JOINT INSTITUTE FOR NUCLEAR RESEARCH

Dzhelepov Laboratory of Nuclear Problems

FINAL REPORT ON THE START PROGRAMME

Parameter Analysis and Visualization for TAIGA-IACT: Gamma-Ray Events

Supervisor: Mr. Artur Borodin

Student: Shahd Muhammed Moghazy
Cairo University, Egypt

Participation Period: 14 September - 8 November, Summer Session 2025

Dubna, 2025

Contents

Abstract

This report presents a Python-based framework for analyzing experimental data from the TAIGA IACT telescopes. The work focuses on separating gamma-ray-like events from hadronic backgrounds through parameter studies, data visualization, and stereoscopic analysis using data from the 1st and 2nd telescopes.

Several automated functions were developed for data loading, cleaning, and event matching, ensuring efficient organization of plots and consistent visualization. This framework provides a structured basis for future analyses of gamma-ray-induced air shower events using TAIGA data.

Introduction

1.1 About TAIGA IACT

The TAIGA (Tunka Advanced Instrument for cosmic ray physics and Gamma Astronomy) Observatory is a complex, hybrid detector array for ground-based studies of cosmic rays and gamma-ray astronomy. It combines two main instruments: **TAIGA-IACT** (Imaging Atmospheric Cherenkov Telescopes) and **TAIGA- HiSCORE** (High Sensitivity Cosmic ORigin Explorer) is a wide-angle integrating air Cherenkov telescopes). Which work together to detect extensive air showers through complementary techniques.

(a) 1st Telescope (b) 4th Telescope

Figure 1: Two Telescopes from TAIGA-IACT System

The observatory also includes: **Tunka-133**, an integrating air Cherenkov array and scintillation detectors and **TAIGA-Grande** and **TAIGA-Muon**, which extend its sensitivity to a wider energy range and provide muon measurements for shower composition studies.

This report focuses on the IACT component, which detects Cherenkov light produced by extensive air showers (EAS) initiated when high-energy gamma rays or cosmic rays interact with Earth's atmosphere. Located in the **Tunka Valley, Siberia** (51.49° N, 103.04° E) at an elevation of about 675 m above sea level. The TAIGA Observatory aims to study very-high-energy (VHE) gamma rays in the **TeV** energy range.

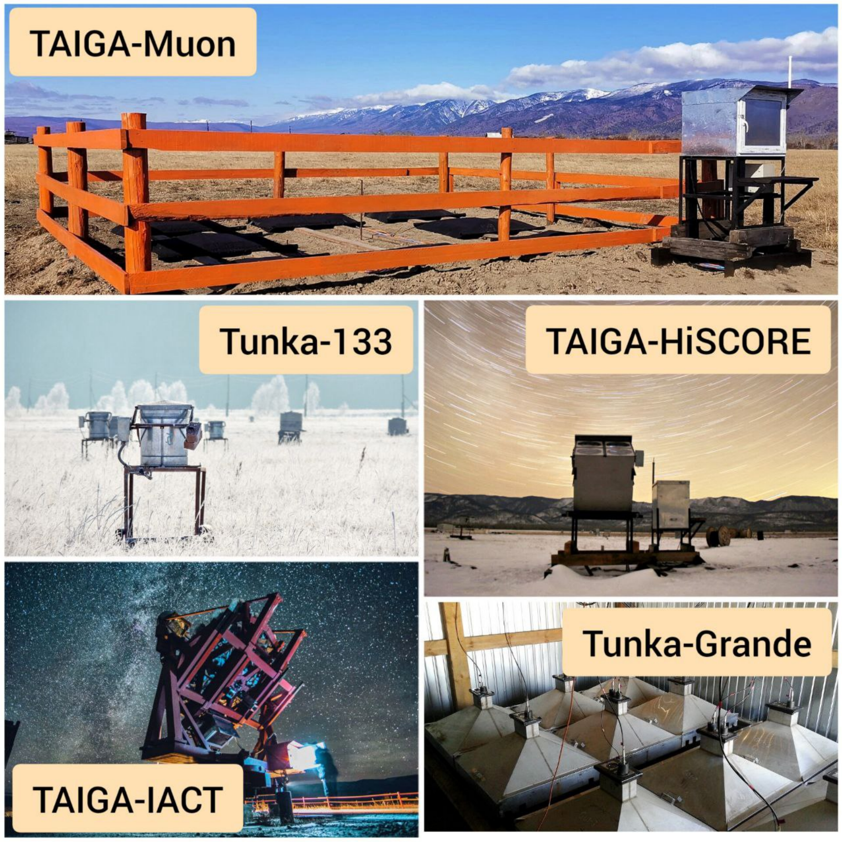

Figure 2: TAIGA Components

1.2 Experimental Parameters

In a previous report, I categorized all the experimental parameters to better understand their roles and meanings. This approach makes the data analysis process more straightforward.
Below is a summary of their categorization based on their roles:

Parameter	**Description**
`por`	Portion number.
`event_numb`	Used to track events number inside the portion.
`time`	Event timestamp in `hh:mm:ss:ms:`μ`s:ns`
`unix_time_ns`	Timestamp in **nanoseconds** since Unix epoch.
`delta_time`	Time difference from the previous event in the same telescope.
`error_deg`	Telescope pointing uncertainty in **degrees**.

Table 1: Event identification parameters.

Parameter	**Description**
`num_islands`	Number of isolated pixel clusters after cleaning.
`con_selected_island`	Fraction of light in brightest island.
`con2`	Fraction of light in two brightest pixels.
`numb_pix`	Number of pixels passing cleaning.

Table 2: Image cleaning parameters.

Parameter	Description
tel_az	Telescope azimuth in **degrees**.
tel_el	Telescope elevation in **degrees**.
tel_ra	Telescope right ascension in **degrees** .
tel_dec	Telescope declination in **degrees**.
source_az	Azimuth of the source in **degrees**.
source_el	Elevation of the source in **degrees**.
source_ra	Source right ascension in **degrees**.
source_dec	Source declination in **degrees**.
source_x, source_y	Source position projected on the camera plane in **degrees**.

Table 3: Telescope pointing parameters.

Parameter	Description
size	Total number of photoelectrons (brightness of shower) in **phe**.
Xc, Yc	Centroid coordinates of the shower in **degrees**.
length	Second-order moment along major axis.
width	Second-order moment along minor axis .
a_axis	Slope of ellipse major axis relative to camera grid (orientation) in **degrees**.
alpha1, alpha2	Angle between major axis and source/antisource in **degrees**.
dist1	Distance between centroid and source position in **degrees**.
dist2	Distance between centroid and antisource position in **degrees**.

Table 4: Hillas parameters describing shower images.

Parameter	Description
skewness_l, skewness_w	Third-order moments describing asymmetry along major and minor axes.
kurtosis_l, kurtosis_w	Fourth-order moments measuring peakiness or flatness compared to Gaussian.

Table 5: Higher-order image moments.

Parameter	Description
median_background	Median of pixel amplitudes that did not pass cleaning.
MAD_background	Median absolute deviation of pixel amplitudes that did not pass cleaning.

Table 6: Background estimation parameters.

Parameter	Description
CR400phe	Event count rate for size > 400 phe, averaged over 30 s.
CR_portion	Average count rate for all events in a portion, not restricted by size.

Table 7: Event-rate parameters.

Parameter	Description
`tracking`	Telescope actively tracking (1) or not (0).
`good`	Telescope pointing accuracy flag (1 = good, 0 = not good).
`star`	Presence of bright star among cleaned pixels.
`edge`	Edge pixels passed cleaning.
`weather_mark`	Observation conditions related to atmospheric transparency(10 = excellent, -1 = no data).
`alpha_c`	Camera rotation angle correction for orientation measurements.

Table 8: Observation and telescope status flags.

These parameters are used in different ways throughout the code:

- Some are used in **cuts** to select high-quality events.
- Some help **visualize** the Hillas parameters.
- Others track how data changes over **time and conditions**.

I'll use these categories throughout the code analysis.
The analysis presented here uses experimental data from the TAIGA IACT telescopes provided by the TAIGA Collaboration (unpublished)[1].

Code Explanation

This section explains the codes made for analyzing TAIGA IACT experimental data, focusing on **separating** gamma ray like events from hadronic events. First, I created helper functions to automatically choose the best (fitted) scale for plots. Then I made functions to show how parameters like change over time, with moon phases, and with count rates. Finally, I developed multiple functions to combine data from two different telescopes to study the same events from different angles (stereoscopic analysis).

The 3 codes created in order of explanation are:

1. **distribution_and_variation_functions**
2. **visualization_pixel_data**
3. **stereo_mode_functions**

2.1 Plotting Distribution and Variation Functions

Before discussing the distribution and variation functions, two helper functions (`plotting_histogram` and `param_scale`) are first defined to simplify the workflow and avoid code repetition.Both functions share the same logical structure and conditions but serve different purposes.

The `plotting_histogram` function produces one-dimensional histograms of the TAIGA-IACT experimental parameters. It automatically determines whether the parameter axes and bin widths should be displayed on a linear or logarithmic scale.
The `param_scale` function, on the other hand, contains only the scale-selection logic, allowing it to be integrated into the remaining plotting functions later on.

Scale selection logic (linear or logarithmic): Each function contains a "smart" scaling mechanism that automatically selects the most appropriate axis scale and binning based on the distribution of the data.
This ensures that parameters covering wide dynamic ranges are displayed clearly, while uniformly distributed data are represented without distortion.
There are two main conditional steps in determining the appropriate scale and binning scheme:

1. Axis scaling: The axis scale is selected automatically based on the parameter's data range:

- If the minimum value is less than or equal to zero (data_min $\leq$ 0), the scale must be **linear**, since logarithmic scales cannot represent zero or negative values.

- If the entire data range lies between 0 and 1 (0 $\leq$ data_min & data_max $\leq$ 1), a **linear** scale is preferred. Using a logarithmic scale in this range would distort the data, as dividing by values smaller than one would amplify them across multiple orders of magnitude, producing misleading results.

- If the data spans several orders of magnitude (e.g., range exceeds 10^3), a **logarithmic** scale is chosen to represent the data more effectively.

If none of these conditions are met, the data is considered to have a roughly uniform distribution, and a **linear** scale is applied.

2. Binning type: The binning strategy depends on the data properties. Two approaches are used:

1. **Square-root binning:** This is used for linearly scaled data and is still considered a linear binning scheme. The number of bins is calculated as:

$$N_{\text{bins, sqrt}} = 0.15 \times \sqrt{N_{\text{data}}}$$

   The factor 0.15[1] is applied as a scaling to reduce the number of bins, since the standard square-root rule ($N_{\text{bins}} = \sqrt{N_{\text{data}}}$) gives very large bin counts for very large datasets. This adjustment ensures that the resulting histograms remain computationally efficient and visually clear .

2. **Logarithmic binning:** For data distributed over several orders of magnitude, a logarithmic bin spacing is used. The number of bins is estimated using:

$$N_{\text{bins, log}} = 15 \times \log_{10}(N_{\text{data}})$$

   The factor 15 The factor helps reduce how quickly the number of bins grows with data size when using the logarithmic rule. This helps maintain a reasonable number of bins while still capturing the wide dynamic range of the data.

3. Dual-scale visualization: The function includes an optional keyword argument, `dual_scale`, which determines how the plots are generated:

- If `dual_scale = False`, the function returns two plots with the recommended axis scale, but comparing the **two binning strategies**. In the title of the plot indicates which binning type is recommended.

- If `dual_scale = True`, the function again returns two plots. This time with comparing the **two axis scales** and with the recommended binning .The title specifies which scale is recommended for the data.

[1]This value comes directly from the data and was determined through trial and error rather than being a constant.

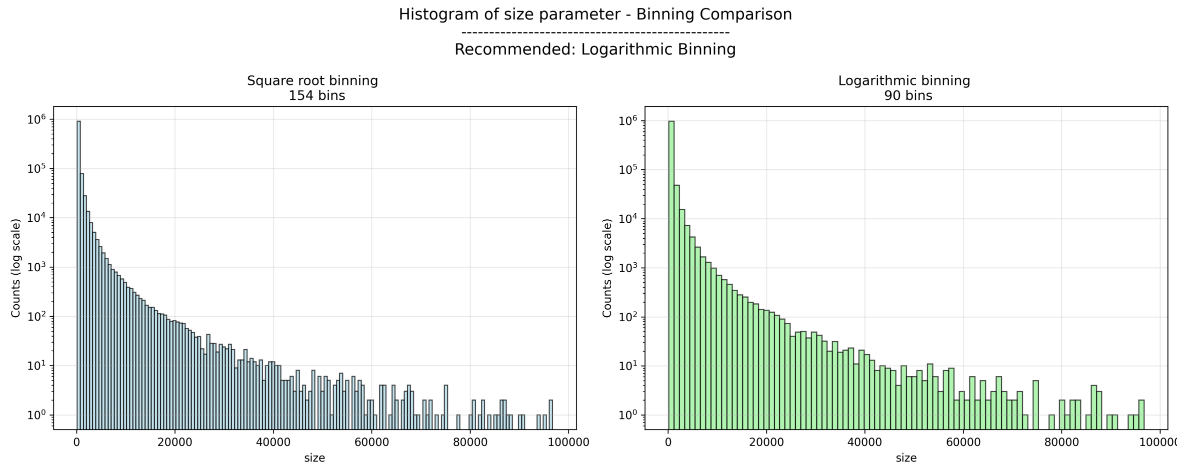

Figure 3: A Histogram Plot of `size` Parameter with Fitted Axis Scale and Binning Comparison.

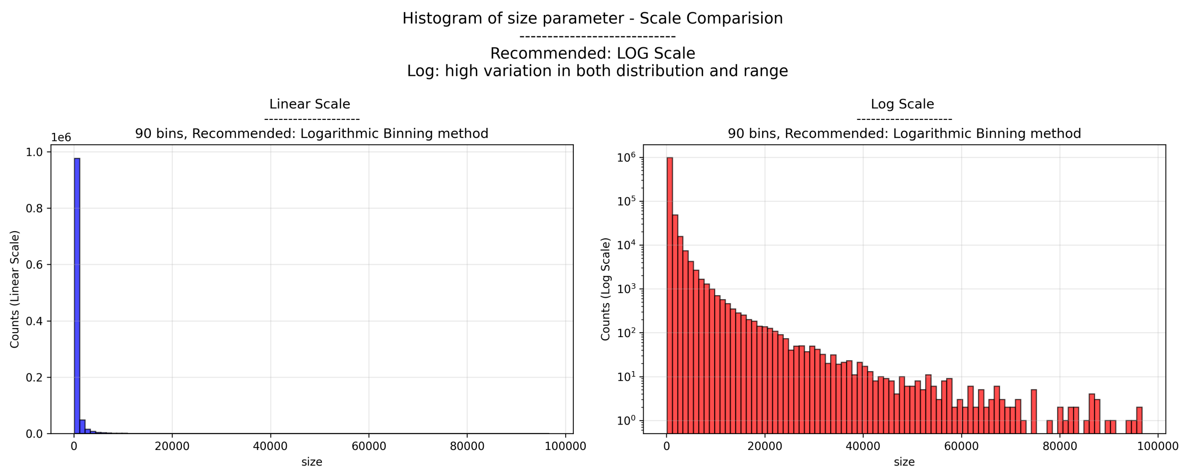

Figure 4: A Histogram Plot of `size` Parameter with Fitted Bin Width and Scale Comparison.

Figures 3 &4 shows an example output from the `plotting_histogram` function. Applied on the "`size`" parameter for data spanning nearly a year from the second TAIGA-IACT telescope.

This dual-plot structure helps visually confirm whether the automated scale selection aligns with the data's true distribution.

For the `param_scale` function, it returns a dictionary where each key is a column name, and each value is a tuple containing:

- The selected scaling axis type.
- A short description of why the scale is chosen.
- The binning value for both methods.
- The recommended bin count and method.
- The cleaned data used for plotting.

After defining these helper functions. **Six main plotting functions** were defined. These functions are organized into pairs: one for plotting the **distribution** and another for showing the **variation** (which includes the mean and standard deviation (σ)). Each pair is repeated for three different x-axis parameters: *time*, *lunar phase*, and *count rate*.

- The `unix_time_ns` parameter represents the event time in nanoseconds and is directly given in the data tables.
- The **lunar phase** is calculated from `unix_time_ns` to study periodicity and possible dependencies on moonlight.
- The **count rate** parameter (`CR_portion`) represents the average event count rate, calculated over 30-second intervals.

For the variation function, since both the mean and standard deviation are plotted (mean $\pm\sigma$), negative values can occur. Therefore, a logarithmic scale is **disabled**, and a linear scale is used instead to better represent the data, this will be shown below in some examples.

X-Axis as Time

The distribution function here generates **2D histograms** that illustrate how physical parameters *size*. The param_scale function is applied to ensure that each parameter is returned with the right axis and binning scales. A heat map (color map) is also included to display the event counts, with the color scale automatically set to linear or logarithmic based on the data distribution.

This produces plots with clearly labeled axes and scaled ranges. For example, using data from the second telescope over a one-year period (2020-2021) with the *size* parameter, the resulting plot reveals its temporal evolution as shown in Figures 5 and 6.

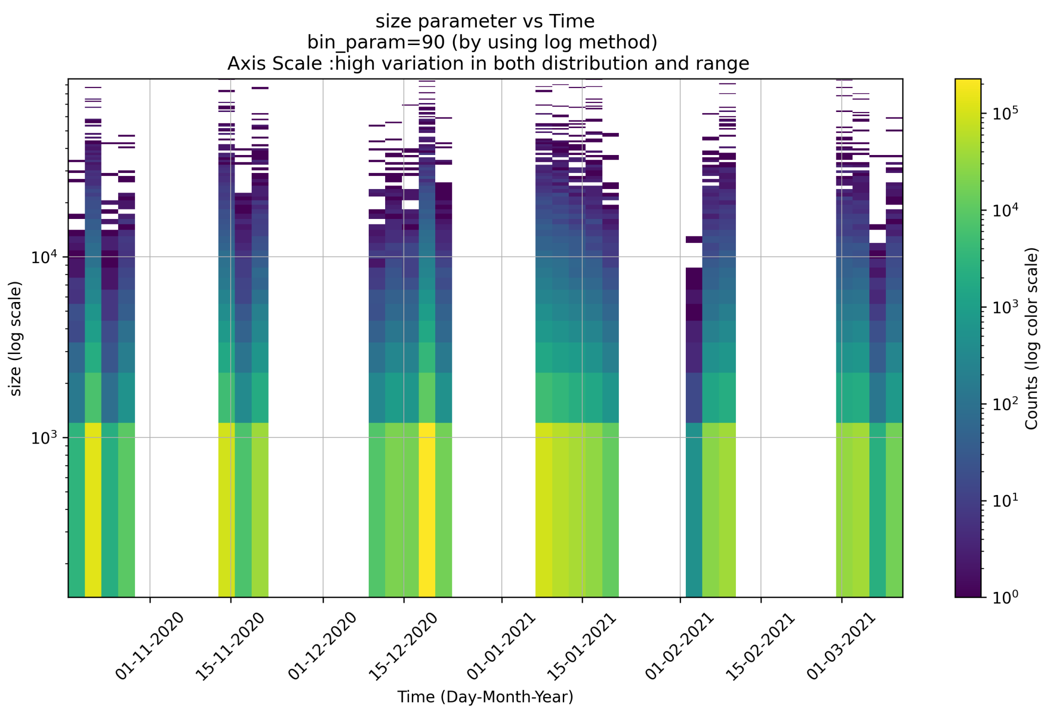

Figure 5: The Distribution of `size` Parameter vs Time

Note

The "`size`" parameter will be used as a representative example for all other defined functions to demonstrate the output and behavior of each plotting method/function.

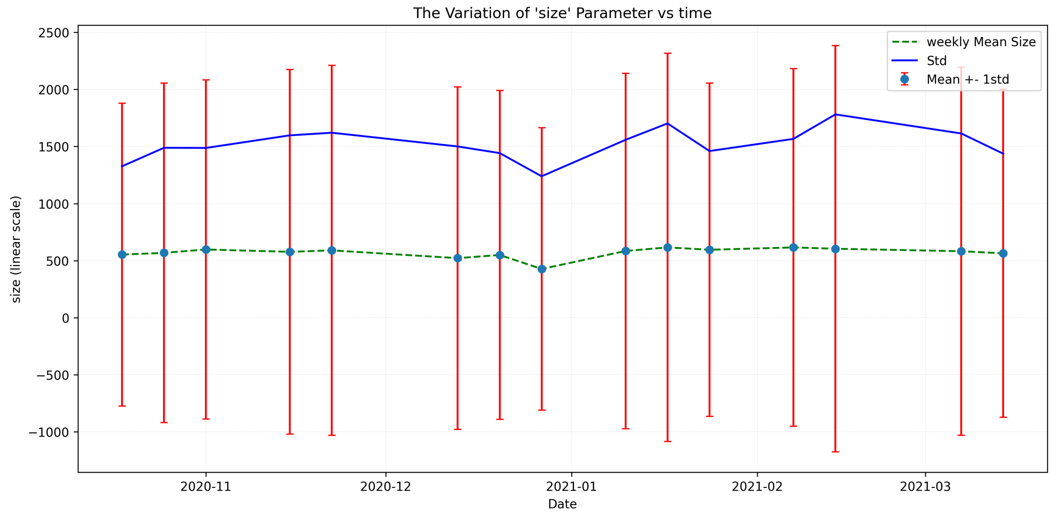

Figure 6: The Variation of `size` Parameter vs Time

X-Axis as Lunar Phase

The distribution function for the lunar phase also generates **2D histograms**, but here it shows how physical parameters vary throughout the **lunar cycle**.
The lunar phase is used instead of Gregorian time, as it better reveals periodic patterns and possible dependencies on moonlight conditions.
Similarly, the function automatically selects the appropriate color scale based on the data distribution. The bin width of the x-axis is set to 30, corresponding to the $\tilde{3}0$ lunar cycle, while the parameter scales and binning are obtained from `param_scale`.
This approach as seen in figures 7 and 8 gives a more physically meaningful interpretation of parameter variations in relation to the Moon's phase.

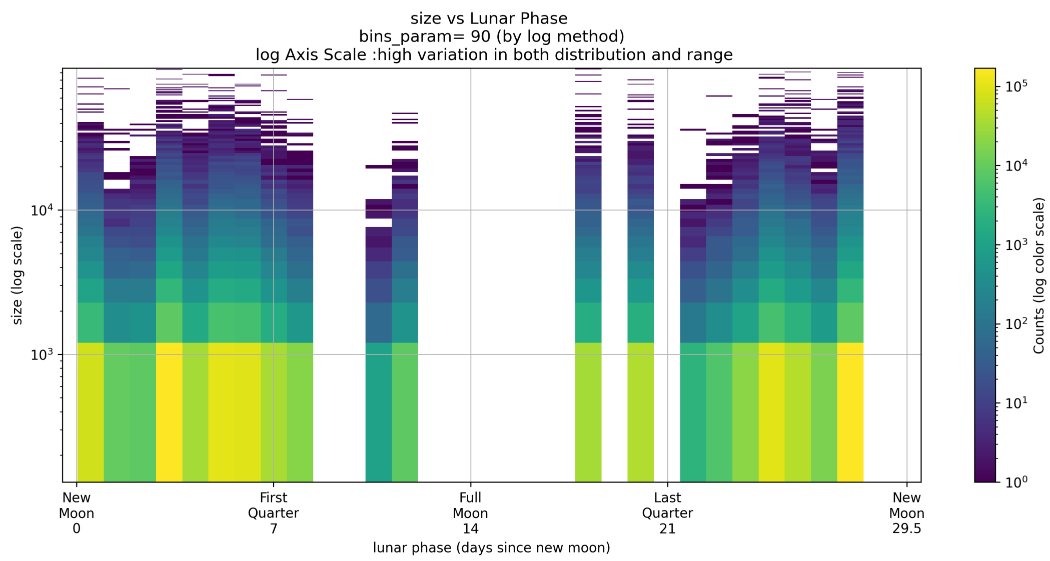

Figure 7: The Distribution of `size` Parameter vs Lunar Phase

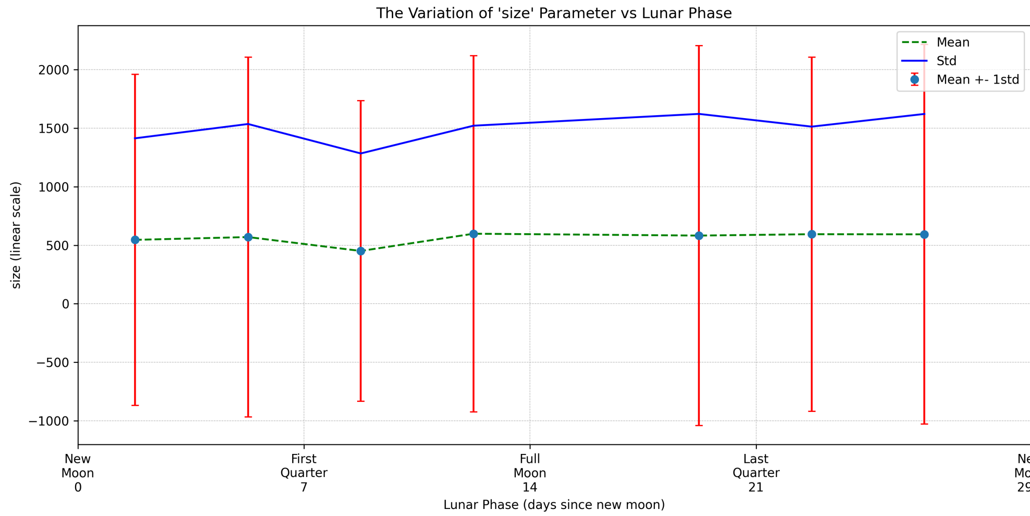

Figure 8: The Variation of `size` Parameter vs Lunar Phase

X-Axis as Count-Rate

Here, a similar method is followed, but unlike the lunar phase, which represents a temporal cycle, the count rate corresponds to how many events the telescope records per unit time, allowing us to study parameter trends as the detection rate fluctuates.

For the count rate parameter (`CR_portion`), an additional statistical method is implemented to determine the right binning. The bin width is estimated using the **Freedman–Diaconis** rule[2], implemented in the (`freedman_diaconis_calc`) function.

1. The count rate data are first filtered to remove missing values.

2. The inter-quartile range (IQR)[3] is computed, and the **Freedman–Diaconis rule** is applied to estimate the bin width:

$$\text{bin\_width} = \frac{2 * \text{IQR}}{n^{1/3}}$$

   Where n is the number of valid entries. **The number of bins** is then calculated as the data range divided by bin_width.

To check how well the method works, a comparison function (`compare_binning_methods`) was used to test several binning strategies, including:

- The square root rule.
- The Sturges' rule.
- The Freedman–Diaconis rule.

This method finds a bin count that captures the data accurately without too much noise, as shown in Figure 9. The results indicate that the **Freedman–Diaconis** method provides the most reliable and balanced binning, particularly for large datasets or those containing outliers.

[2]A data-driven approach that adapts to the distribution and sample size.

[3]The inter-quartile range represents the spread of the middle 50% of the data, calculated as the difference between the 75th percentile (Q3) and the 25th percentile (Q1). It is less sensitive to outliers than the standard deviation.

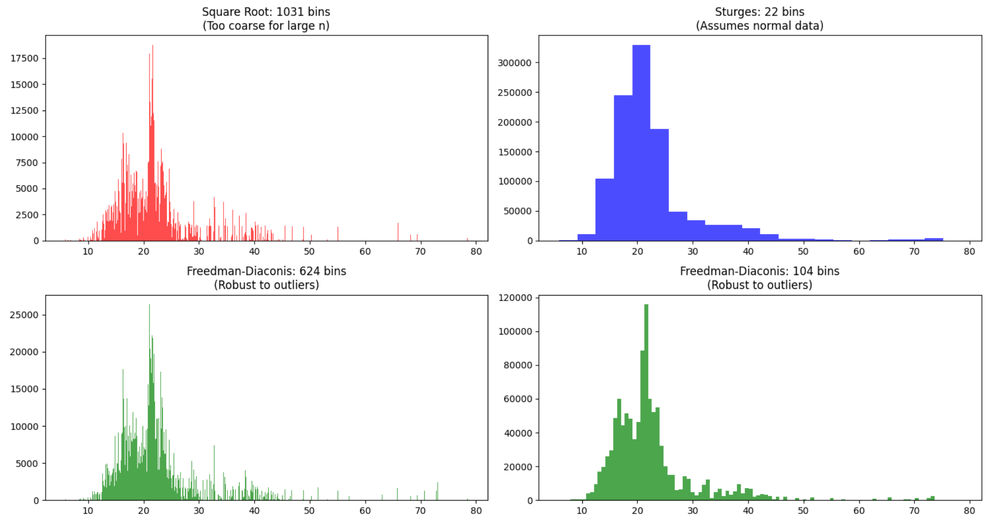

Figure 9: The same one-year dataset from the second telescope is used here to determine which method best suits the data.

Figures 10 and 11 show the resulting plots from the distribution and variation functions of count rate.

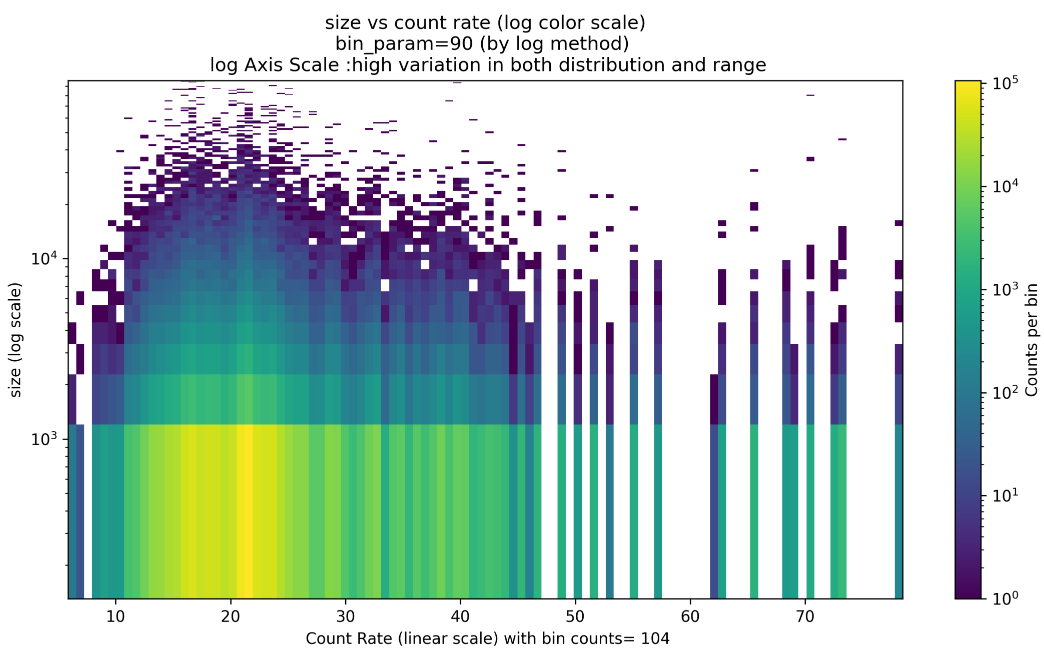

Figure 10: The Distribution of `size` Parameter vs Count Rate

Overall, these functions serve to visualize the main parameters and explore their variations across different observational and instrumental conditions. This helps identify correlations and trends that may reflect the detector's performance or atmospheric effects.

2.2 Event Visualization (Pixel and Hillas)

To better understand the recorded events, the next function visualizes both the experimentally derived Hillas parameters and the associated pixel data, providing a more detailed view of the event structure.

The experimental (Hillas) data and the pixel-level data share common columns as timestamp and size. The `size` parameter is used in the code to match events between the two datasets.

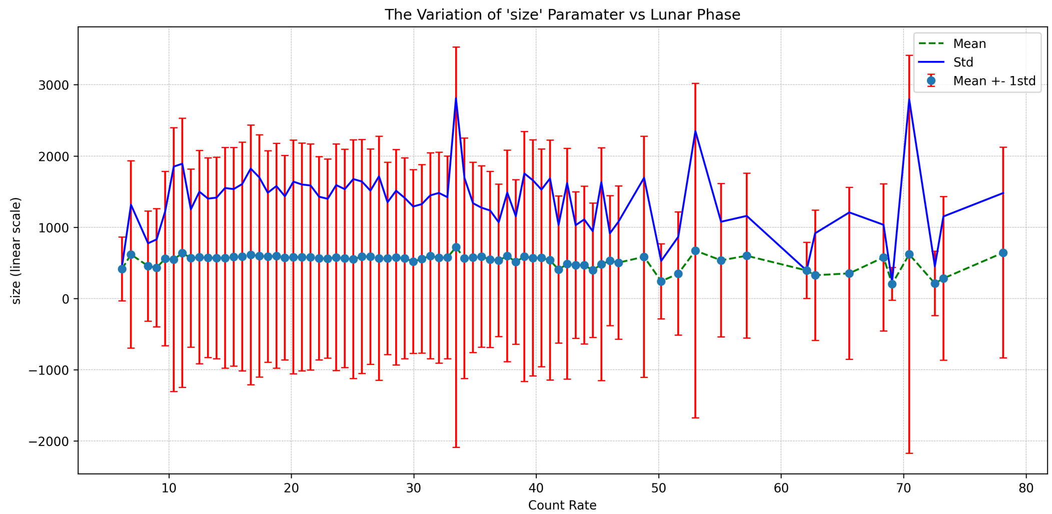

Figure 11: The Variation of `size` Parameter vs Count Rate

The pixel data initially comes as a text file and has to be converted to an Excel file for easier handling and merging.
It contains the following columns:

- `event_numb` (event id).
- `num_pixels` (the number of pixels that recorded that event)
- `timestamp`.
- `size`.
- `cluster_numb`.
- `pixel_numb`(the pixel number that recorded the signal)
- `x`, `y` (coordinates of each pixel in cm).
- `amplitude` (the amplitude or value of the signal recorded).

Event cleaning in this function follows a procedure similar to the previous analysis, but with additional criteria to remove events **near the edge** of the camera and keeping those with small alpha angles, ensuring that the visualized events are likely gamma-induced showers.

The function `visualize_all_events_per_one_file` takes these pixel data and the corresponding Hillas parameters to generate four plots combined in a single figure for each event:

- **Plot 1:** The camera field of view showing all pixel signals, the calculated centroid, the source position, and the reconstructed shower axis.
- **Plot 2:** A real sky map at the time of the event, displaying bright stars, telescope pointing, source position, and pixel signals.

- **Plot 3:** The Hillas reconstruction, including the centroid, ellipse representing the event shape, major and minor axes, and orientation relative to the source.
- **Plot 4:** A combined view overlaying pixel-level data and the Hillas parameters, highlighting the correspondence between raw signals and the reconstructed event.

This visualization shows the structure of individual events and how the experimental measurements match the pixel data, giving insight into the event reconstruction. An example is shown in Figure 12.

As we know, the `size` parameter is defined as a weighted mean, this effect becomes evident when the pixels are plotted. The statistical Hillas parameters (such as the ellipse shape) tend to cluster around pixels with higher amplitudes, emphasizing regions of stronger signal.

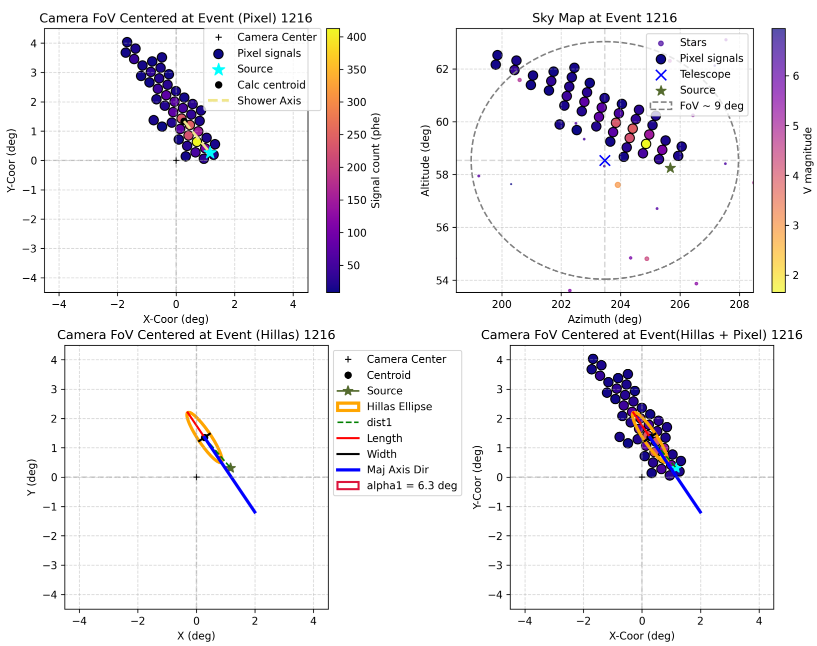

Figure 12: An Actual Event Visualization

Note

The pixel data in the sky map plot appears stretched because the sky projection is spherical, while in the camera-plane plots the data is more compact due to the planar projection.

2.3 Telescope Data Matching (Stereo Mode)

After defining all the previous functions for individual telescope data.
The next step is to connect and merge[4] the datasets from **Telescope 1** and **Telescope 2**.
This stage focuses on identifying air shower events that were recorded simultaneously by both telescopes between 2020 and 2021, enabling **stereoscopic analysis** and improved **gamma-ray event identification**.

[4] Although the merging method is general and can be applied to any combination of telescopes, for the purpose of this analysis it was implemented for **Telescope 1** and **Telescope 2**.

Data Characteristics

- The dataset contains a mix of gamma-ray and cosmic-ray (hadronic) air showers.
- Quality cuts are applied to enhance gamma-like events.
- Raw timestamps include only time components (hh:mm:ss:ms:μs:ns) without explicit date information. Therefore, dates are extracted from filenames.

Pipeline

The analysis pipeline proceeds as follows:

1. **Data Cleaning**: Apply event selection cuts to maintain high-quality gamma-ray candidates.
2. **Date Extraction**: Extract observation dates from filenames.
3. **Event Matching**: Combine data from both telescopes to identify the same air shower events. This is done in two steps:
   - *First pass*: Events are matched using exact date and timestamp up to millisecond precision, with any differences occurring at the microsecond scale.
   - *Second pass*: Perform a nearest-neighbor search within a 3 μs window to ensure physical consistency.
4. **Stereo Analysis**: Merge the matched events for stereoscopic studies and improved direction reconstruction.

Assumption: True gamma-ray showers should appear in both telescopes within a 3 μs delay, ensuring that the same event is recorded by both instruments.

Supporting Functions

Several helper functions are defined to simplify the generation and analysis of stereo events:

- Save output to a text file for inspection of event values, dates, and matched entries.
- Extract the observation date from filenames and return it as a `datetime` object.
- Load and merge all CSV files for a single telescope into one `DataFrame`.
- Match events between telescopes using timestamp precision up to microseconds.
- Count and record the number of successfully matched stereo events.

An adapted version of the previously defined `histogram-vs-time` plotting function is then used to visualize the merged telescope datasets over time and examine the temporal distribution of stereo events, as shown in Figures 13 and 14.

By also applying the variation function in an adjusted manner to handle data from both telescopes, Figure 15 below shows an example of the resulting stereo-mode analysis.

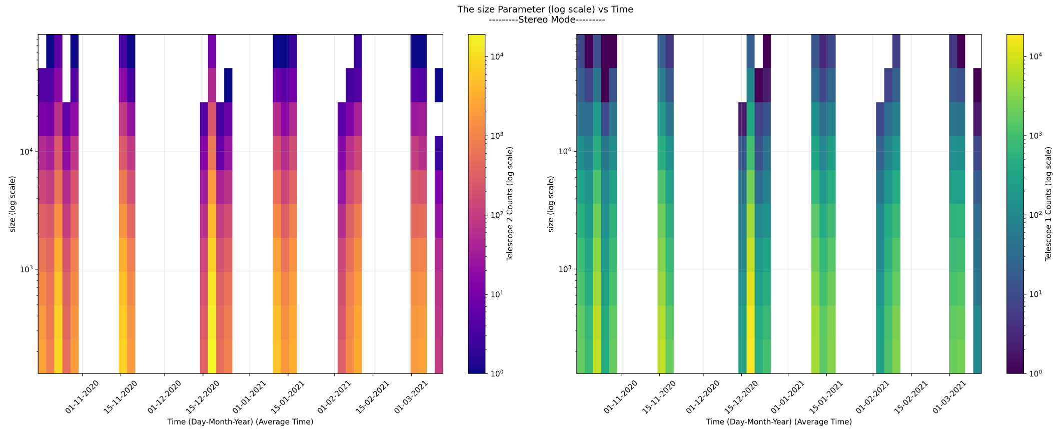

Figure 13: The `size` Parameter Distribution from 1st and 2nd Telescopes.

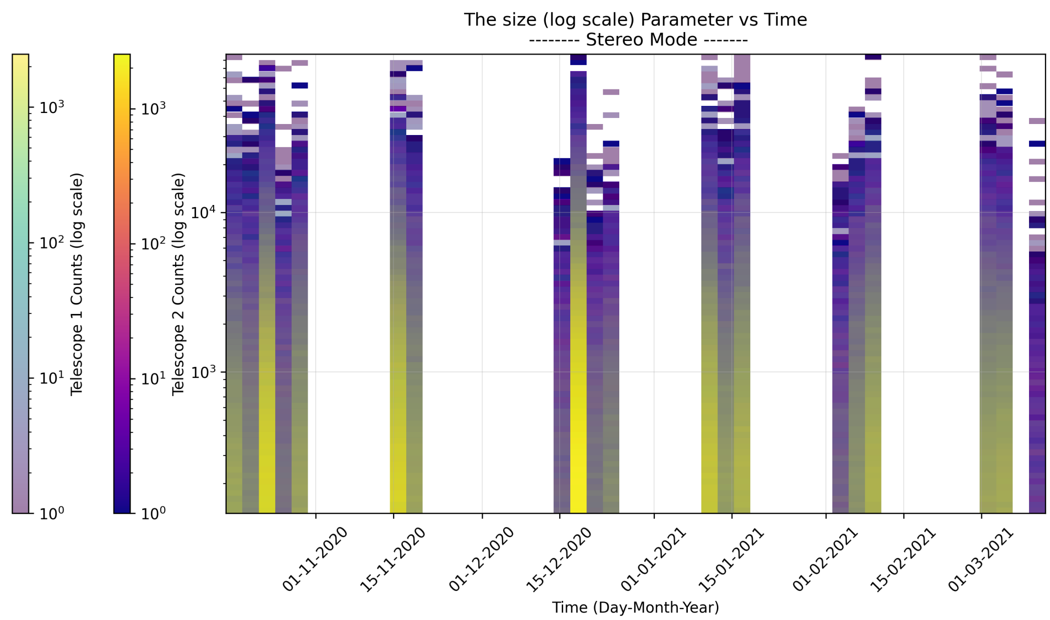

Figure 14: The Same Data as in figure 13, But both telescopes are plotted with the same x-axis.

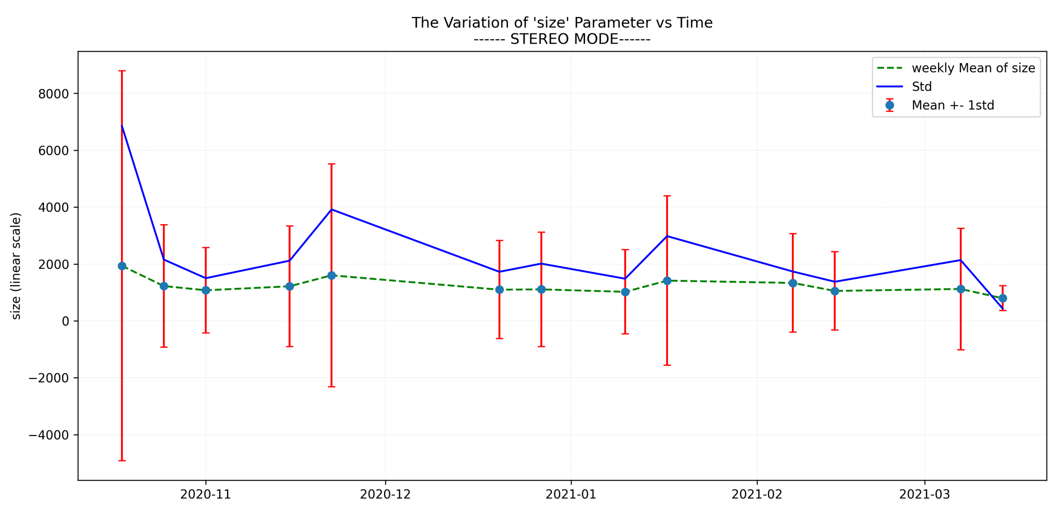

Figure 15: The Variation of `size` Parameter from 1st and 2nd Telescope Data.

In future versions, these functions could be organized into classes to make the code more reusable, and easier to maintain. This would help group telescope specific data together and make the analysis process smoother.

Conclusion

In this work, I developed and applied a set of functions to analyze experimental data from the TAIGA IACT 2nd telescope, and subsequently used data from the 1st telescope for stereoscopic analysis. The focus was on separating gamma-ray-like events from hadronic events, with methods implemented for plotting, parameter variation studies, and stereoscopic comparison between the two telescopes.

The workflow efficiently organizes plots into sub-folders, provides automated scaling for visualization, and enables direct comparison of events from the experimental data. This framework offers a structured approach for studying air shower events, particularly those induced by gamma rays, and can be extended to support more detailed analyses in future work.

References

[1] TAIGA Collaboration. *TAIGA IACT experimental data (unpublished)*. Provided by the TAIGA Collaboration. 2025.

[2] A. M. Hillas. "Cerenkov Light Images of EAS Produced by Primary Gamma Rays and by Nuclei". In: *19th International Cosmic Ray Conference (ICRC19), Volume 3*. Ed. by F. C. Jones. Vol. 3. International Cosmic Ray Conference. Aug. 1985, p. 445.