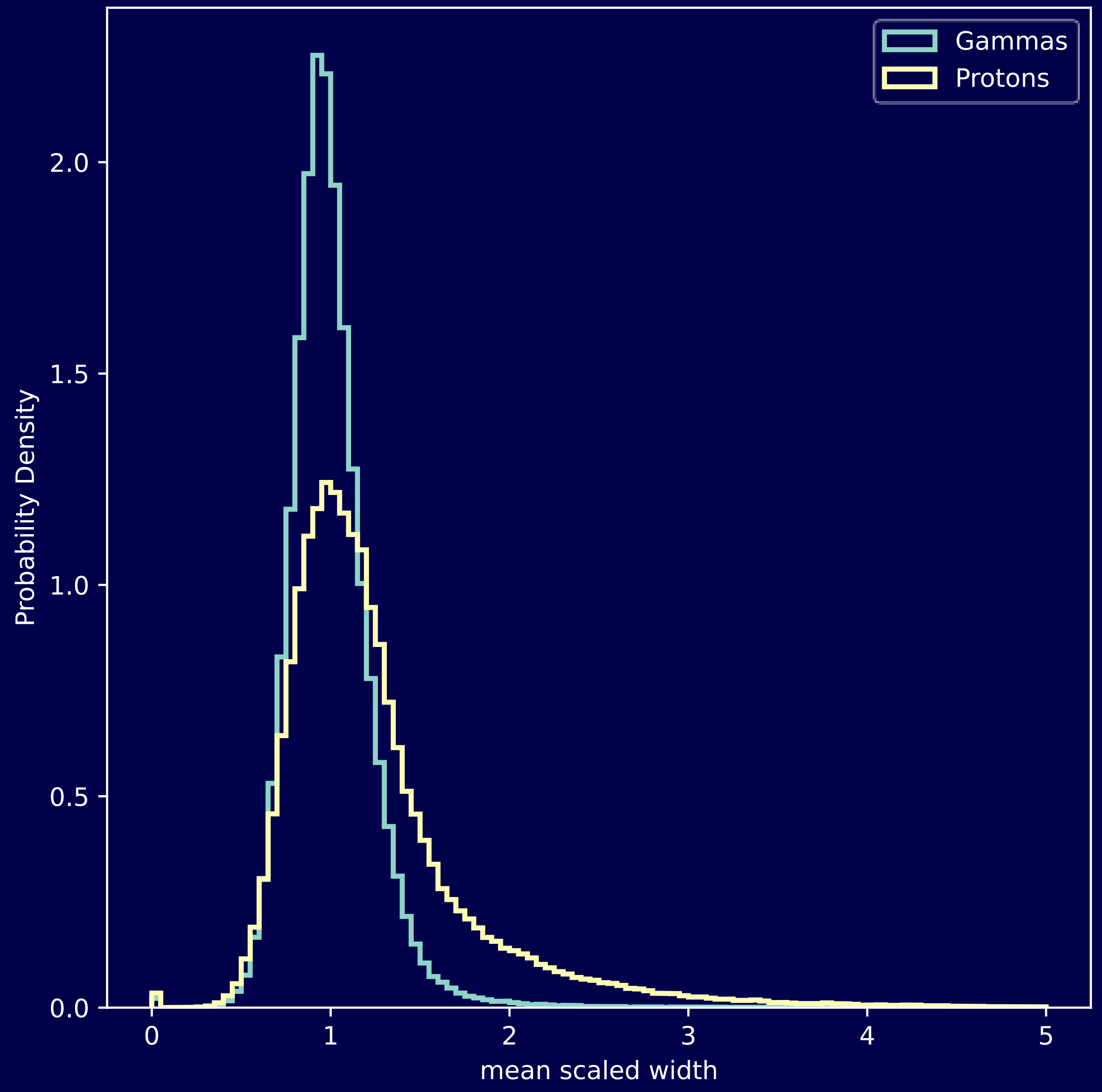

Gammas
Protons
Probability Density
hillas_width / deg
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
0.0
0.1
0.2
0.3
0.4
0.5



Gammas
Protons
Probability Density
timing_deviation
0.0
0.5
1.0
1.5
2.0
0
1
2
3
4
5

Mean Scaled Parameters

- This is a higher-level analysis step “from the old days, before machine learning”
- Still helpful to gain insights into parametrization
- Idea: scale image parameter by the mean of this image parameter for gamma rays (depending on image intensity, impact distance, zenith angle)
- Can also be averaged over telescopes
  ⇒ one number per subarray event
- Values around 1 are “gamma like”
- Values aways from 1 are “hadron like”
- Well suited for manual cut analysis or machine learning models that cannot deal with many features
- Less useful in e. g. tree-based learners



Reconstruction

Reconstruction Tasks

Three main tasks for each air shower event:

- Primary energy
- Direction on the sky
- Particle type

Additionally, these quantities might be useful to reconstruct:

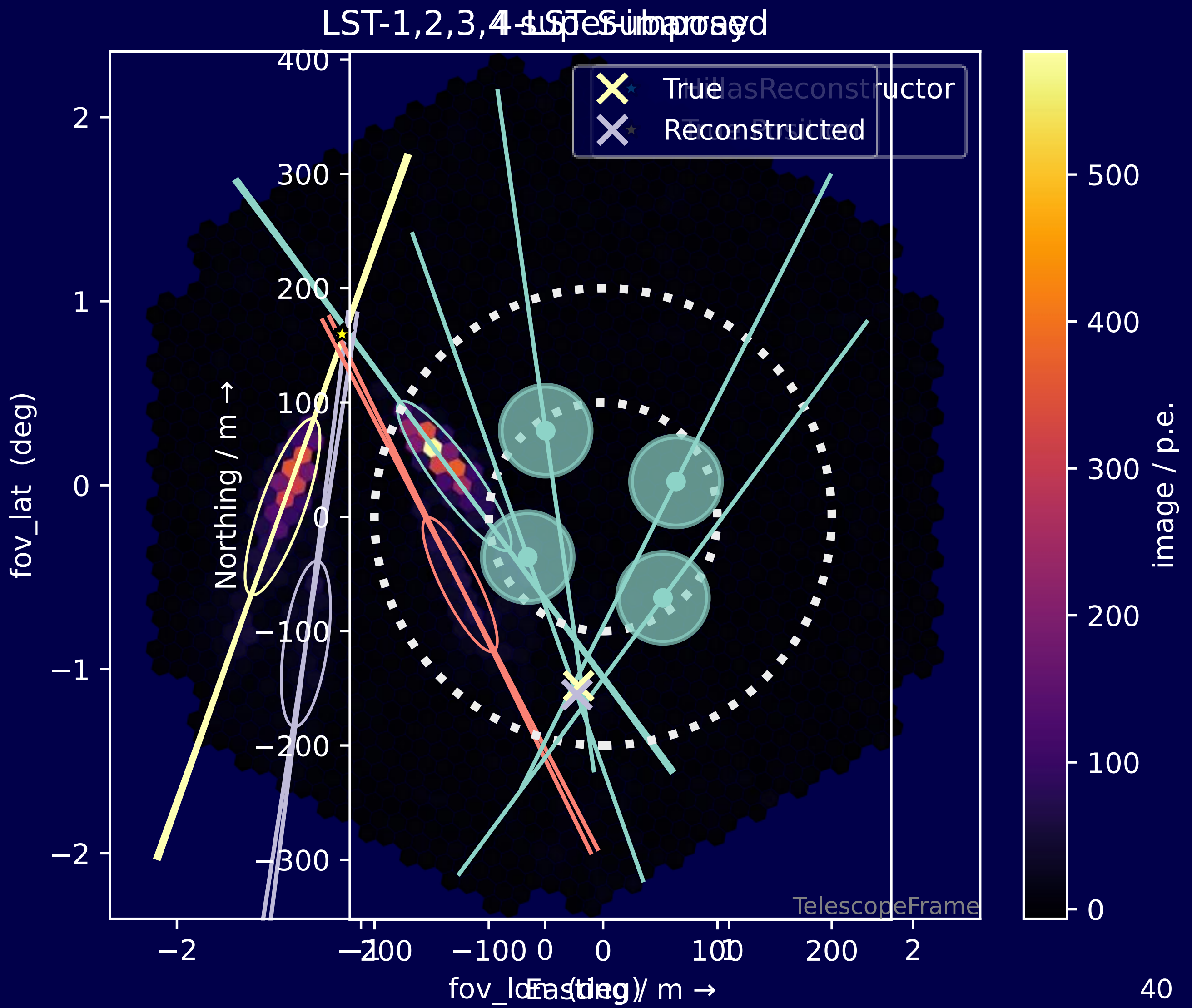
- “Impact” point on the ground
- Atmospheric depth of the shower maximum X_{max}

Together with the direction, these are also referred to as “shower geometry”

Hofmann, W., Jung, I., Konopelko, A., Krawczynski, H., Lampeitl, H., & Pühlhofer, G. (1999). *Comparison of techniques to reconstruct VHE gamma-ray showers from multiple stereoscopic Cherenkov images*. *Astroparticle Physics*, *12*(3), 135-143. doi:10.1016/S0927-6505(99)00084-5

Hillas Reconstructor

- Only method we'll discuss today that does not need simulated training data
- Purely geometrical
- Reconstructs 3D shower axis:
    - Direction in horizontal coordinates: altitude, azimuth
    - Impact point on the ground: x, y
- Least squares intersections of planes spanned by each telescope
- Requires at least 2 telescopes



Energy Reconstruction

- There is no way to “calibrate” IACT energy reconstruction purely by measurements:
  - No tens of GeV to hundreds of TeV photon test beams
  - Atmosphere part of the detector

⇛ Energy reconstruction is always based on simulated “training” data

- Classical approach is training a machine learning model per telescope type and averaging the predictions over the telescope events
- Tree based ensemble methods have proven robust and performant
  - Random Forests
  - Boosted Decision Trees

Excursion: Basics of Machine Learning

What is Machine Learning?

- We usually employ *supervised* machine learning, where a model is *trained* on a *labelled* dataset and can then be *applied* to new data
- We will use X and Y to denote the generic random variables
- A particular sample of these random variables will be x_i or y_i
- We can stack several samples in rows and combine multiple variables as columns of a matrix $\boldsymbol{X}$, e.g. if you measure $p = 2$ observables (height and weight) of $N = 200$ people, you get a matrix $\boldsymbol{X}$ of shape $(200, 2)$
- A more formal definition of supervised machine learning:

> "Given an $N \times p$ Matrix $\boldsymbol{X} \in \mathbb{R}^{\boldsymbol{N}\times\boldsymbol{p}}$ and some associated output vector $\boldsymbol{Y} \in \mathbb{R}^N$, find a function $f(X) = \hat{Y}$, that takes a vector $\boldsymbol{x} \in \mathbb{R}^p$ and returns a prediction $\hat{Y}$, where some 'loss function' $L\big(Y, f(X)\big)$ is minimized for all X"

Based on: "The Elements of Statistical Learning", Hastie, Tibshirani and Friedman (2009). Free PDF online.

Regression vs. Classification

- We distinguish two main cases:
  - Predicting a discrete output $Y \in \{g_1, g_2 \dots, g_k\}$, called *classification* with the special case of $k = 2$ called *binary classification*
  - Predicting a continuous output $Y \in \mathbb{R}$, called *regression*

- For our tasks, in general,
  - Energy reconstruction is 1-dimensional regression
  - Particle type is a classification, most often just a binary classification for signal vs. background
  - Geometry reconstruction is at minimum a two-dimensional regression, more variables might be desirable (X_{max}, Impact point).

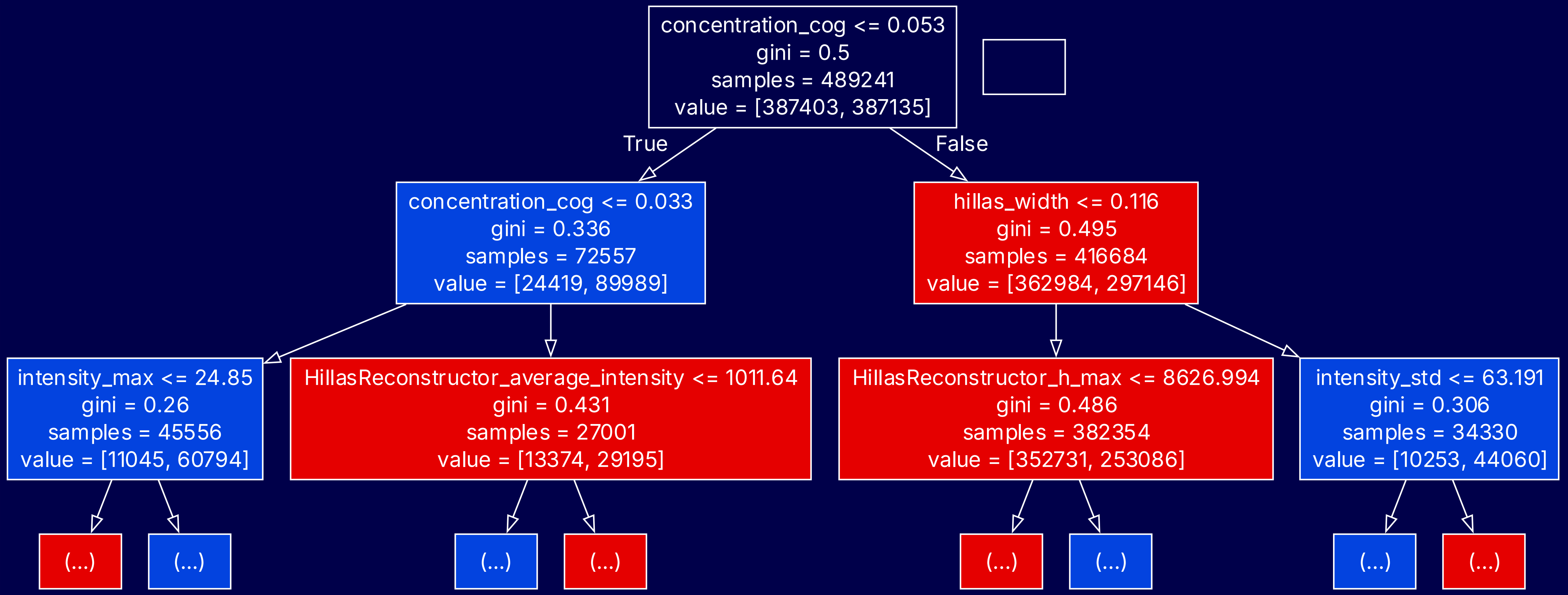
- Many different algorithms are available for both types of machine learning; some algorithm can do either

Decision Trees

- Algorithm:
  - Split the data space recursively by splitting in one feature at a time:
$$R_1(j,s) = \{X | X_j \leq s\}$$
$$R_2(j,s) = \{X | X_j > s\}$$
  - One split is called a *node*
  - Repeat until no more splits can be made, either because the leaf is *pure* or a maximum depth has been reached
  - For classification: predict the most common value in a leaf, for regression: predict the mean

- Training:
  - Finding the globally optimal splits is computationally infeasible (N-P-hard)
  - Instead, we recursively split, making the best split possible according to loss function for the current node (greedy approach) and never question again splits we already made
  - In each node, we need to find the best split among all possible splits in all possible attributes
  - Typical loss functions are based on entropy (classification) or mean squared error (regression)

Decision Trees



Random Forests & Boosted Trees

- A single tree is not very robust, due to the greedy algorithm, small fluctuations in the training sample result in large changes in the model
- Prone to *overfitting* if not constrained by setting a maximum depth, the model learns the statistical fluctuations of the training data instead of the underlying distributions
- Two approaches use *ensembles* of decision trees to improve over the single model
- *Random Forests* train N Decision Trees, randomizing the training in two ways to create slightly different models:
    - Each tree is trained on a dataset created by sampling with replacement from the original training data (bootstrapping)
    - In each node, only a random subset of all attributes is searched for the best split
  - The prediction is then the average of all individual trees
  - *Boosting* uses small trees (low maximum depth) that iteratively improve over the previous generation
  - Both Random Forests and Boosted Decision Trees have proven suitable, robust methods for IACT event reconstruction

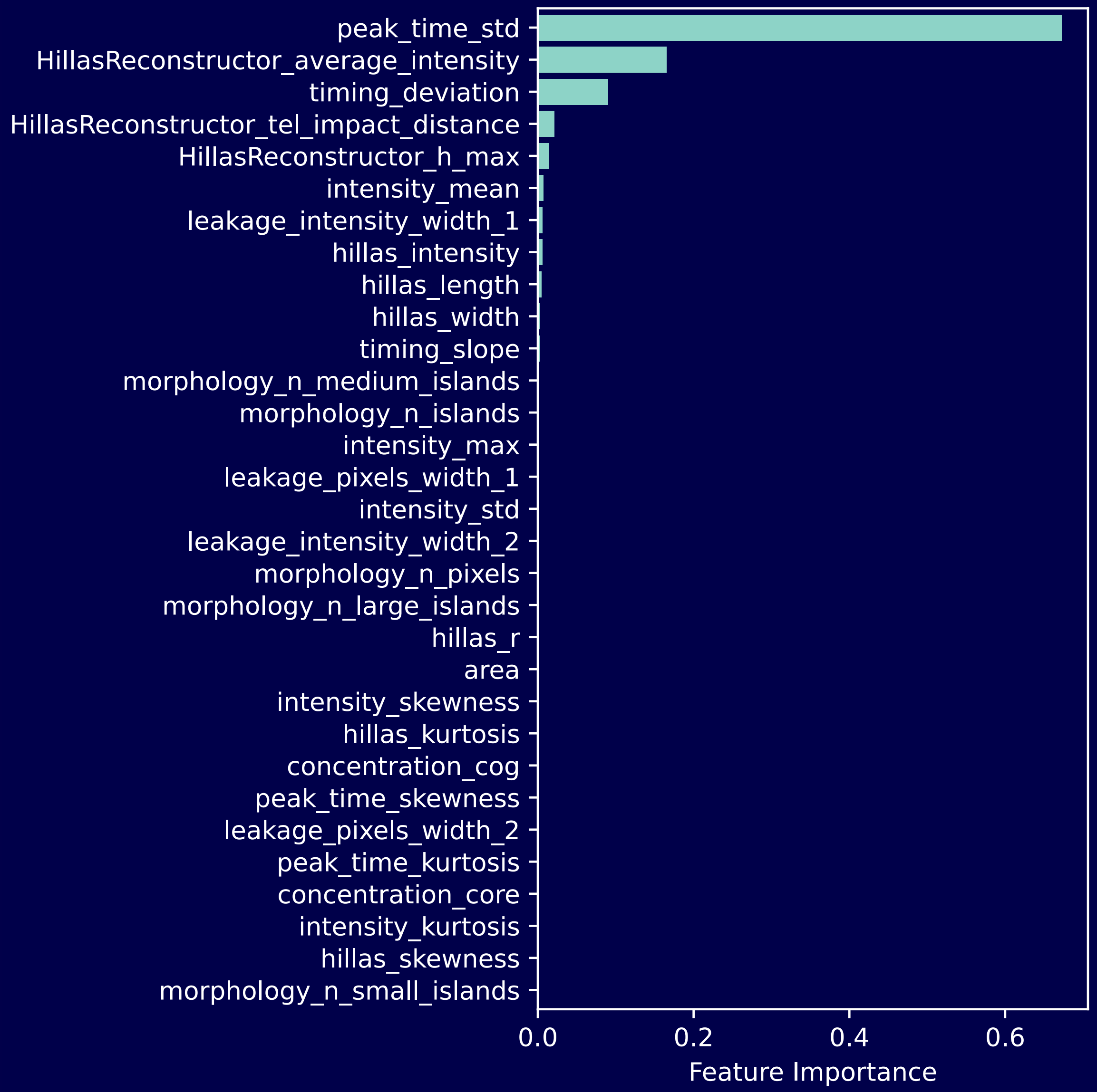
Energy Reconstruction

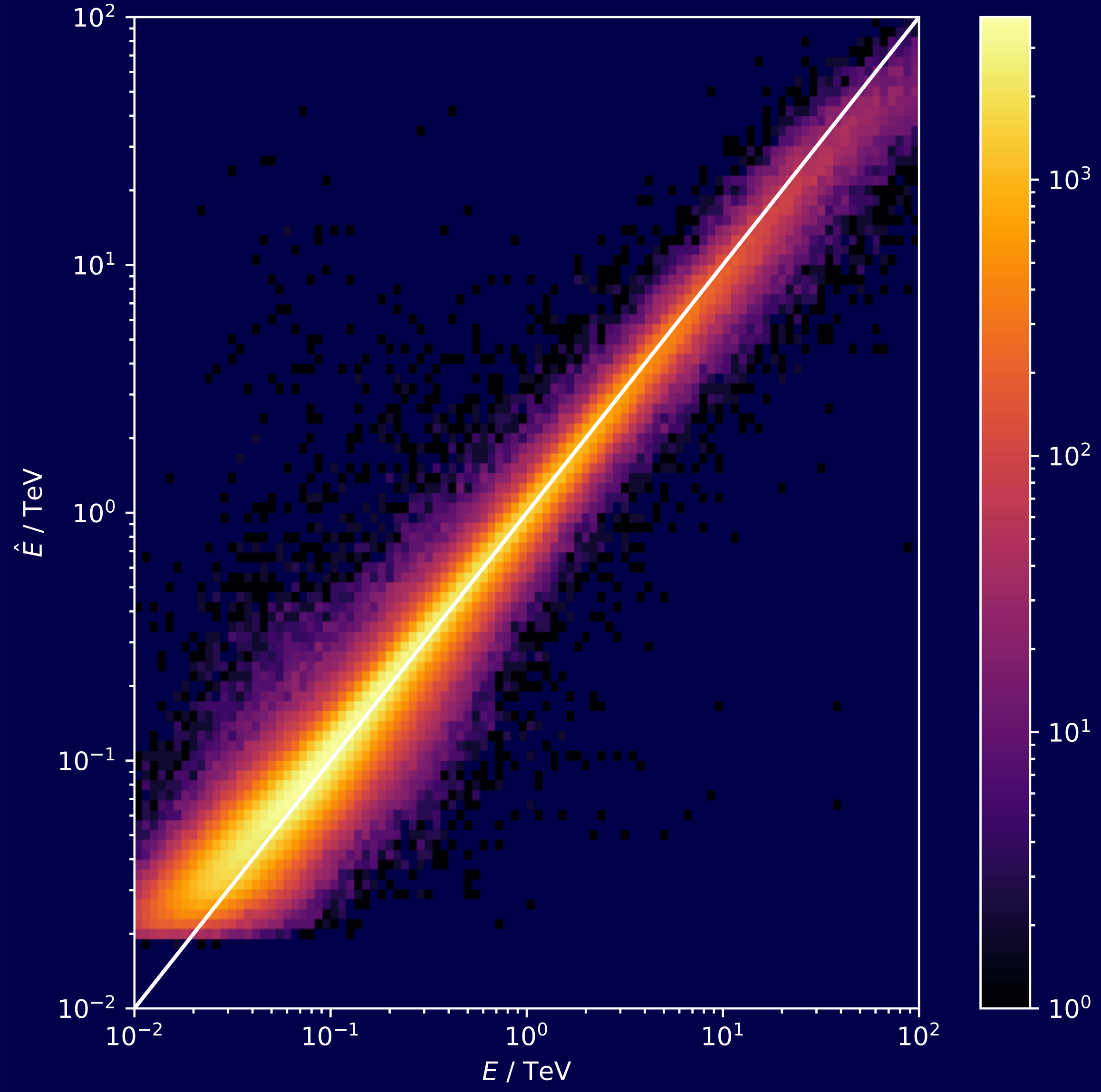
- There is no way to "calibrate" IACT energy reconstruction purely by measurements:
    - No tens of GeV to hundreds of TeV photon test beams
    - Atmosphere part of the detector

⇒ Energy reconstruction is always based on simulated "training" data

- Classical approach is training a machine learning model per telescope type and averaging the predictions over the telescope events
- Tree based ensemble methods have proven robust and performant
    - Random Forests
    - Boosted Decision Trees

Energy Reconstruction





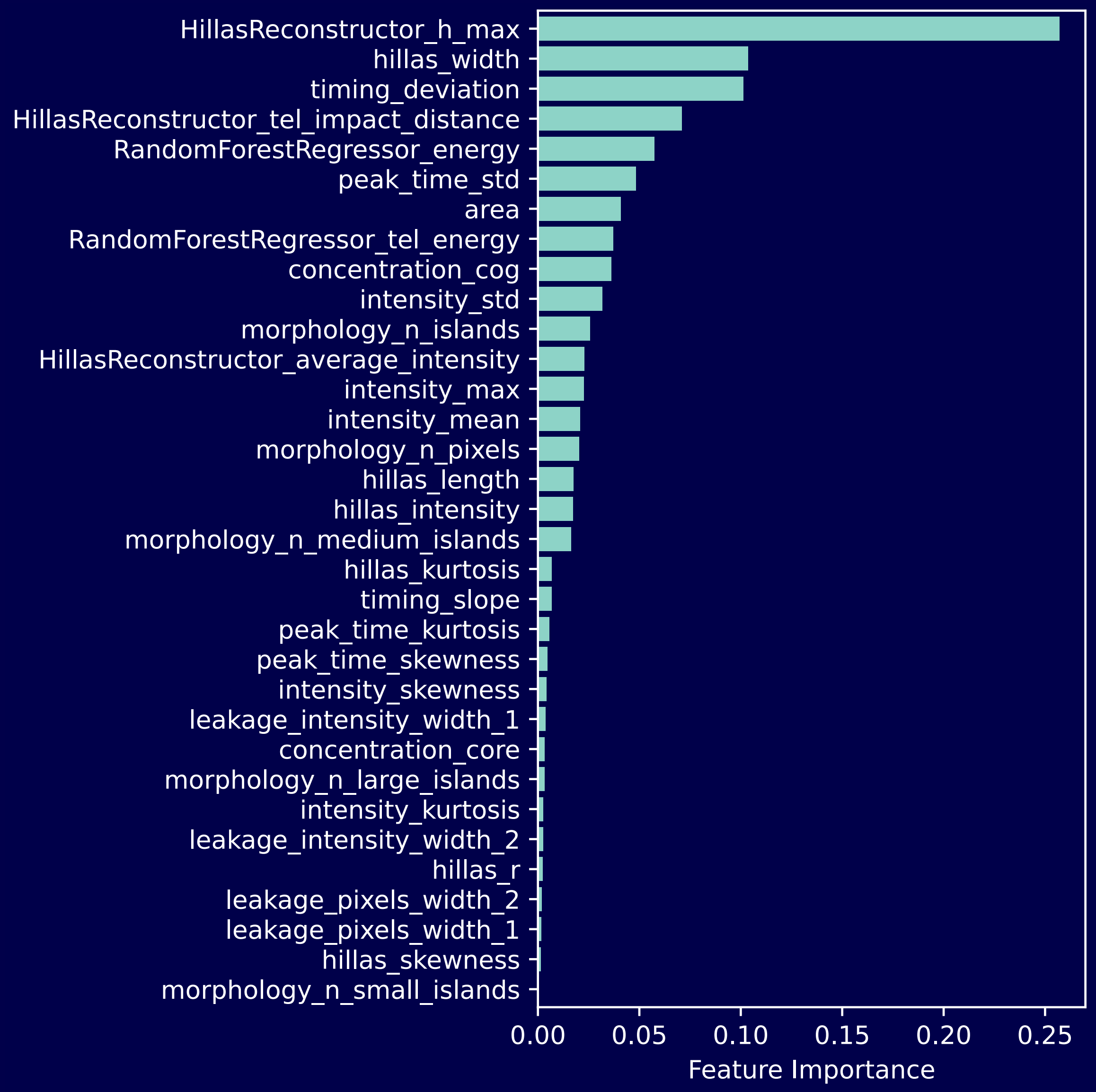
Particle Type Classification

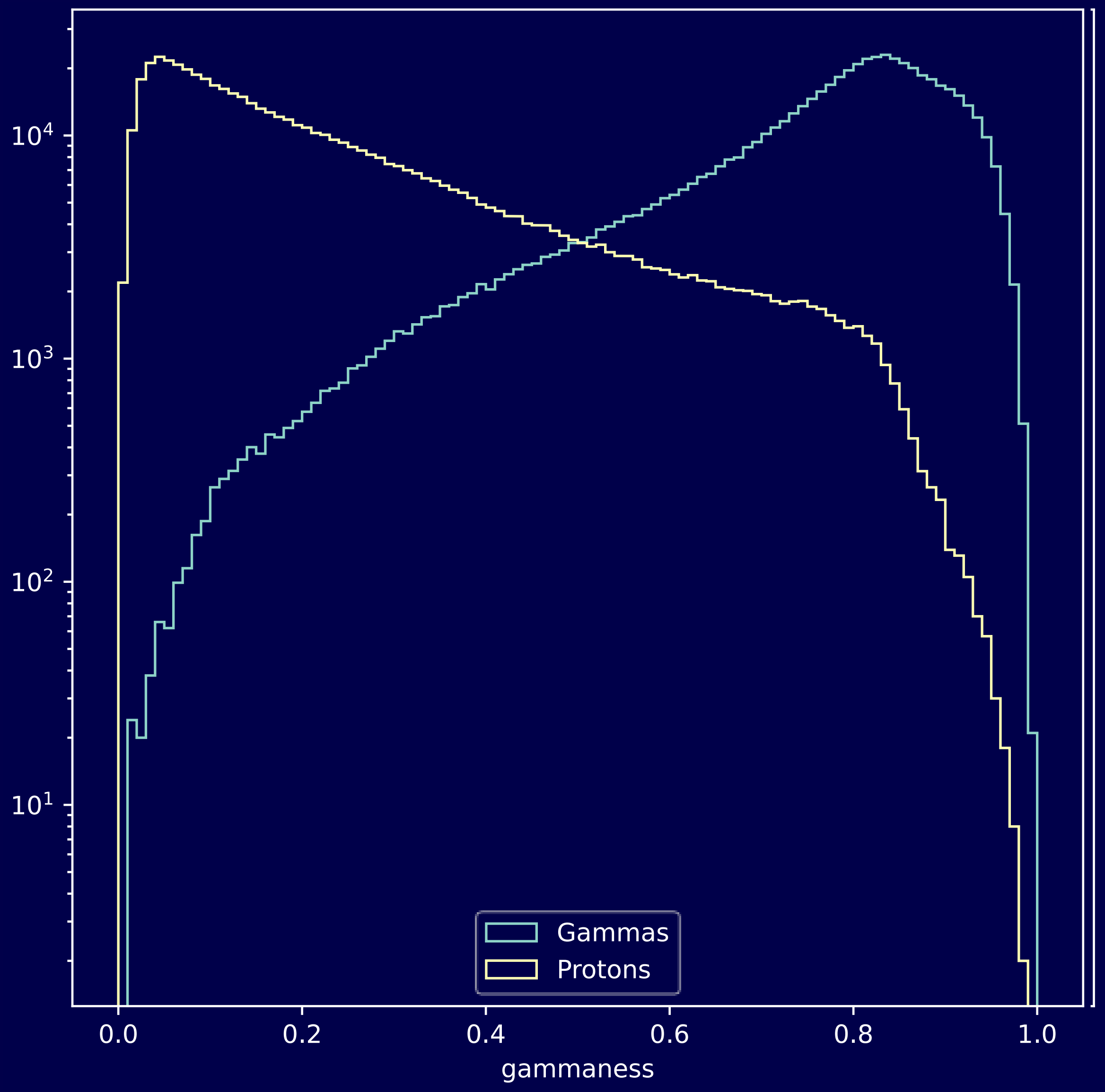
- As a gamma-ray observatory, for almost all science cases we can simplify to binary classification:

  Signal, gamma rays ⇔ background, everything else

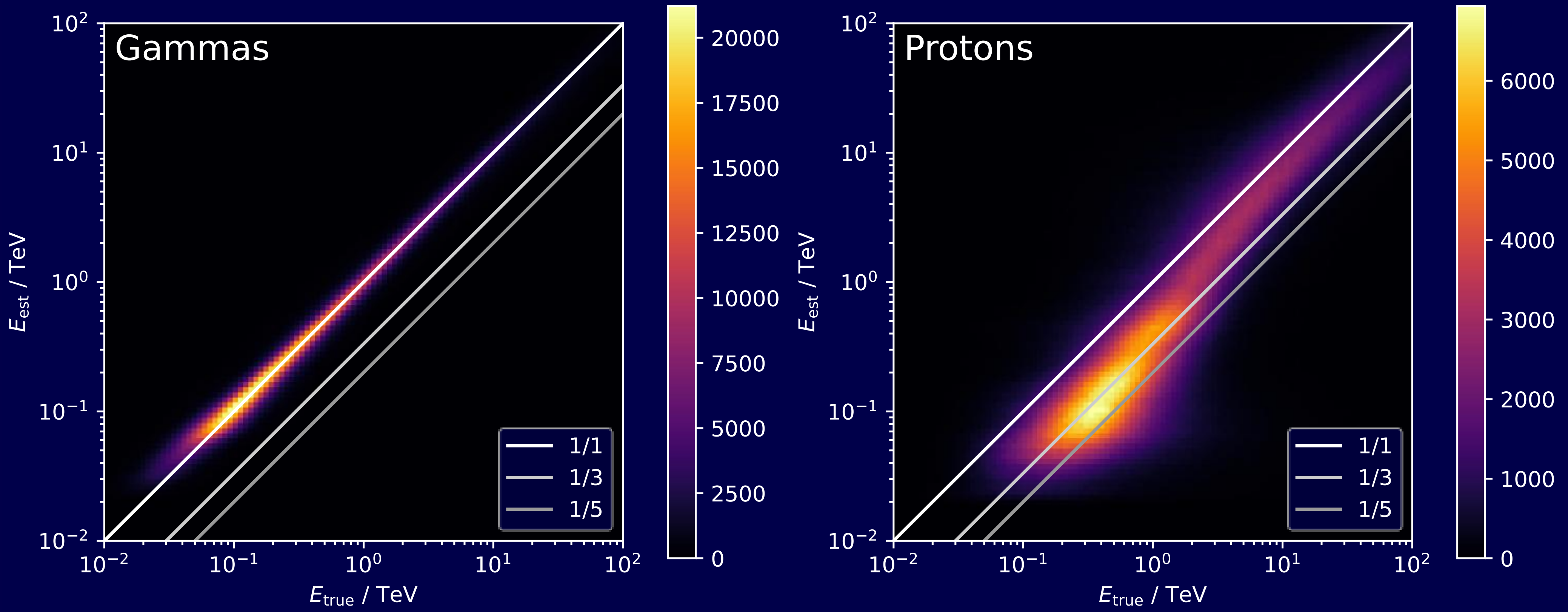
- Training needs labelled datasets for signal and background
- Signal is almost always taken from simulated gamma-ray events
- Background can either be simulations as well or taken from observations of "dark spots"
- Hadronic air showers:
  - are much more expensive to simulate
  - interaction models have known issues in describing air showers (muon puzzle)
- Using real observations for background training comes with its own caveats:
  - Model might learn to differentiate simulations from observations
  - Needs available observations matching conditions of observations to be analyzed
- MAGIC usually uses observed background, LST-1 uses simulated protons

Particle Type Classification





Side note on comparing energies

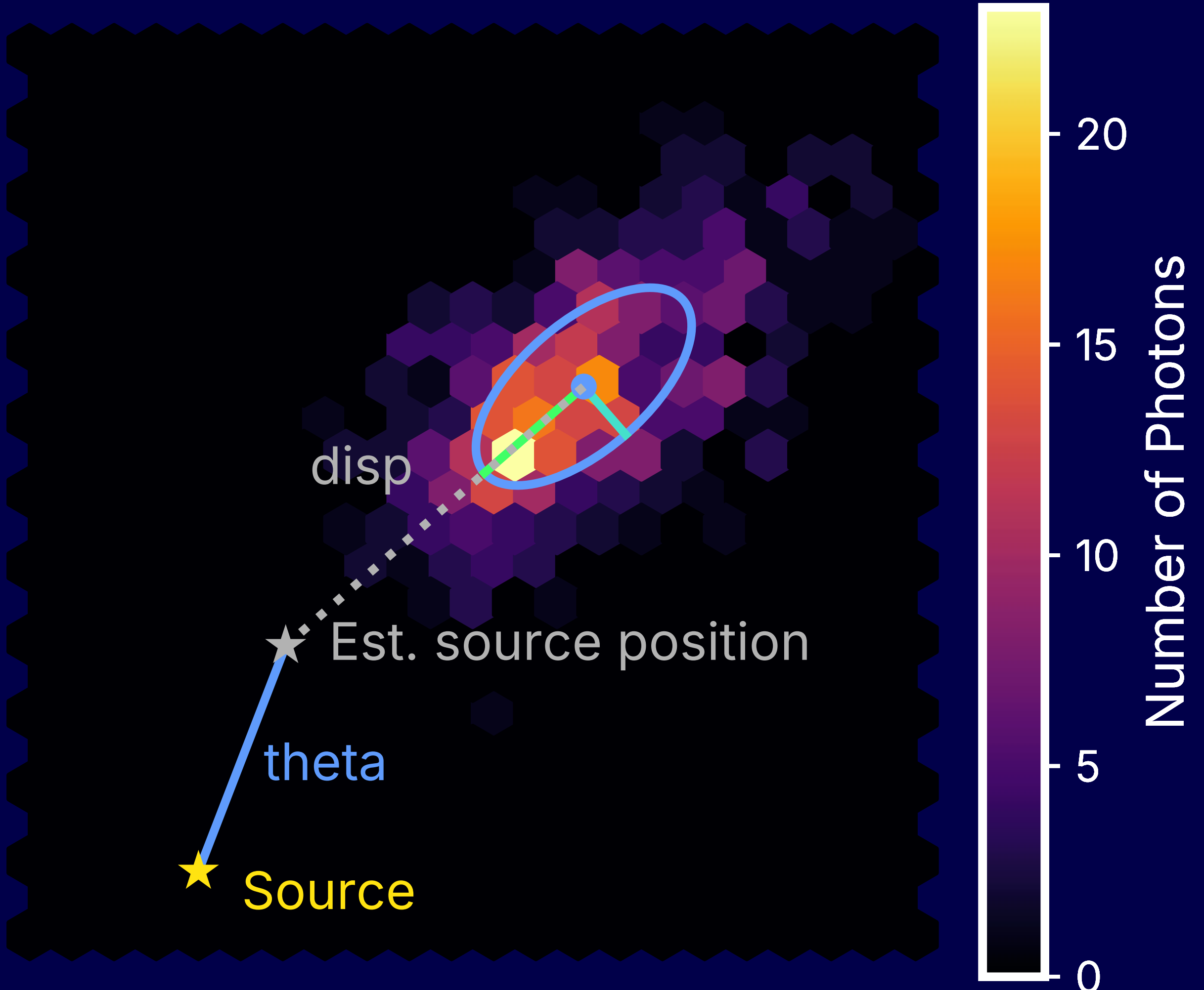


- The energy regressor is only trained on gamma rays → predicts "gamma-like" energy
- Hadronic showers create less Cherenkov light at the same true total energy
- Never compare events at the same true energy
- Use observables like Hillas intensity or an estimated gamma-ray energy

Monoscopic Reconstruction: The Disp-Method

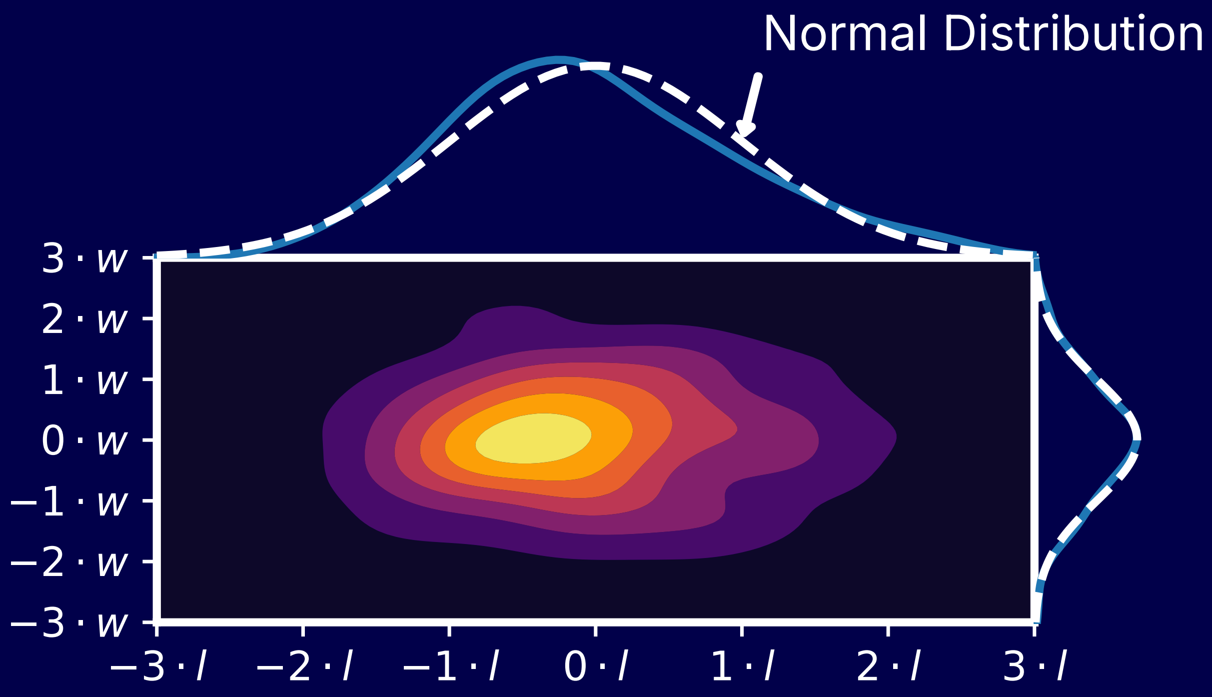
- Assume the source lies on the shower axis estimated by the Hillas parametrization
- Simplifies general 2D regression to
  - |disp|: 1D regression
  - sgn(disp): Binary classification
- ⇒ train one machine learning model for each task using the image features as input

Lessard, R. W., Buckley, J. H., Connaughton, V., & Le Bohec, S. (2001). A new analysis method for reconstructing the arrival direction of TeV gamma rays using a single imaging atmospheric Cherenkov telescope. *Astroparticle Physics*, *15*(1), 1-18. doi:10.1016/S0927-6505(00)00133-X



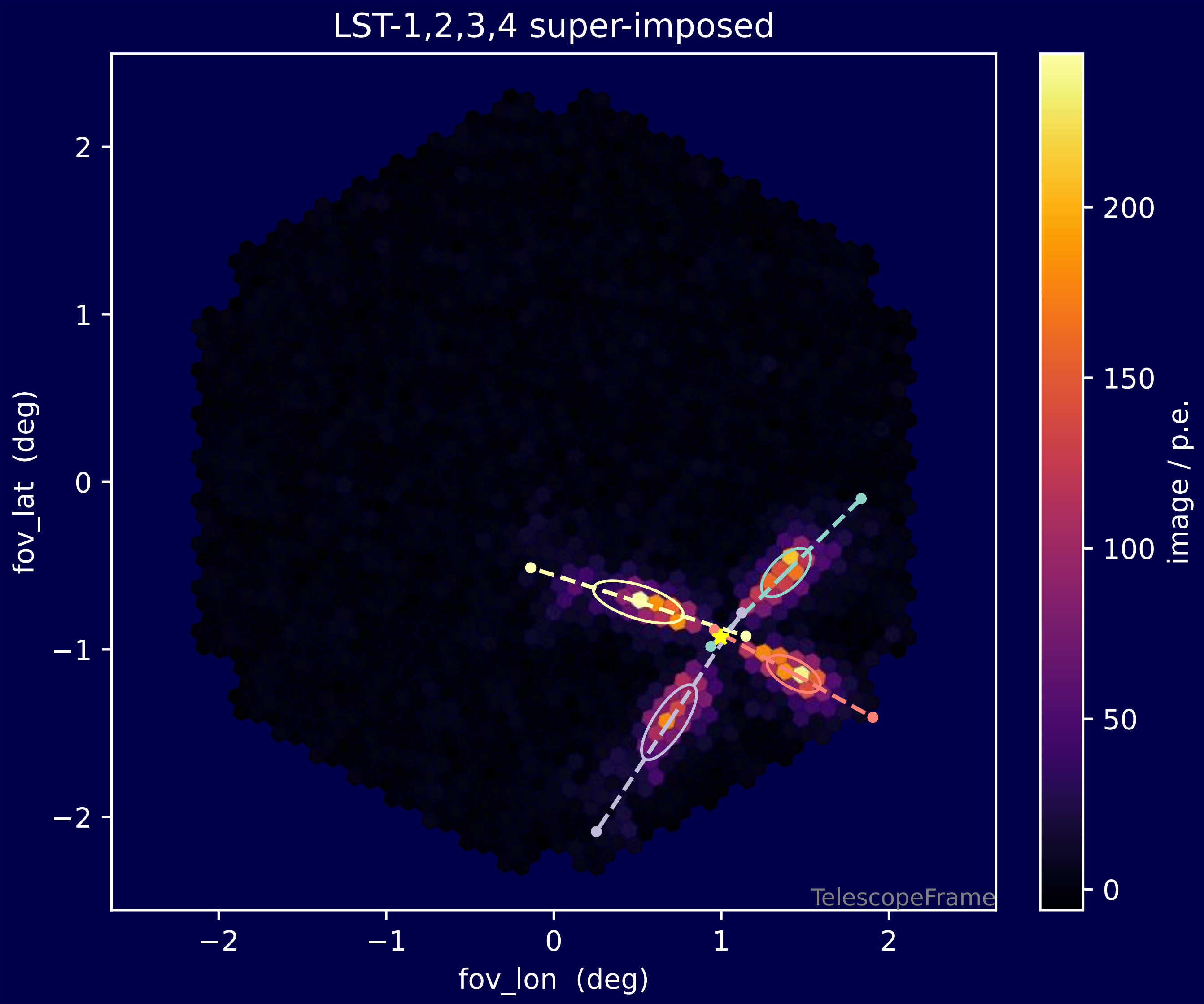
The Disp-Method: Head-Tail Disambiguation

- Showers are not perfectly symmetric along the main shower axis
- Allows models to learn and predict sign(disp)
- Usually, Random Forest Classifiers or Boosted Decision Trees



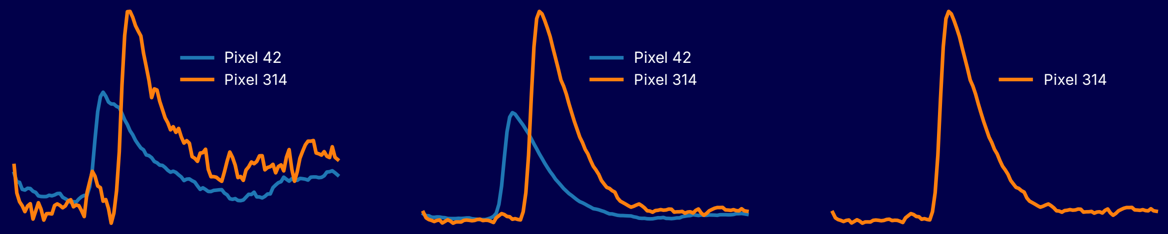
Stereo-Disp

- We can also combine the individual disp predictions to form a stereo prediction
- Has shown slightly better performance than the 3D reconstruction in EventDisplay, especially at low multiplicities and high energies
- Usually does not use sgn(disp) prediction
- ⇒ Search for best-fit among all solutions.



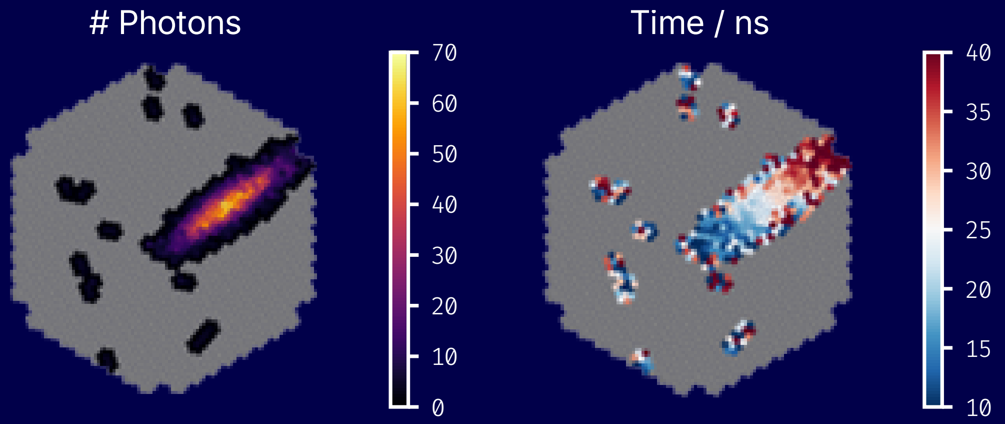
Advanced Reconstruction Methods

Can we take shortcuts here?



R0 → Calibration → R1 → Data Volume Reduction → DL0

Image Extraction



DL1a

Image Cleaning

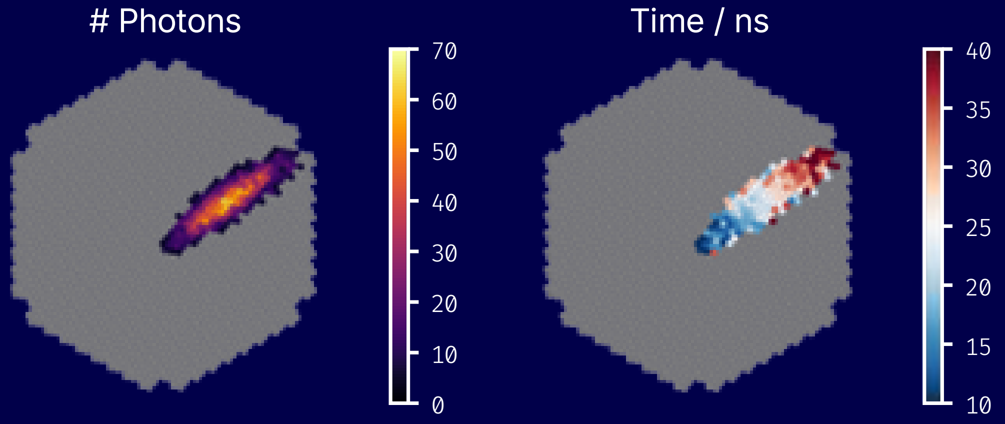


Image Parametrization

DL1b

Reconstruction

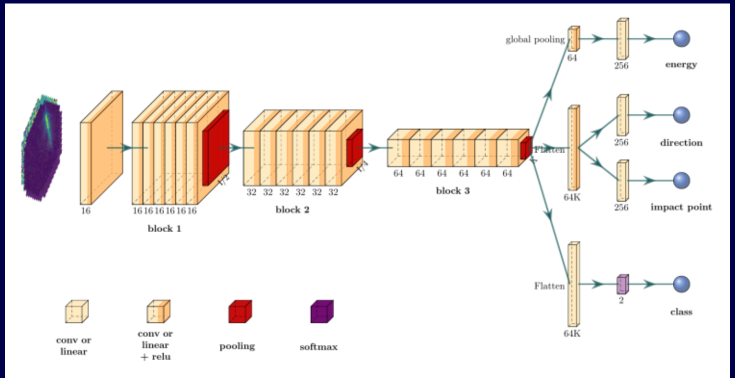
DL2

event_id	energy	gammaness	ra	dec	time
	GeV		deg	deg	mjd
1	1500	0.82	83.6	22.1	59024.63123
2	400	0.73	83.5	21.9	59024.64183
5	680	0.92	83.7	22.0	59024.67093

event_id	tel_id	hillas_width	hillas_length	…	intensity
1	1	0.15	0.52	…	1253.1
1	2	0.07	0.15	…	323.3
1	3	0.23	0.42	…	2012.1
2	1	0.03	0.11	…	321.3
2	4	0.05	0.12	…	263.3
5	2	0.09	0.15	…	512.7
5	3	0.08	0.19	…	637.4

Modern Machine Learning Approaches

- Several groups are working on using Deep Learning using Convolutional Neural Networks for IACT event reconstruction:
  - https://gammalearn.pages.in2p3.fr/pages/ (based on pytorch)
  - https://github.com/ctlearn-project/ctlearn (based on tensorflow)
- Predicting reconstructed properties directly from DL1 images or even DL0 waveforms
- Shown to outperform Image Parameters + Random Forests under controlled conditions (simulations, good quality data)
- Seem to struggle more with data/monte carlo mismatches, changing conditions, in general less robust



Likelihood-based Methods: ImPACT + FreePACT

- Based on simulations, it is possible to have a "template" function that predicts the image of a telescope given the true parameters of the air shower (energy, direction, impact, X_{max}) and the telescope properties (pointing, position on the ground).
- These templates can be used to perform a likelihood fit on the actual event data
- Initial method: ImPACT with look-up tables of the mean image built from specialized simulations interpolated at runtime
- FreePACT: a neural network trained to predict the likelihood of pixel values from the true gamma ray properties and the position and direction of the pixel replaces the template library
    - Better description of the likelihood (not just mean)
    - Can be trained on standard simulations

Schwefer, G., Parsons, R., & Hinton, J. (2024). A hybrid approach to event reconstruction for atmospheric Cherenkov Telescopes combining machine learning and likelihood fitting. *Astroparticle Physics*, *163*, 103008. [doi:10.1016/j.astropartphys.2024.103008](https://doi.org/10.1016/j.astropartphys.2024.103008)

Optimizing Event Selection

From DL2 to DL3 Event Lists

- After having performed the event reconstruction, we are left with a list of air shower events with predicted properties, maybe even from multiple algorithms
- In the classical analysis scheme, as last step of the event analysis, we need to:
    - Decide which set of reconstructed parameters to use
    - Apply the gamma/hadron separation to only retain gamma ray candidate events
    - Optionally apply additional event selection cuts, e. g. to discard badly reconstructed events
- How to optimize these decisions will vary greatly by your science case
- In current collaborations, people mostly start analysis at DL2 level and optimize for their analysis
- CTAO for most science cases, will hand out DL3 data, where this optimization has already happened.
- This is somewhat less flexible and restricts which science cases are possible / optimized for a given analysis configuration

Optimizing event selection

- There are many metrics that can be used to optimize event selection, most commonly:
  - Point source detection sensitivity in a given observation time (0.5 h, 5 h, 50 h)
  - Angular resolution
  - Gamma ray efficiency
  - Systematic uncertainties

- But: no clear high-level metrics for many science cases (extended sources, dark matter line searches, ...)
- Optimizing event selection for point source sensitivity will not yield the best results for almost any other science case, it's a very specific metric
- This is something you should keep in mind when using the public CTAO IRFs.

Instrument Response Functions

IRF Formalism

What we discussed up to now is a very *indirect* form of measurement suffering from two effects:

1. Limited *Acceptance*

   - Not all gamma rays that reach the Atmosphere create enough Cherenkov photons to trigger at least two telescopes
   - Not all images survive the image cleaning and can be parametrized
   - At least two telescopes are needed for stereo reconstruction in most cases
   - Gamma/Hadron separation is not perfect, leading to rejecting gamma rays as false negatives

2. Limited *Resolution*

   - All reconstruction methods have finite, non-negligible errors that can only be described on a statistical basis

IRF Formalism

We model the whole measurement process stochastically:

$$\underbrace{g(\hat{E},\hat{\alpha},\hat{\delta},\hat{t})}_{\text{Observed Signal Distribution}} = \int \overbrace{R(\hat{E},\hat{\alpha},\hat{\delta},\hat{t}|E,\alpha,\delta,t)}^{\text{Instrument Response Function}} \cdot \underbrace{f(E,\alpha,\delta,t)}_{\text{True gamma-ray signal}}\,\mathrm{d}E\,\mathrm{d}\Omega + \overbrace{b(\hat{E},\hat{\alpha},\hat{\delta},\hat{t})}^{\text{Residual Background}}$$

- The goal of any gamma-ray high-level analysis is to solve the *Inverse Problem* of deducing f from known g, R, b
- Computing R requires datasets with known ground-truth (labelled datasets)
- In high energy astroparticle physics, simulations are the only way to obtain such datasets
- The measurement equation is an inhomogeneous Fredholm integral equation of the second kind, by definition an ill-posed problem

IRF Factorization

- Thanks to very precise clocks and the time scales involved in the science cases of interest, it is almost always possible to treat
$$t = \hat{t}$$
- This still leaves us with a six-dimensional IRF that changes on minute time-scales
⇒ computationally infeasible
- Solution: Make *strong assumptions* on the statistical independence of the variables
⇒ factorization into independent, lower-dimensional *IRF Components*
- This assumption is *wrong*, energy and direction reconstruction are strongly correlated
⇒ Systematic errors

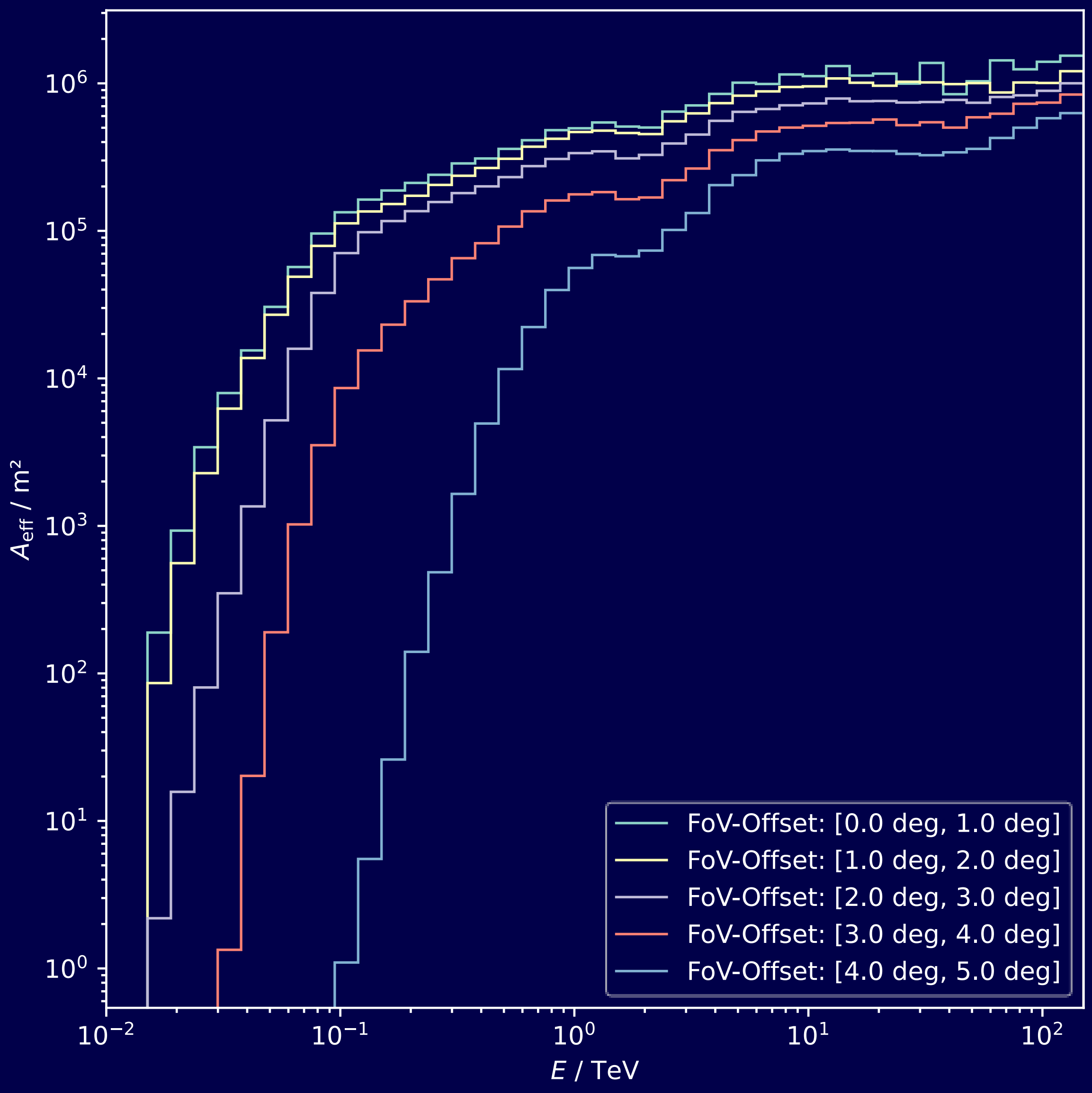
$$\overbrace{R(\hat{E}, \hat{\alpha}, \hat{\delta} \,|E, \alpha, \delta, t)}^{\text{Instrument Response Function}} = \underbrace{A_{\text{eff}}(E, \alpha, \delta, t)}_{\text{Effective Area}} \cdot \overbrace{M(\hat{E} \,|E, \alpha, \delta, t)}^{\text{Energy Dispersion}} \cdot \underbrace{\mathrm{PSF}(\hat{\alpha}, \hat{\delta} \,|E, \alpha, \delta, t)}_{\text{Point Spread Function}}$$

Computations of IRF

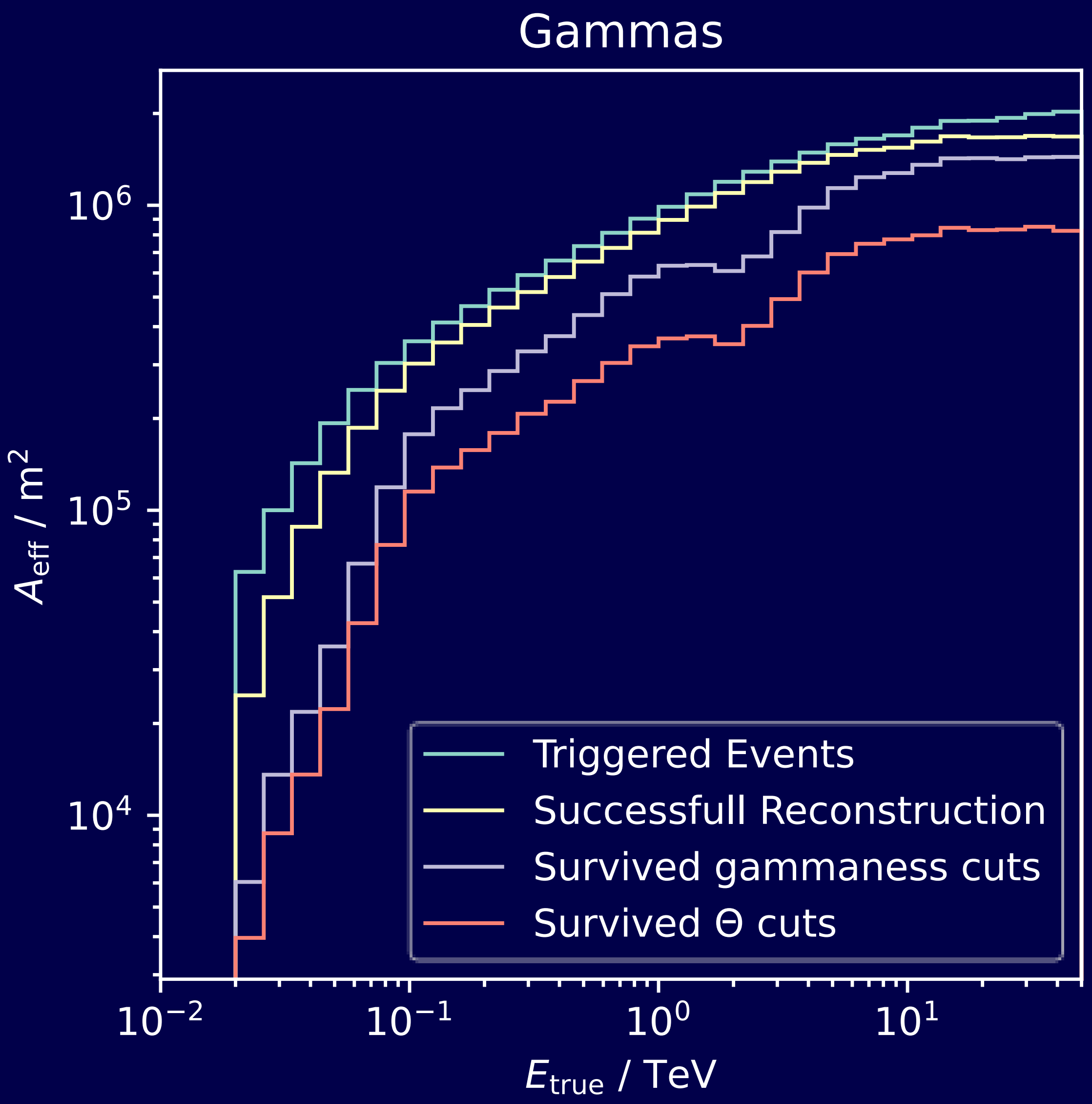
- In general, IRF components are computed from simulated events that have gone through the full low-level analysis chain
- Specific parametrizations of these components and how to store them in FITS files are defined in the Data Formats for Gamma-Ray Astronomy (GADF)
- Most components use tabulated values in parameter grids (energy, position in the field of view) of the discretized IRF components
- Most IRF components in GADF are limited to being radially symmetric in the field of view ⇒ another *wrong* assumption that we likely need to lift for CTAO
- IRFs for specific observation conditions can be interpolated from a grid of IRFs computed from simulations

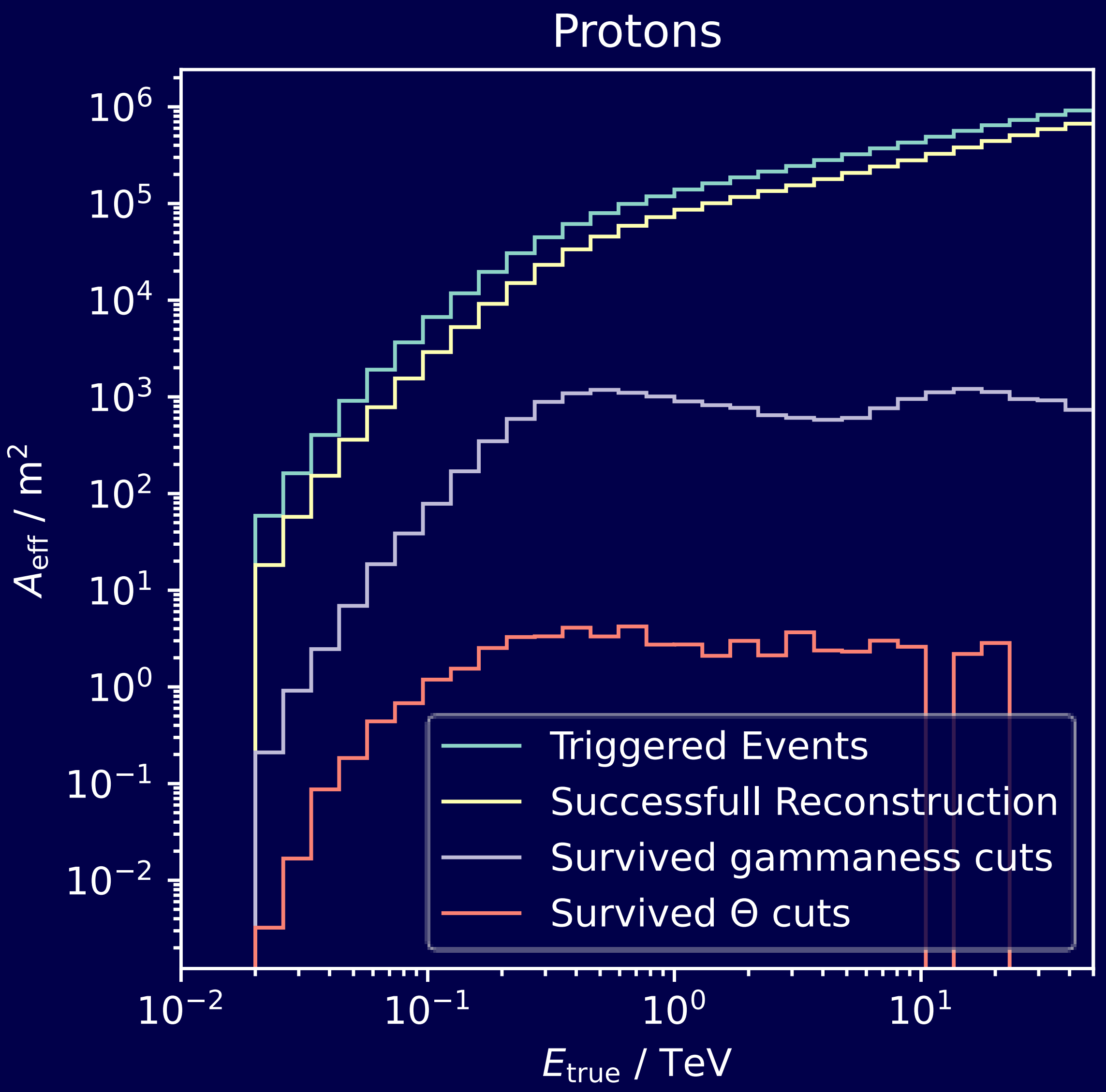
Effective Area

- Effective area combines the detection efficiency with the sensitive area of the detector
- Converts event rates to flux per time and area
- Corrects for the non-detection of events
- Can be interpreted as the area of a perfect, direct detector
- GADF supports effective area as tabulated values in bins of true energy and FoV offset ⇒ radial symmetry



Effective Area



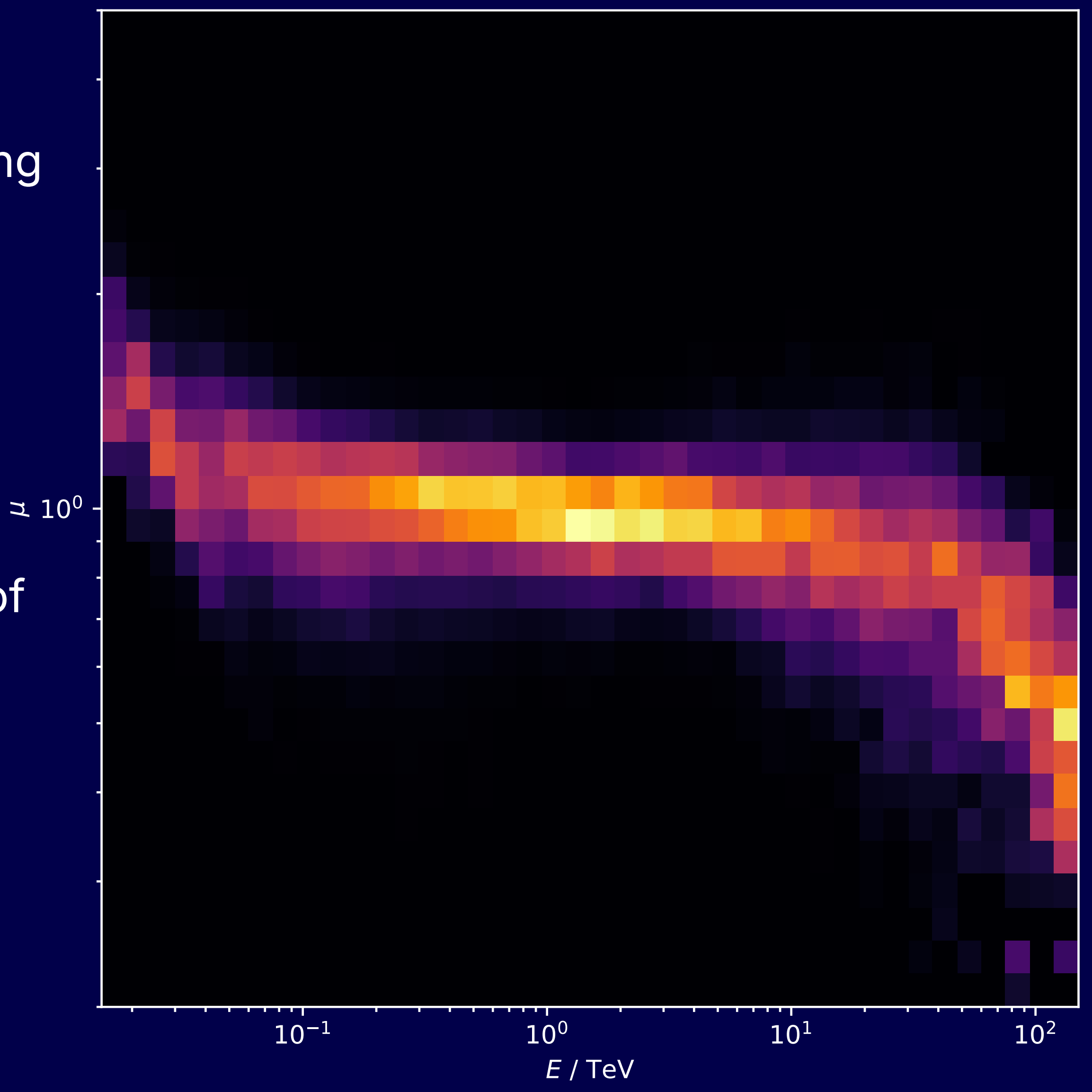


Energy Dispersion

- Energy dispersion is the probability density of observing an event of true energy E at a reconstructed energy $\hat{E}$
- To reduce sparseness of the resulting matrix, GADF defines the energy dispersion in relative terms using:

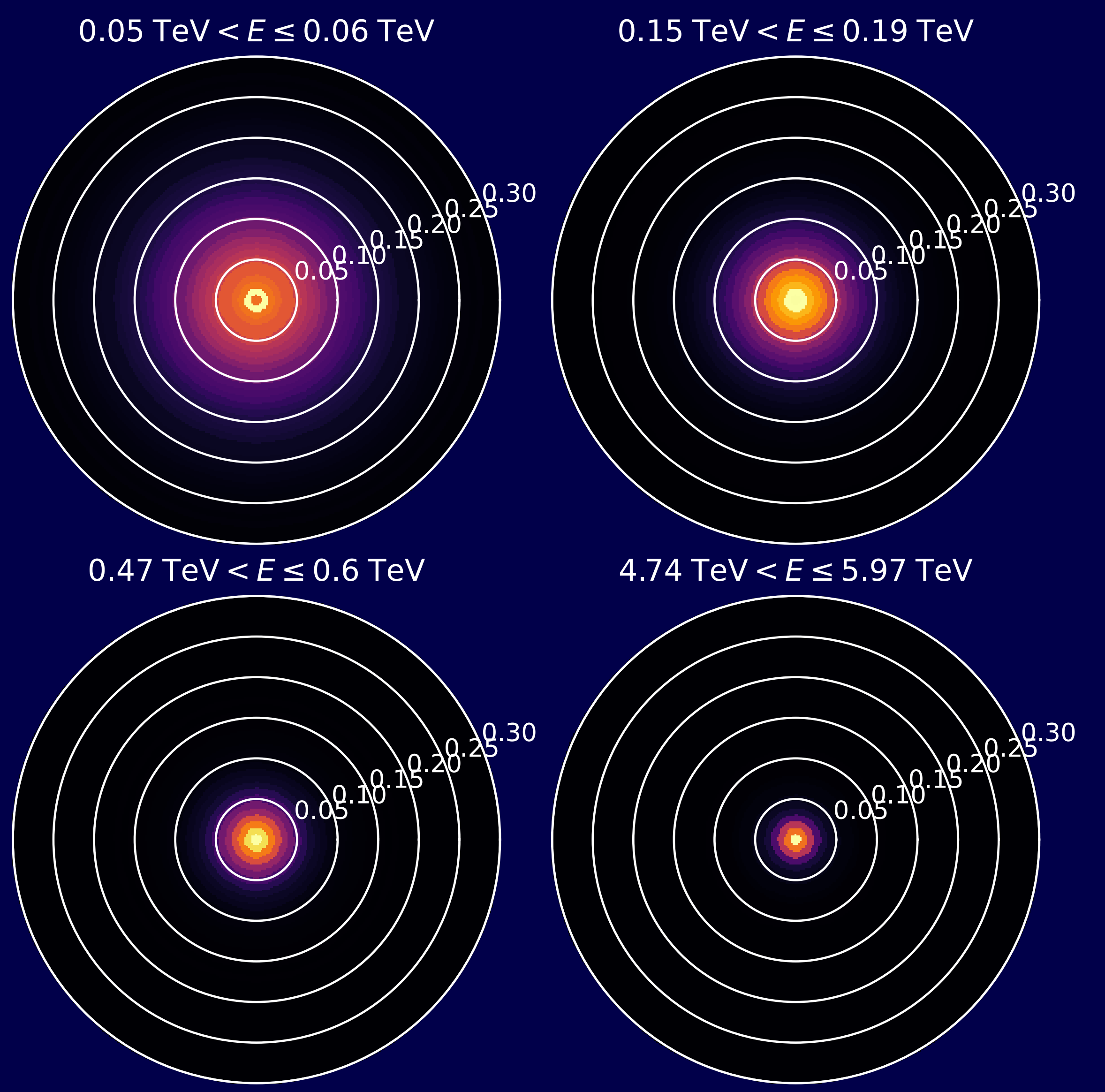
$$\mu = \frac{\hat{E}}{E}$$

- Again, the energy dispersion is stored as tabulated values of this PDF in bins of true energy and the field of view offset ⇒ radial symmetry



Point Spread Function

- The PSF is the probability density of observing an event with true coordinates α, δ at reconstructed coordinates $\hat{\alpha}, \hat{\delta}$
- GADF allows several descriptions
    - Tabulated values
    - Gaussian mixture model with three distributions
    - King distribution
- All three are expressed in the distance from the point source r
  ⇒ radial symmetry around the point source
- Only supported in bins of true energy and FoV offset
  ⇒ radial symmetry in the FoV



Background Rate

- The background rate can in principle be computed from electron and hadronic simulations, but:
  - Very hard to obtain realistic simulations
  - Much more uncertainty in hadronic interaction models
  - Extremely cost intensive
- Instead, the background model will be computed from observations
  - Excluding known gamma ray sources
  - Combining many observations
- Also not without challenges
  - Needs enough suitable observations
  - Especially hard for early science
  - Probably still needs to be re-fitted to the current observation conditions

Berge, David, S. Funk, and J. Hinton. "Background modelling in very-high-energy γ-ray astronomy." *Astronomy & Astrophysics* 466.3 (2007), doi:10.1051/0004-6361:20066674

Point-Like IRFs

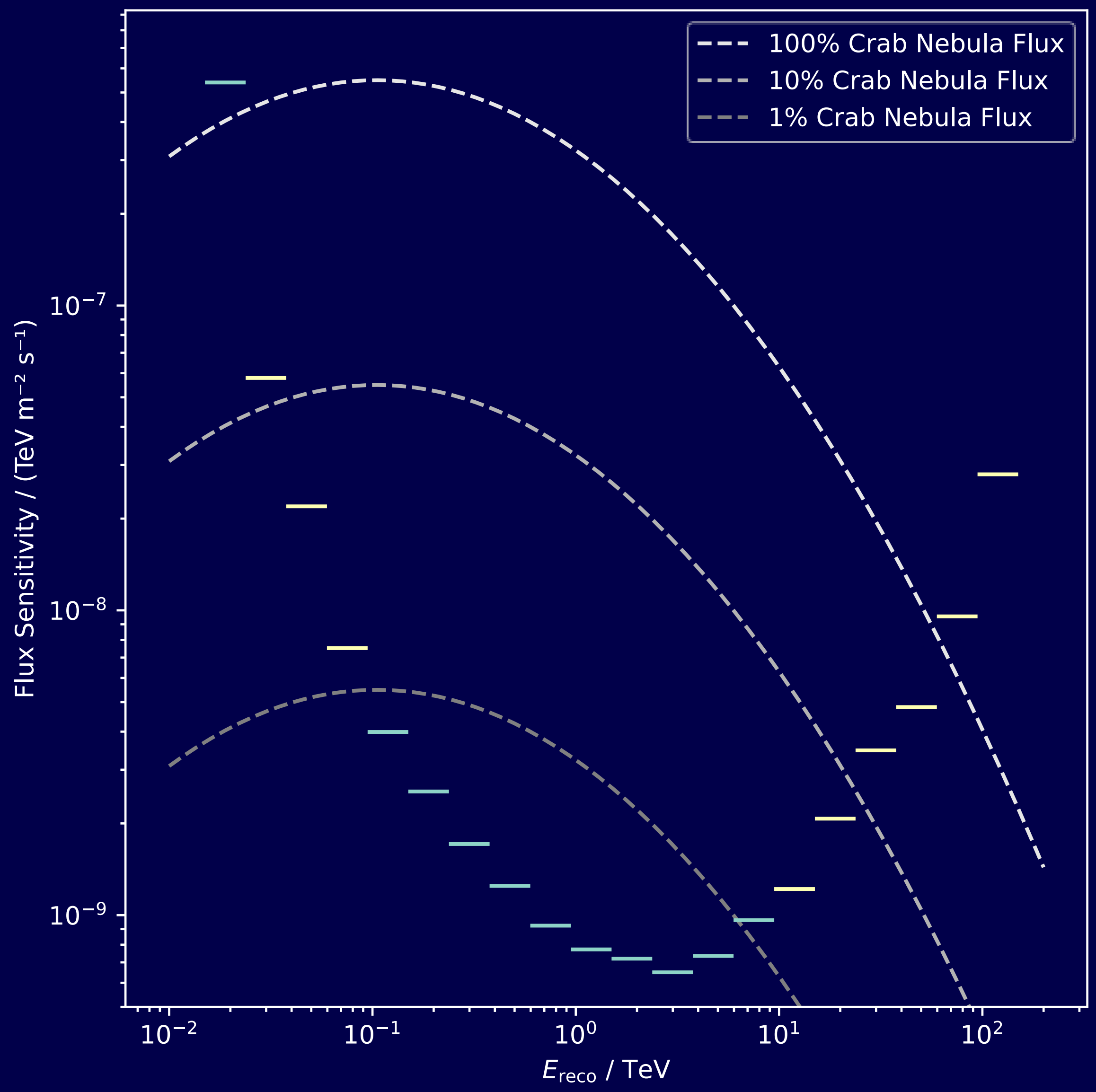
- For the use case of spectral analysis of a point source, the IRF can be simplified further
- We select events only in a circular region around the assumed source position
- The radius of this region can be energy dependent and should scale with the PSF
- The PSF is not needed anymore, the effective area is reduced by discarding the non-selected events
- This also reduces the systematic error due to the factorization, as we compute the energy dispersion only on the well-reconstructed events inside the source region
- The computed IRF is fixed to the radius of the signal region (“θ^2-cut”)
- Only computation of spectra possible, no sky maps or spectral cubes

$$\overbrace{R\left(\hat{E}\,\middle|E,\alpha,\delta,t\right)}^{\text{Instrument Response Function}} = \underbrace{A_{\mathrm{eff}}(E,\alpha,\delta,t)}_{\text{Effective Area}} \cdot \overbrace{M\left(\hat{E}\,\middle|E,\alpha,\delta,t\right)}^{\text{Energy Dispersion}}$$

Performance Benchmarks

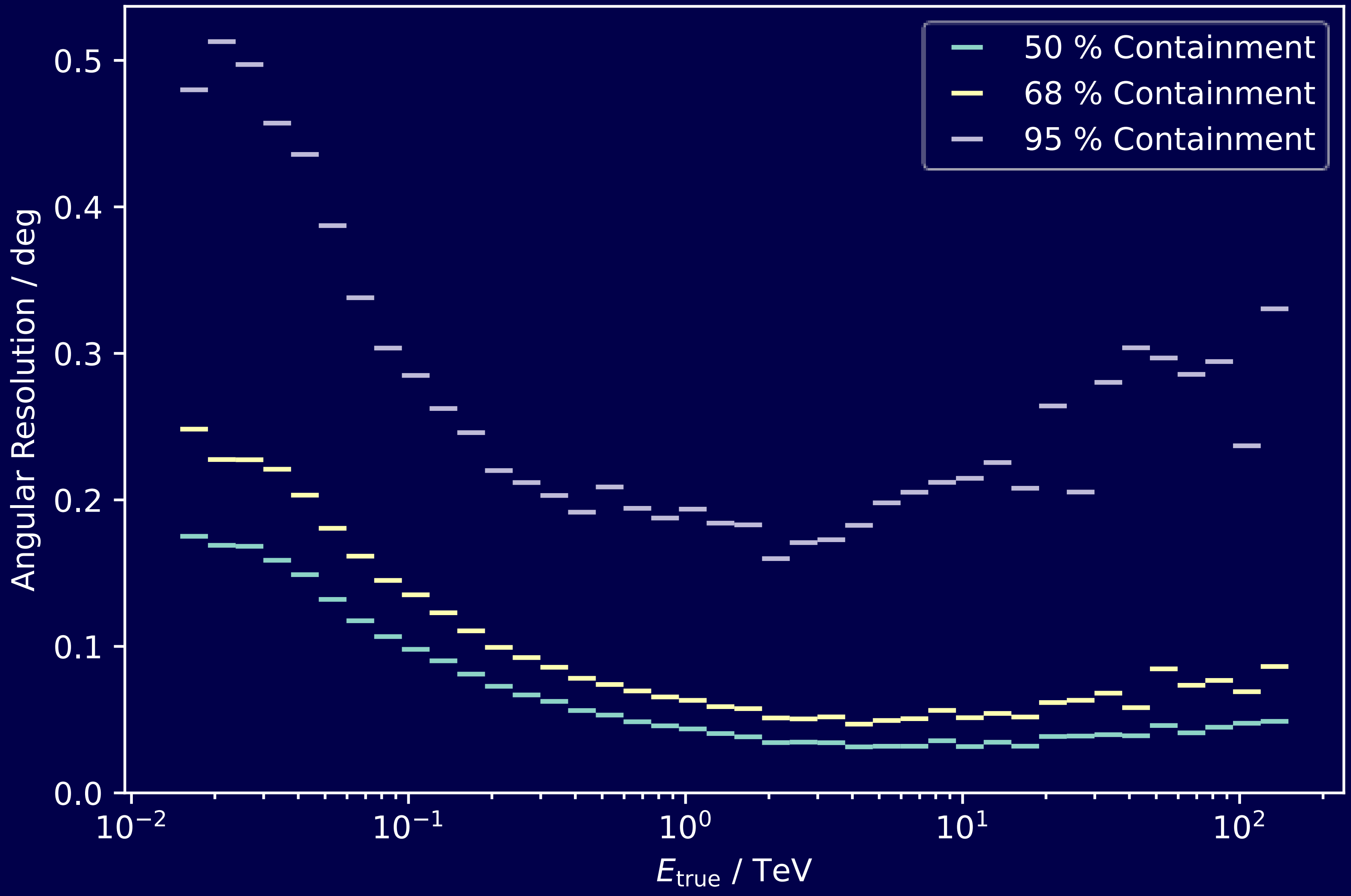
Flux Sensitivity

- Gives the lowest flux of a gamma-ray point source still “detectable” in a given observation time
- Detectable in CTAO is defined as:
  - Significance of at least 5σ according to the Li & Ma likelihood ratio test[1]
  - At least 10 signal excess events over the background rate
  - At least 5 % more excess than the background rate
  - Usually for 50h of observation time
- Some caveats:
  - Can only be defined vs. reconstructed energy
  - Same binning required to be comparable

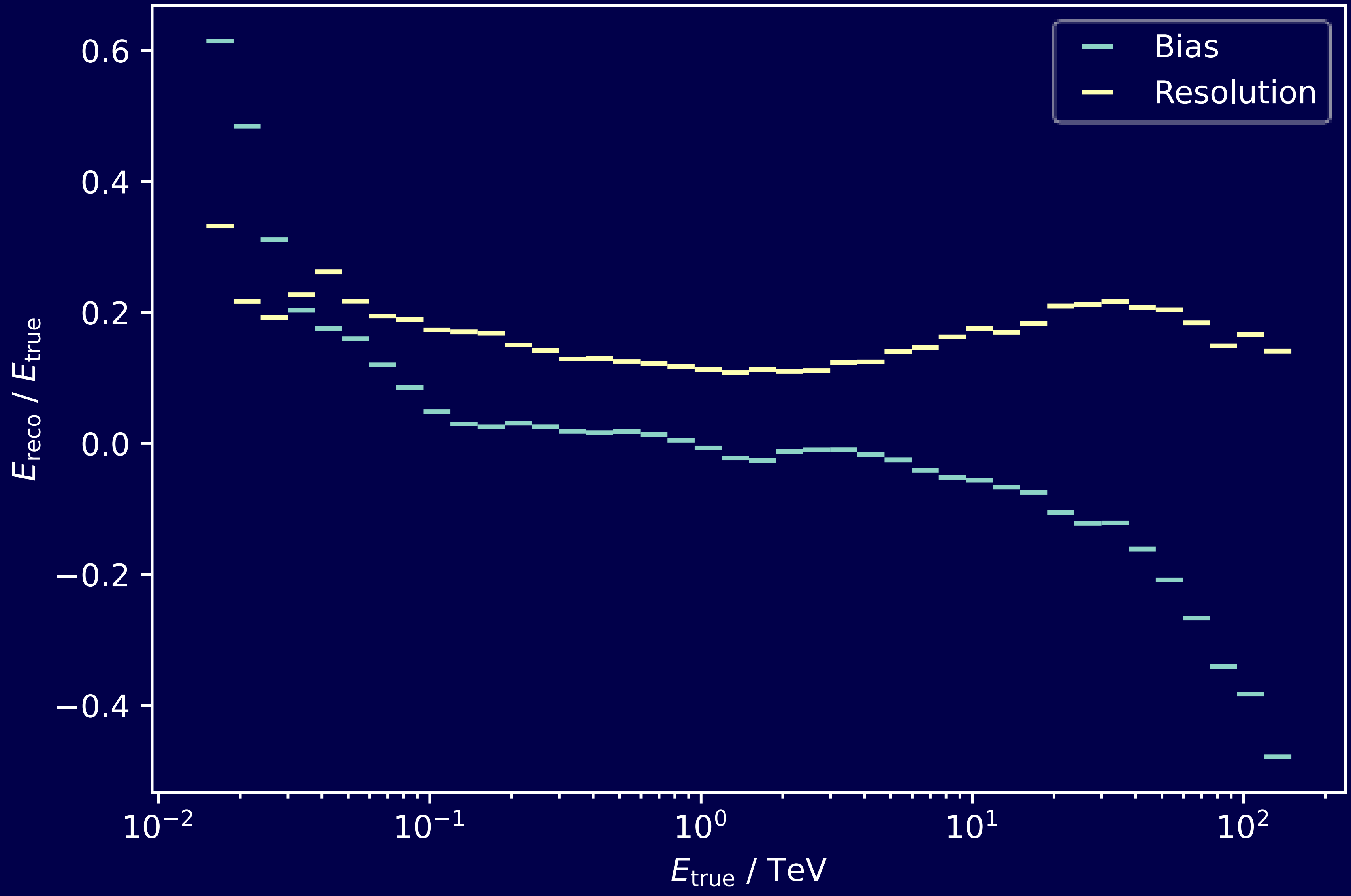


[1] Li, T. P., & Ma, Y. Q. Analysis methods for results in gamma-ray astronomy. *ApJ, vol. 272, 1983.* doi:10.1086/161295

Angular Resolution



Energy Bias & Resolution



Event Types

- Different science cases need different optimizations of the event selections
    - Pulsar analysis is mostly not concerned with background events due to timing information ⇒ can live with higher background contamination
    - Galactic plan has high source density ⇒ requires well reconstructed events to avoid source confusion

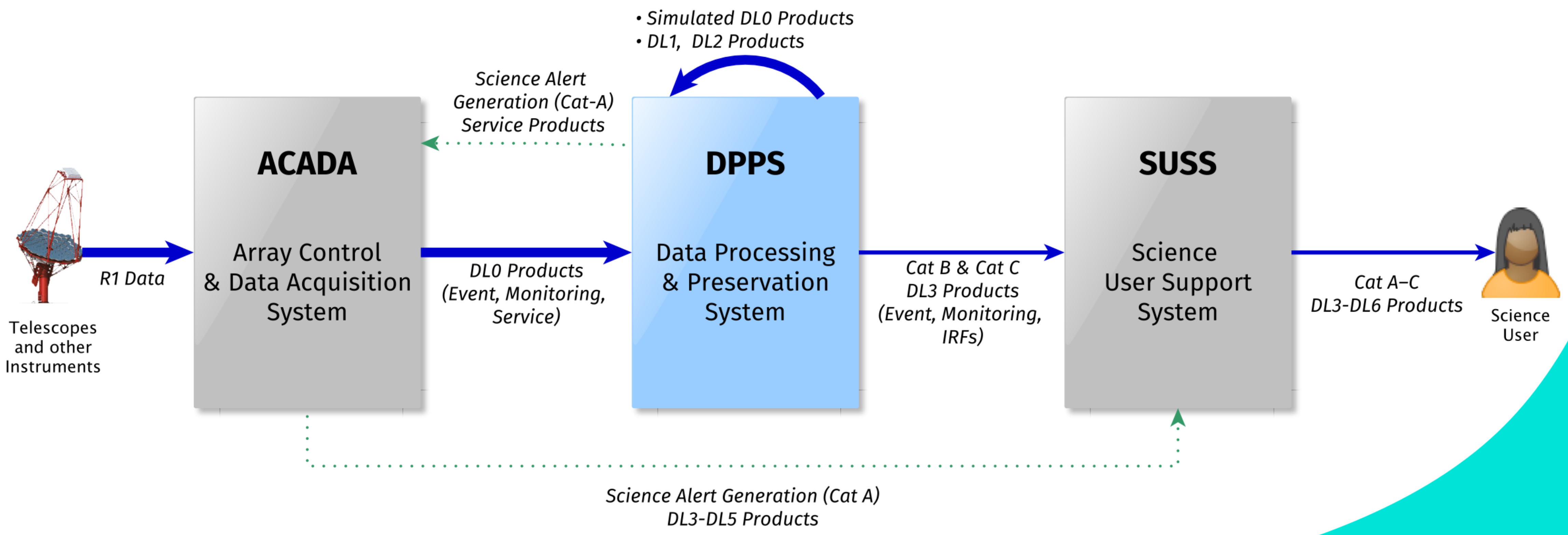
- But: DL3 IRFs are only valid for one set of event selection cuts
- How to give scientists the ability to chose?
- ⇒ Categorize events into "event types" and compute individual IRFs
- Scientists can choose appropriate types for their analysis, combine event types in joint likelihood analysis (See science tools lectures)
- Used extensively by Fermi LAT, in development for IACT / CTAO analysis, stay tuned
- Also reduces the systematic error introduced from the independence assumptions

Bernete, Juan et al. for the CTAO Consortium, "Performance update of an event-type based analysis for the Cherenkov Telescope Array", ICRC 2023, [doi:10.22323/1.444.0738](https://doi.org/10.22323/1.444.0738)

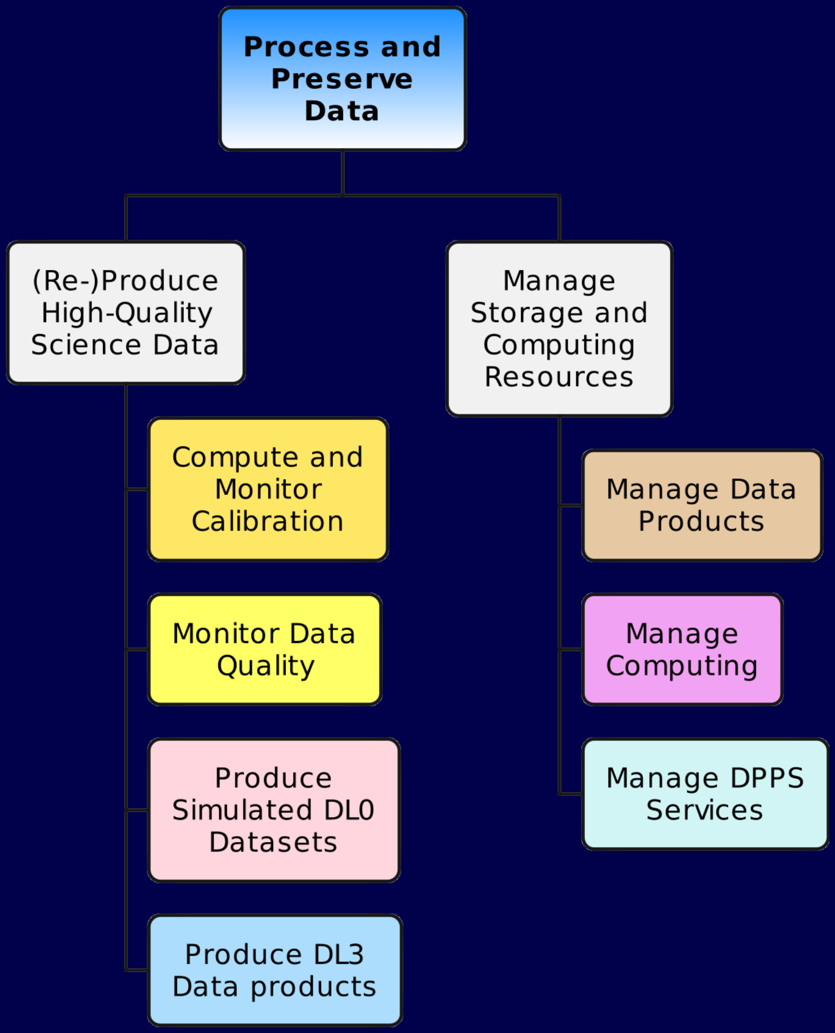
Habemus DL3

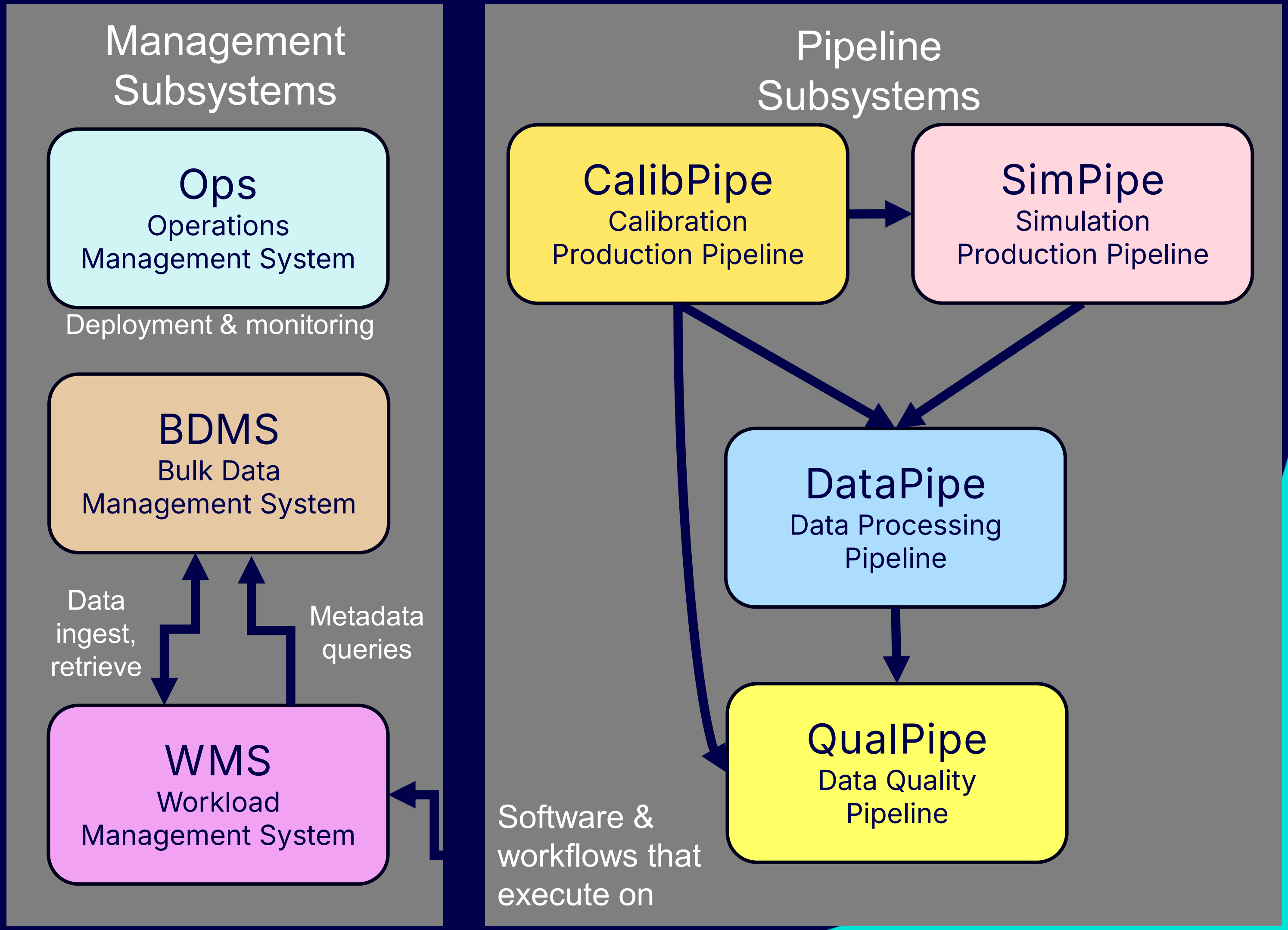
Software

DPPS: Data Processing and Preservation System



DPPS Subsystems





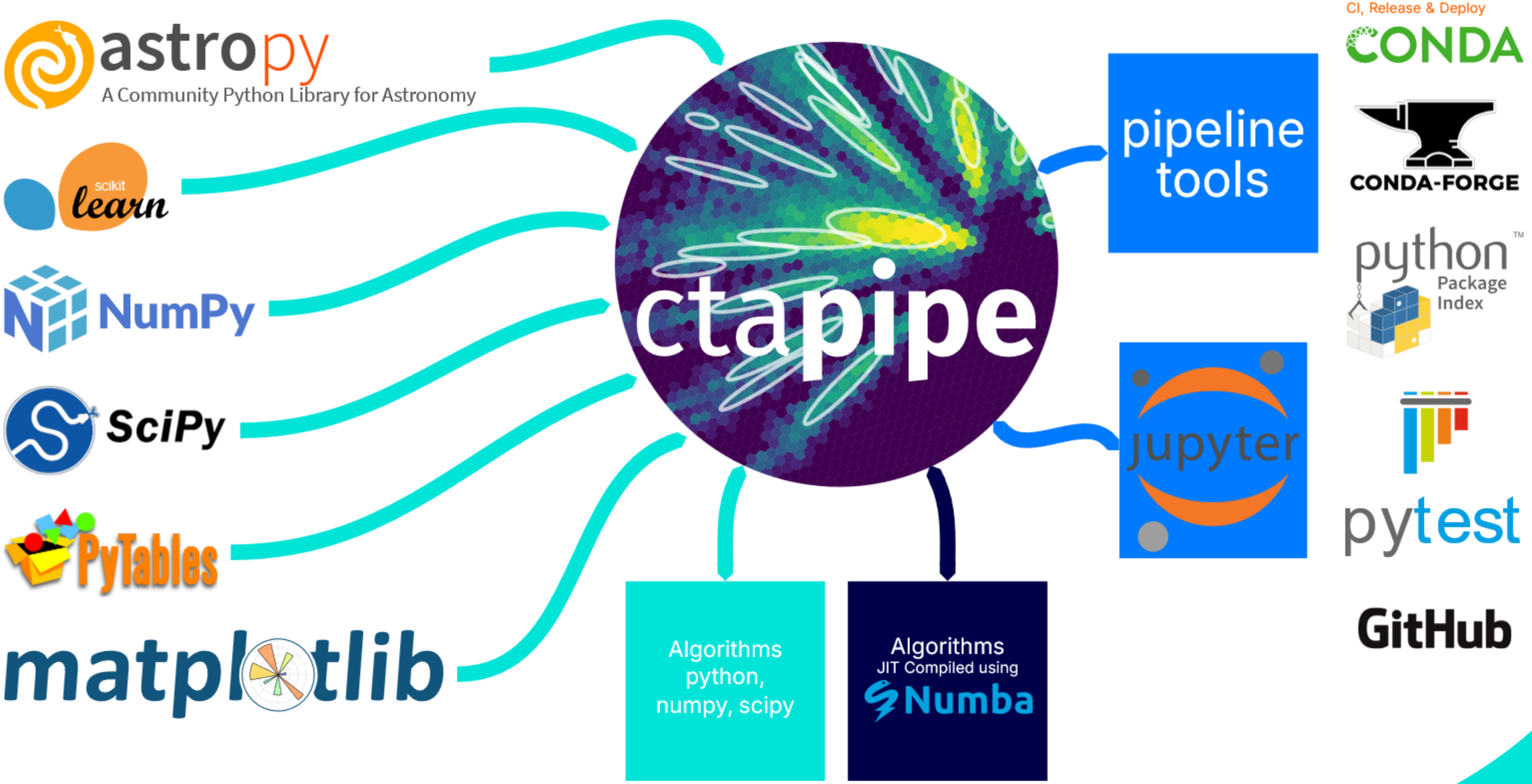
Main Software Components

- WMS is based on DIRAC
- BDMS is based on Rucio
- CalibPipe, DataPipe, QualPipe are all built on-top of ctapipe
- SimPipe uses CORSIKA 7, sim_telarray and ctapipe

RUCIO

ctapipe

DIRAC
THE INTERWARE



https://ctapipe.readthedocs.io/en/latest/

Conclusions

- The low-level analysis of IACT data is complex and has to deal with very high data volumes
  ⇒ we cannot expect astronomers to learn all of this and download terabytes of data
  ⇒ by default, we give out only high-level, reduced, reconstructed data with IRFs (DL3)
- One of the main challenges is the high number of configuration options, lack of good intermediate metrics and slow feedback loops:
  - What is the effect of changing {image extractor, cleaning levels, low-level cuts} on the science analysis of your source?
  - How does it affect systematic uncertainties?
- You need a working, robust analysis chain first before trying new, experimental things
- Physical interpretation is key: how do you adjust your simulations when your DNN does not work on actual observations?
- Calibration vs. Simulation is always a trade-off: they meet in the middle
- Parametrizing the IRFs and making sure the DL3 data we hand out are useful to many scientists are the current main conceptual challenges, contribute!

Questions?