Image Noise Characterization for the Cygno Detector

Guilherme Lopes, Igor Abritta, Rafael Nóbrega

July 11, 2019

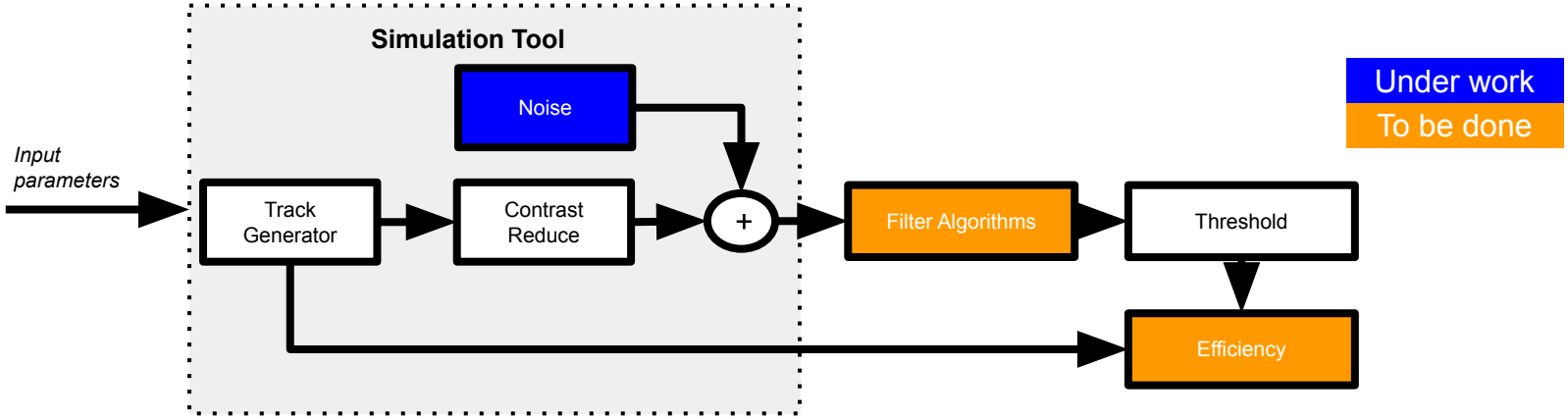
Summary

- ❑ Overview
- ❑ Noise characterization
- ❑ Noise Generation
- ❑ Next Steps

Overview

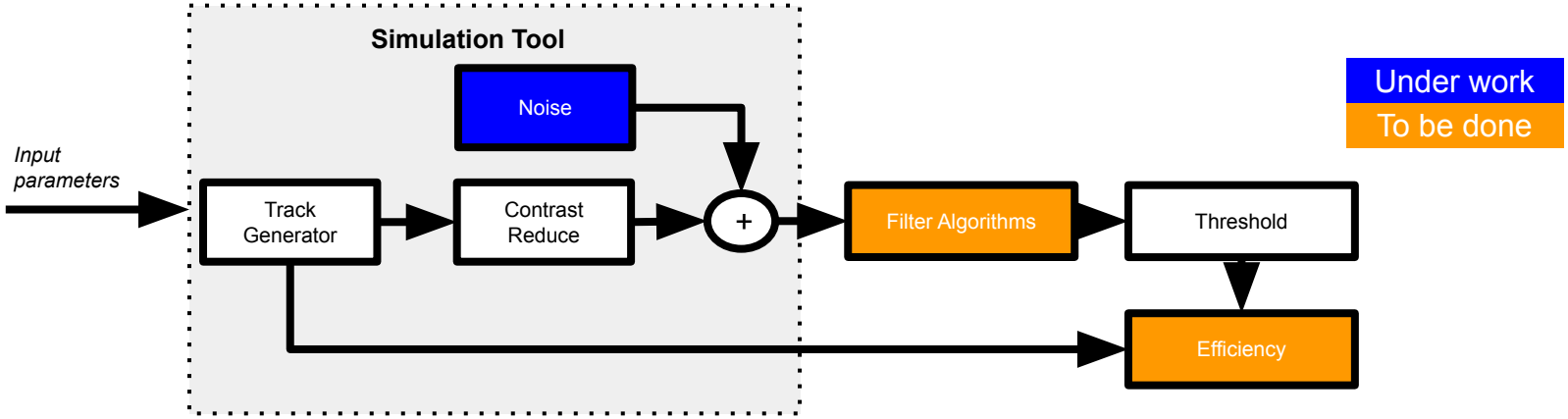
Last presentation

- At the last meeting we presented:
  - A simulation proposal based on Cygno images to be used for evaluation of filtering and clustering algorithms
  - Evaluation of the impact of classical filters when applied to the simulated images



Last presentation

- During the meeting the following next-steps were proposed:
  - Evaluate the possibility of generating noise for each pixel individually based on Cygno images;
  - Use energy estimation as a performance metric
  - Reanalyze filters performance (include new filters if necessary)



Cygno Noise Evaluation

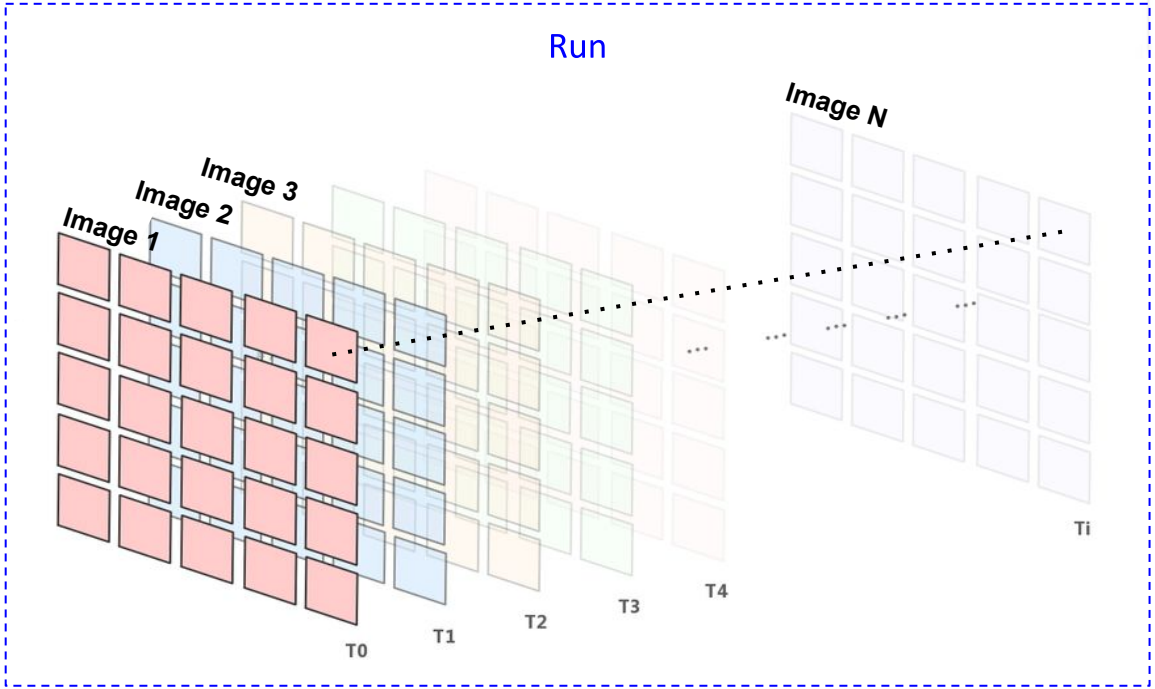
Used dataset: 817, 818, 819 and 820

Methodology

- What was done?
  - Measurement of the pixel-by-pixel mean and STD
  - Study of the noise intensity distribution
  - Spatial Correlation analysis
  - Implementation of a noise generation algorithm
- The following runs were analyzed:
  - 817, 818, 819 and 820

Pixel's pdf extraction

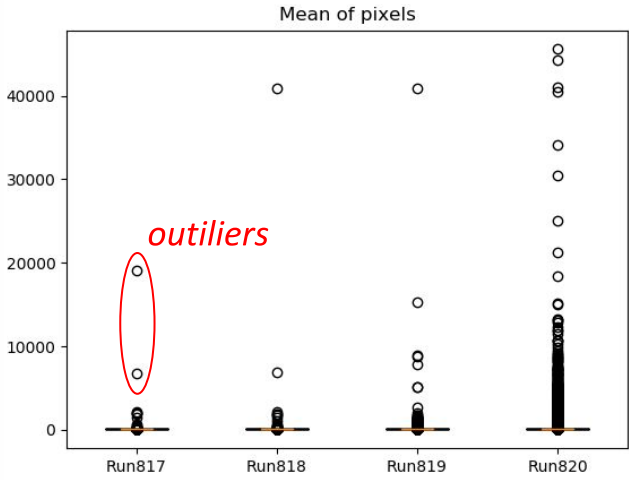
- There are 100 images into each run
- Then, each pixel can be represented by a pdf (estimated from its 100 samples)
- Simple statistical metrics are used to analyze the pixels characteristics.
  - Standard deviation
  - Mean
  - Median
  - Correlation



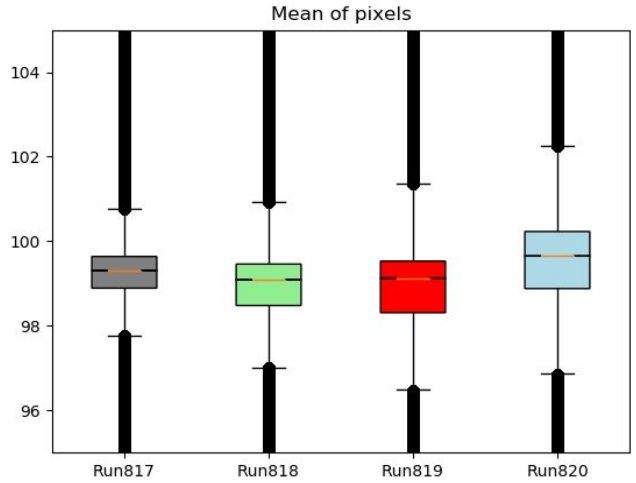
Pixel's pdf extraction

- Mean values (~median)

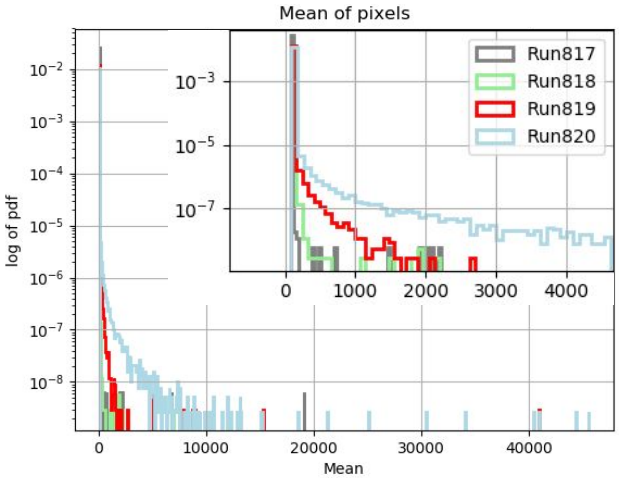
Boxplot



Zoom



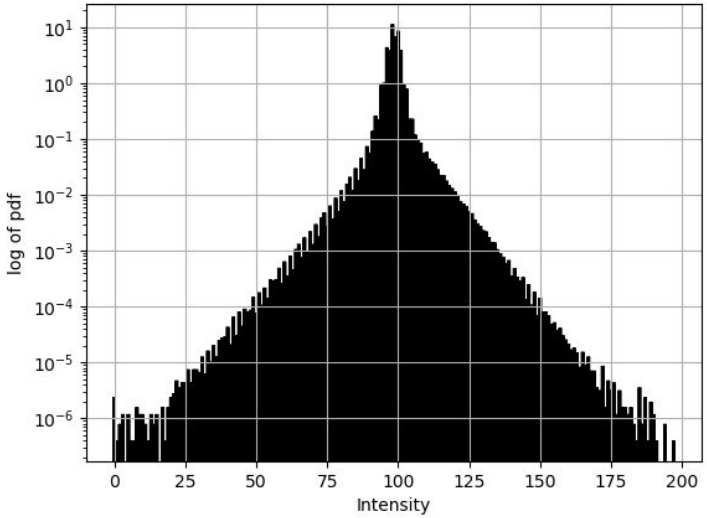
Histogram



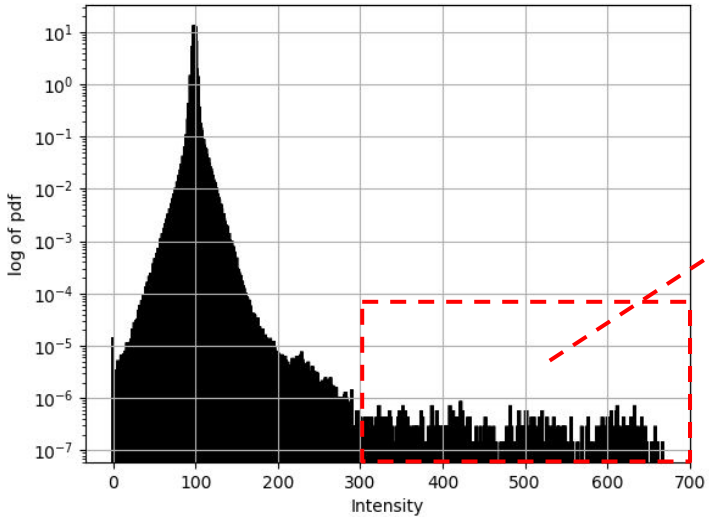
Pixel's pdf extraction

- Noise intensity distribution (Run 818)

Gathering pixels with **mean ≃ < 99**



Gathering pixels with **mean < 1000**

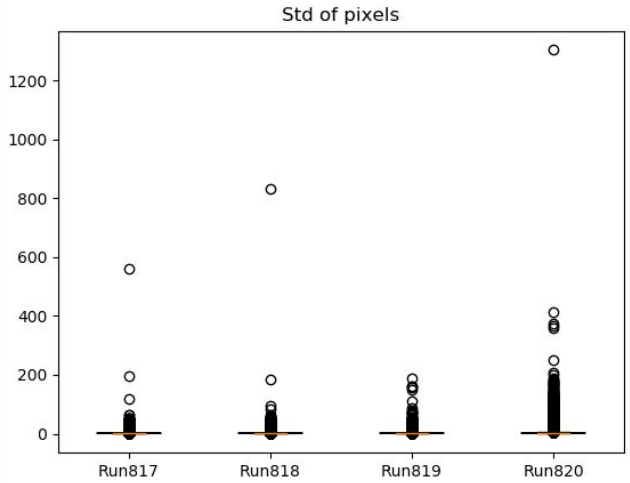


Caused by mean OUTLIERS

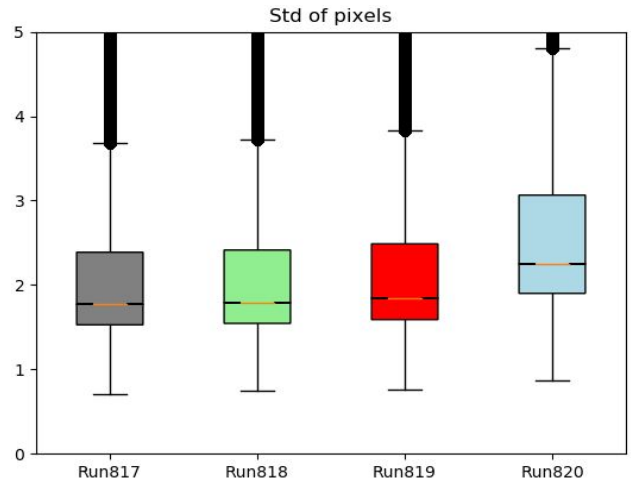
Pixel's pdf extraction

- Standard deviation values

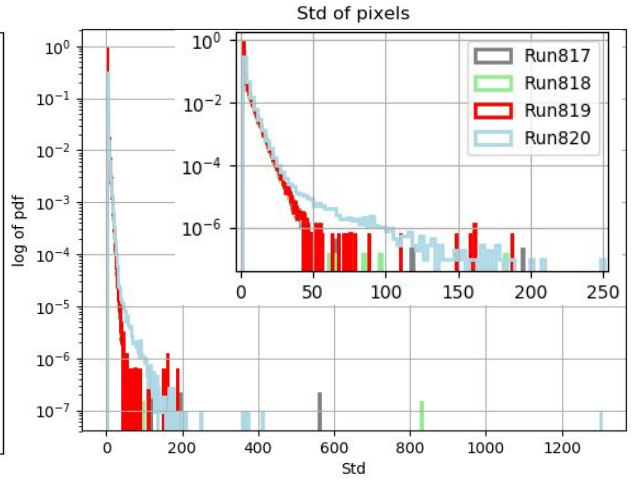
Boxplot



Zoom

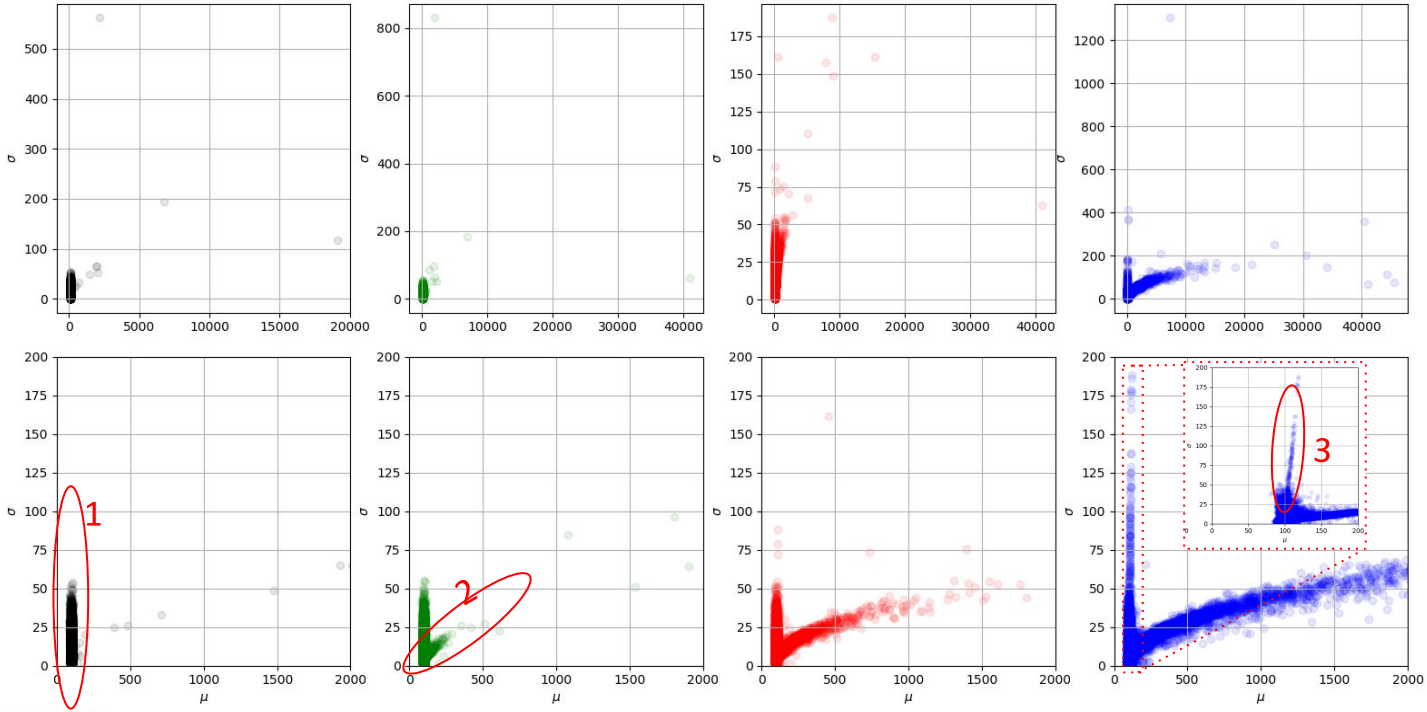


Histogram



Pixel's pdf extraction

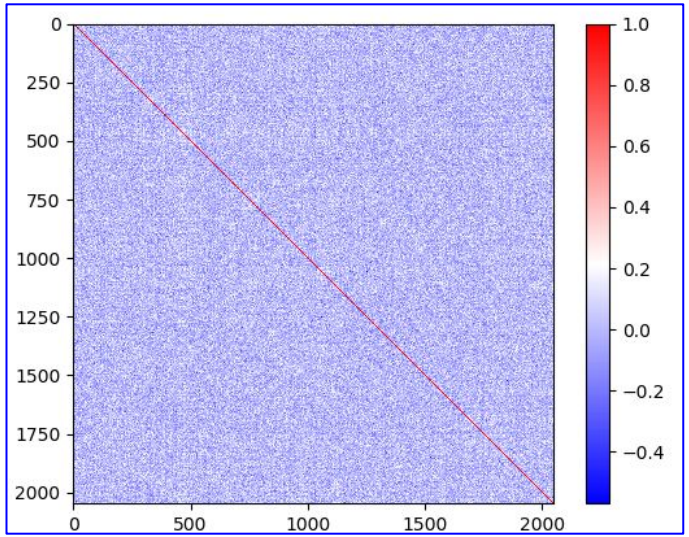
- MEAN versus STD → Two or three groups of pixels can be identified
  - Deeper investigation is needed to find out more



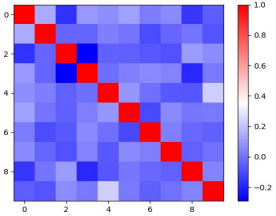
Pixel's pdf extraction

- Spatial correlation within a line
  - Each row will generate a 2048x2048 correlation matrix
  - At right the resulting matrix for the first row

Example for first row(run 818)



Zoom of a small sector



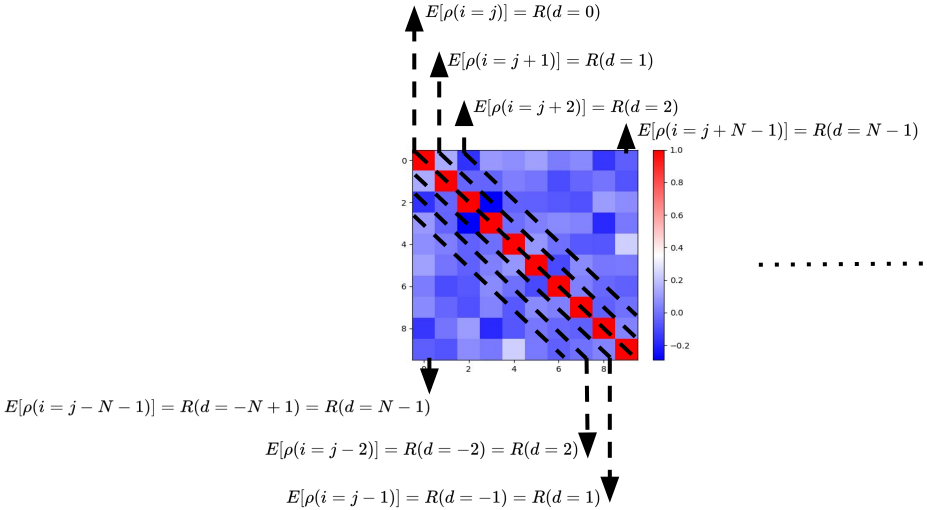
White noise?

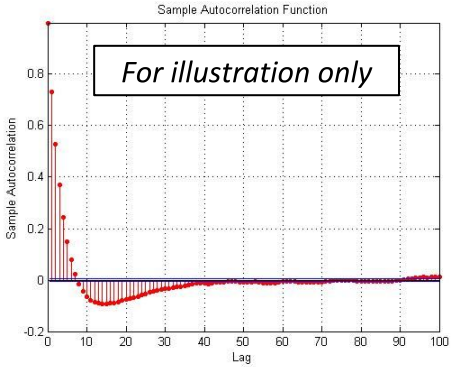
$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y}$$

The normalization avoids the interference of high std pixels in matrix calculation

Feature selection

- Transforming a correlation matrix to an **autocorrelation function** *(stationary process)*;
- Therefore, each row will generate an autocorrelation function





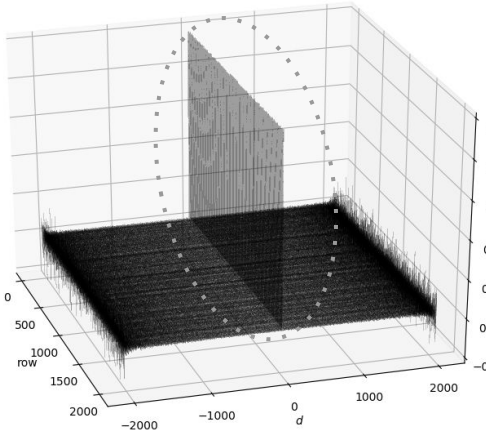
As known, if the neighbors are not correlated, an impulse is expected

Feature selection

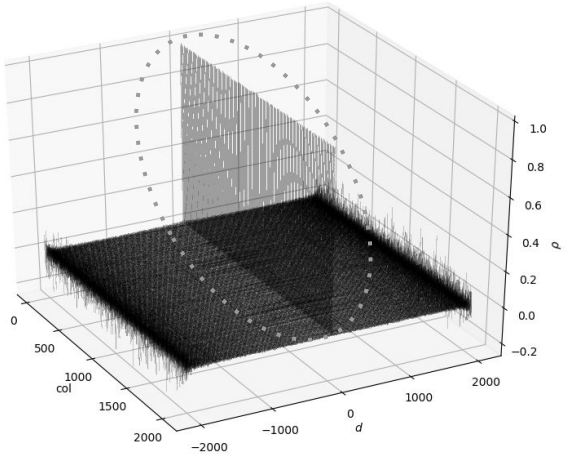
- All the 2048 rows autocorrelation functions are shown below at left
- At right same was done for the columns

Example run 818

Rows correlation



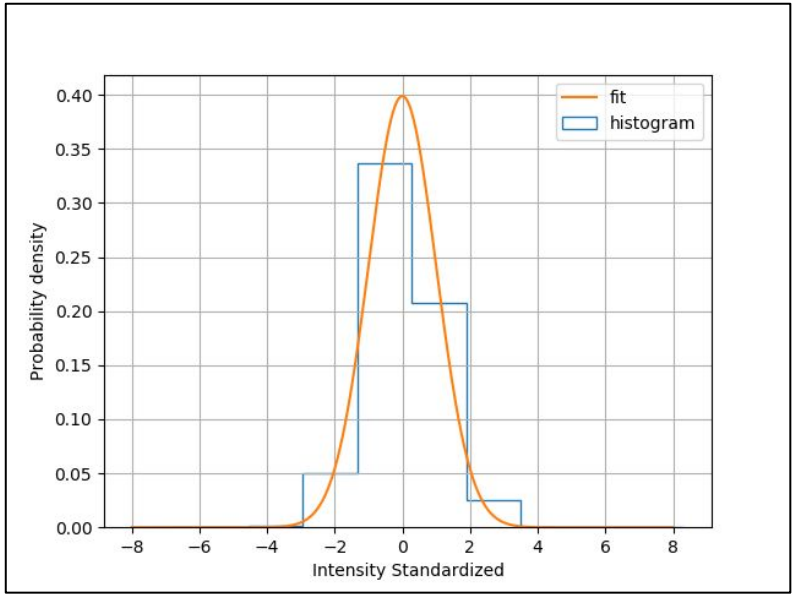
Columns correlation



- No correlation between pixels;
  - The same result was obtained by analyzing the other runs.
- Therefore, pixels can be generated independently from one another.

Pixel's pdf extraction

- Noise distribution for a single pixel
  - A Gaussian function seems appropriate
    - Only pixels with mean = 99 and STD within the boxplot limits were evaluated
    - A deeper model analysis is needed to confirm it



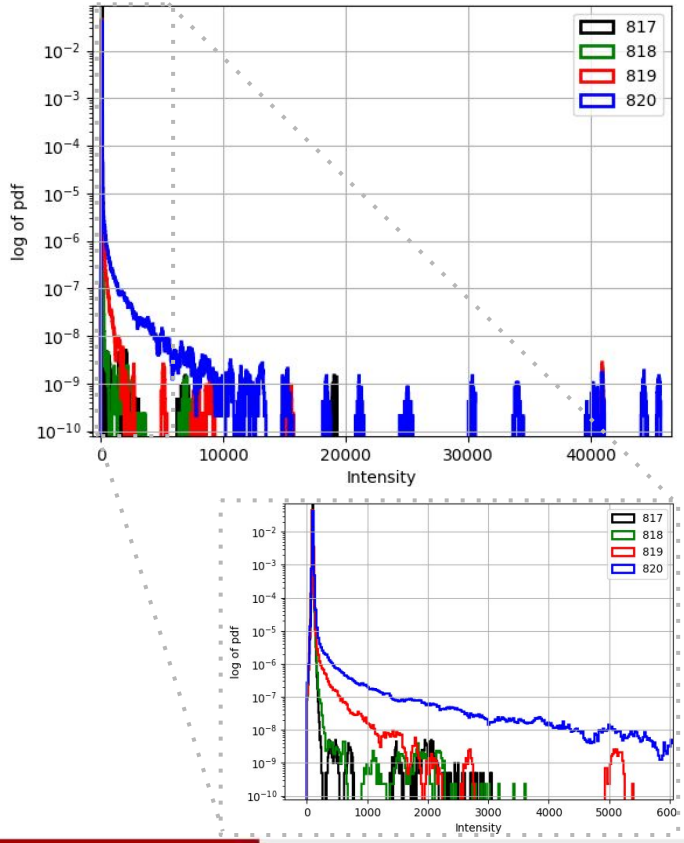
Noise Generation

Noise Generation

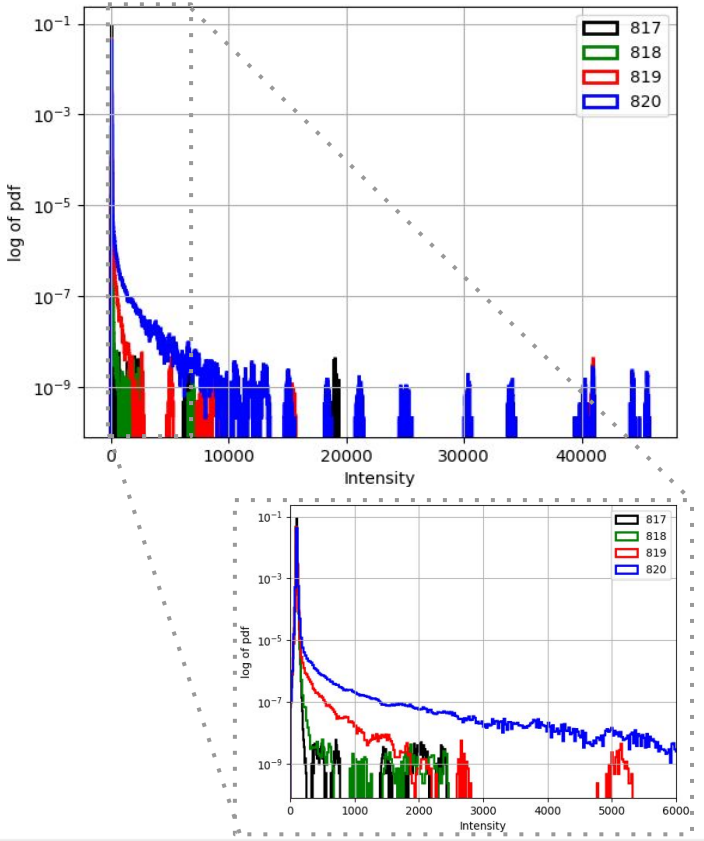
- For each pixel, a Gaussian noise distribution was considered
  - Each pixel got its own value of mean and STD, based on real Cygno images
  - Each pixel is uncorrelated of its neighbors
- A simulation dataset was generated for each run
  - 817, 818, 819 and 820
- Finally, each simulated dataset was compared with its respective real dataset (next slides)

Noise Generation **intensity distribution** - simulation vs. real

Real



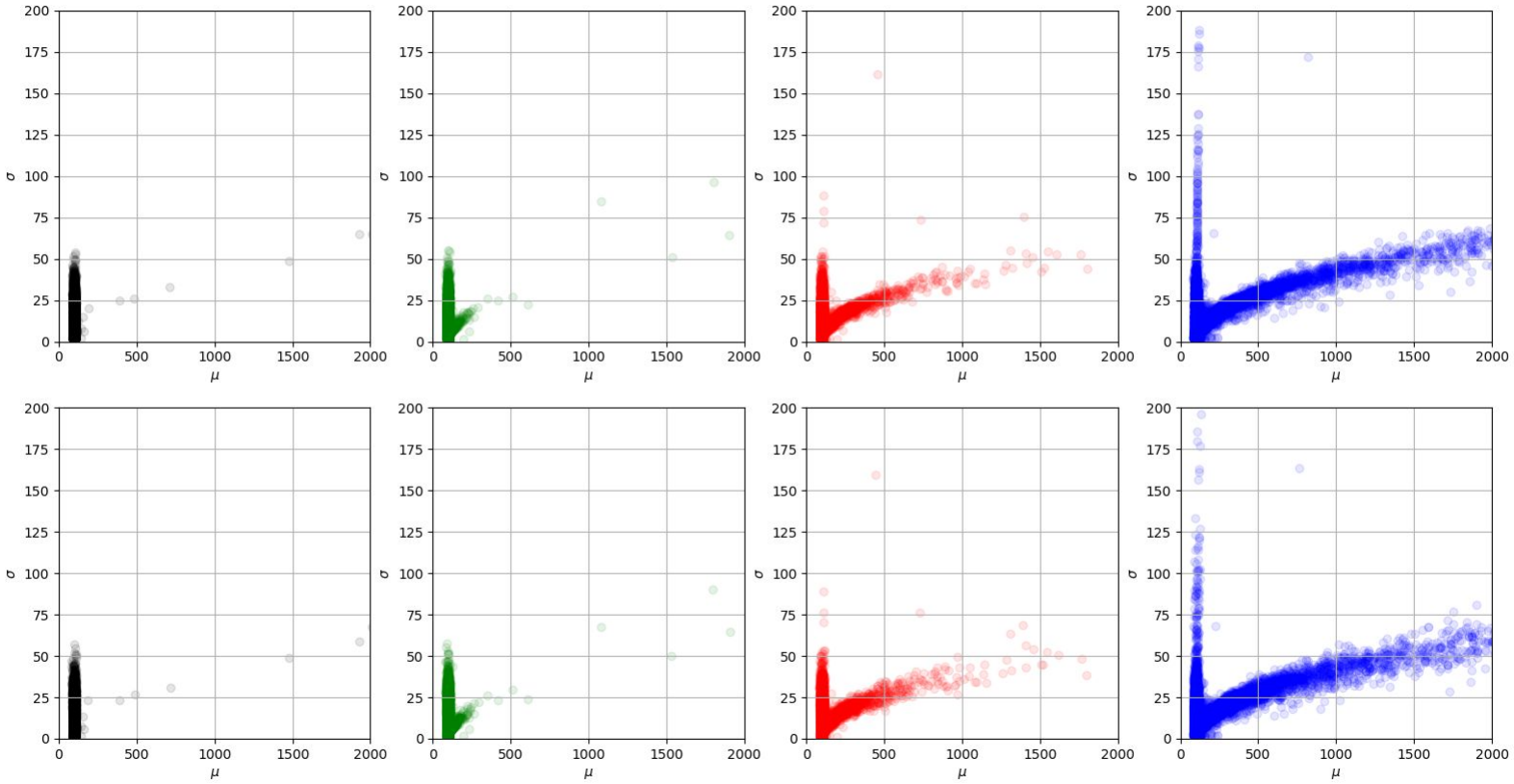
Simulated



Noise Generation **intensity distribution** - simulation vs. real

Real

Simulated



Next steps

Next steps *Considering the feedbacks given during the meeting of 06/06/2019*

- In the short term
  - Evaluate noise pixel-by-pixel and study their correlation (as suggested by Giovanni)
  - Generate noise based on the above analysis
  - Measure filters performance based on tracks energy (luminosity) estimation
  - Check other existing Cygnus simulations [SRIM and DEGRAD based] (as suggested by Elisabetta)
    - It would be important a close contact with an expert to try to understand what can be done and how

- Open issues
  - Create classes of particles to get closer to reality, to simulate for example background tracks mixed to signal tracks in the same image (...thinking how to do that)