Trigger proposal

Igor Pains
with Rafael Nóbrega

09/01/2025
Analysis & reconstruction meeting

Previous presentation

What was done

- Two algorithms were proposed:
  - **Filtering** based trigger.
  - **CNN** based trigger.

- A comparison analysis was done using them:
  - The proposed algorithms may **detect ~80%** of the **0.25 keV NR** and **ER simulated events** with a **small false alarm** ratio**.**
    - **Gaussian filter** with **10% false alarm** (**20 out of 200** pedestal images **misclassified**).
    - **CNN** with **0.5% false alarm** (**1 out of 200** pedestal images **misclassified**).
  - The proposed algorithms may **detect ~100%** of the events **above 0.5 keV.**
  - The **processing time** using **GPU** is **0.02** and **0.2 seconds per image** for the **Gaussian filter** and **CNN** respectively.

Discussion

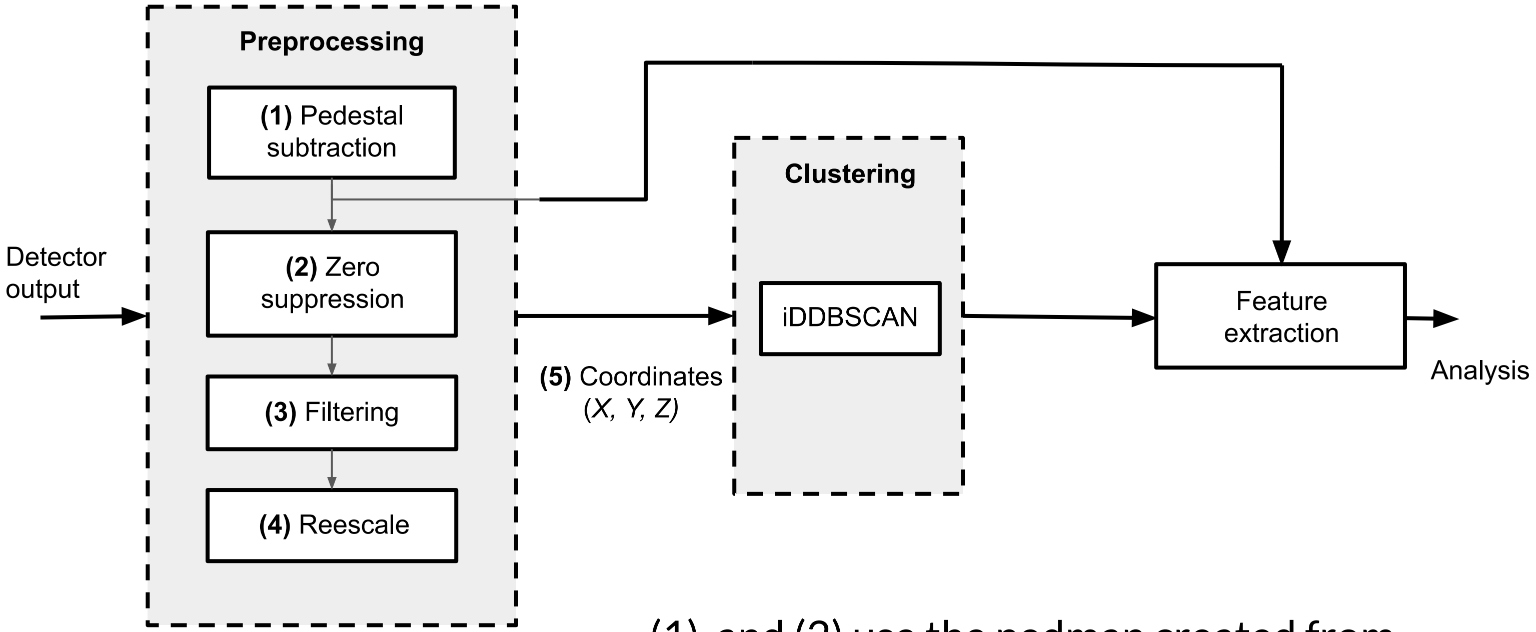
- **Why not** employ a **simpler** method, such as **the zero suppression** used in the reconstruction, for the **trigger?**[1]

- A complex method such as **CNN** or **matched filter** (based on **signal**) will give **biased results**?[2]
    - **Anomaly detection** algorithms **should be used?**

1.

Zero suppression

Reconstruction



(1) and (2) use the pedmap created from noise images.

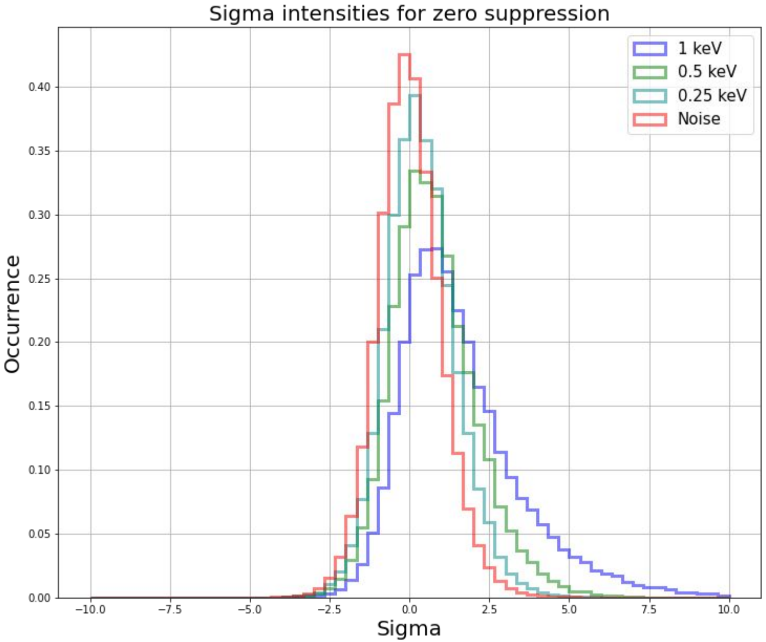
Zero suppression for trigger

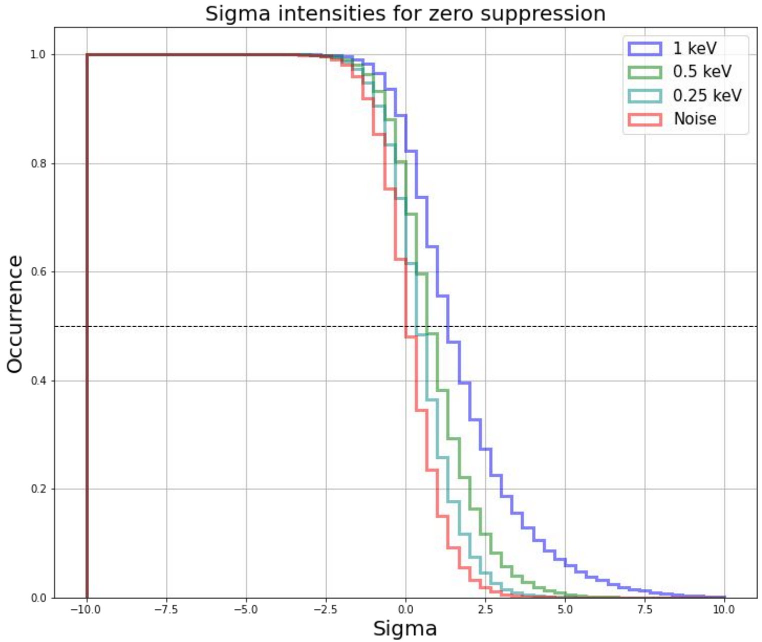
- The algorithm consists of applying **pedestal subtraction** and **zero suppression (sigma based)** on the images containing signal and only noise.
  - The training dataset of the previous methods was used.
  - Applying **pedsub** and **dividing** the **image** by the **std map** gives an output where **intensities** are equal to the **sigma** on each pixel (a **sigma threshold** equal to **2** would **maintain** the **indexes** of the **blue elements** in the **pedsub img**).

- Two approaches were considered:
  - A **pixel level** trigger.
  - An **image level** trigger.

$$\underbrace{\begin{bmatrix} 4 & 4 \\ 2 & 9 \end{bmatrix}}_{\text{Pedsub img}} \div \underbrace{\begin{bmatrix} 2 & 4 \\ 2 & 3 \end{bmatrix}}_{\text{Std map}} = \underbrace{\begin{bmatrix} \color{blue}{2} & \color{red}{1} \\ \color{red}{1} & \color{blue}{3} \end{bmatrix}}_{\text{Sigma img}}$$

Zero suppression trigger





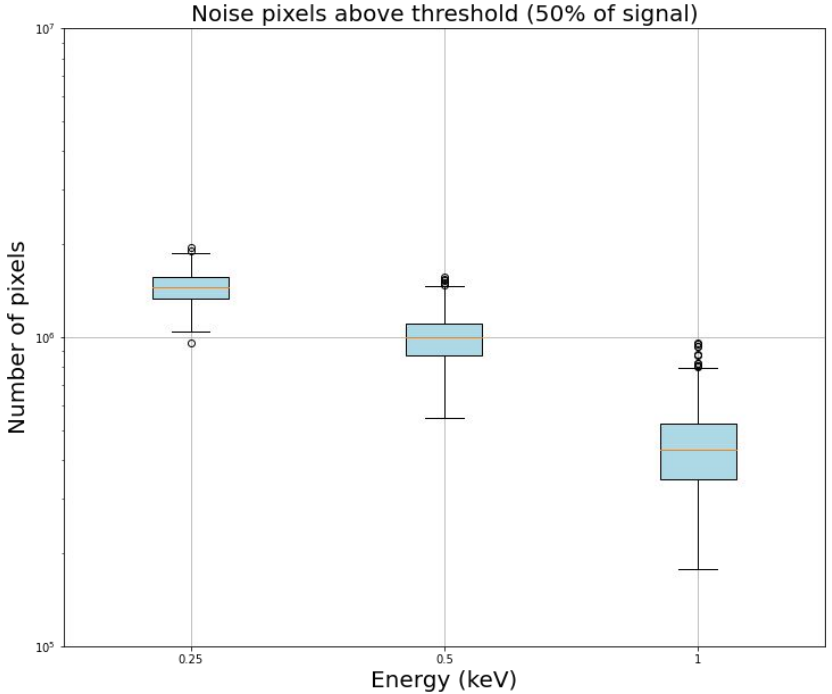
- ▷ Average histogram for signal and noise pixels on 600 images.
- ▷ Lower energy events have pixel intensities inside the noise fluctuation.
- ▷ 1 keV events show a long tail (the highest intensity pixel is generally way above the noise).

Zero suppression trigger (pixel level)

- A **threshold** capable of **detecting half** of the **signal pixels** was used to evaluate this method as a **pixel level trigger.**

- In average, the following **number of noise pixels** are **above these thresholds** for each **energy:**
  - **0.25 keV: 1452031 (reject ~73%)**
  - **0.5 keV: 994217 (reject ~81%)**
  - **1 keV: 449873 (reject ~91%)**

- A **high rejection** in terms of **percentage**, but **low** in **total number** (and would **lose half** of **signal pixels**).

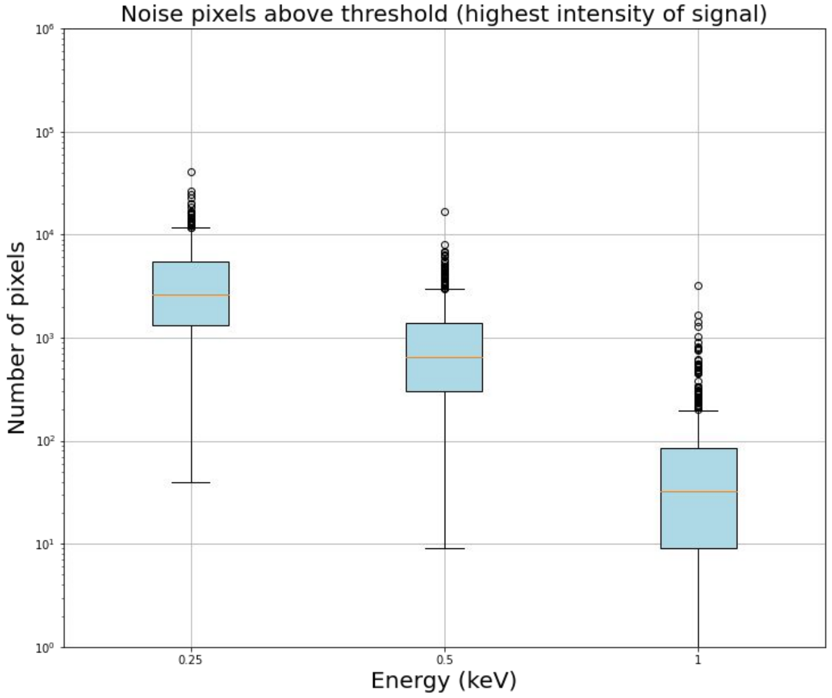


Zero suppression trigger (image level)

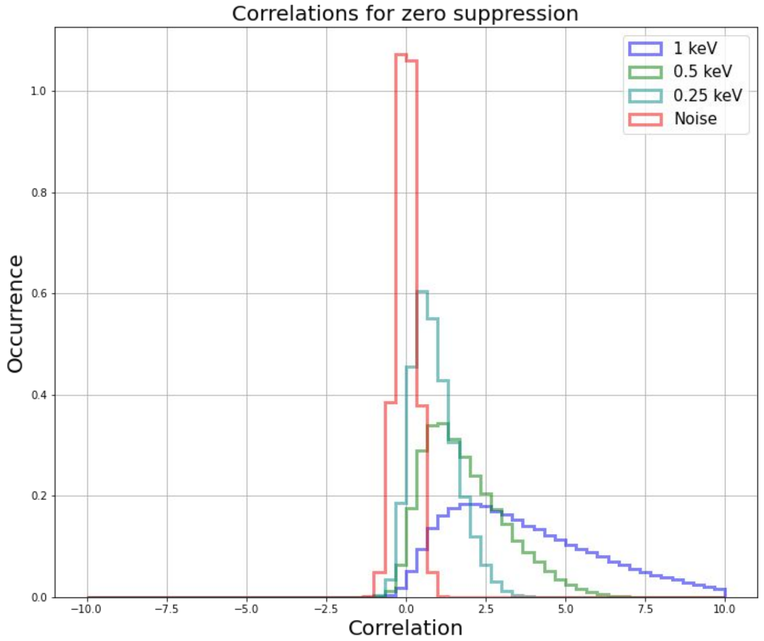
- A **threshold** capable of **detecting** the **highest intensity pixel** from a **signal** is **enough** for an **image level trigger.**

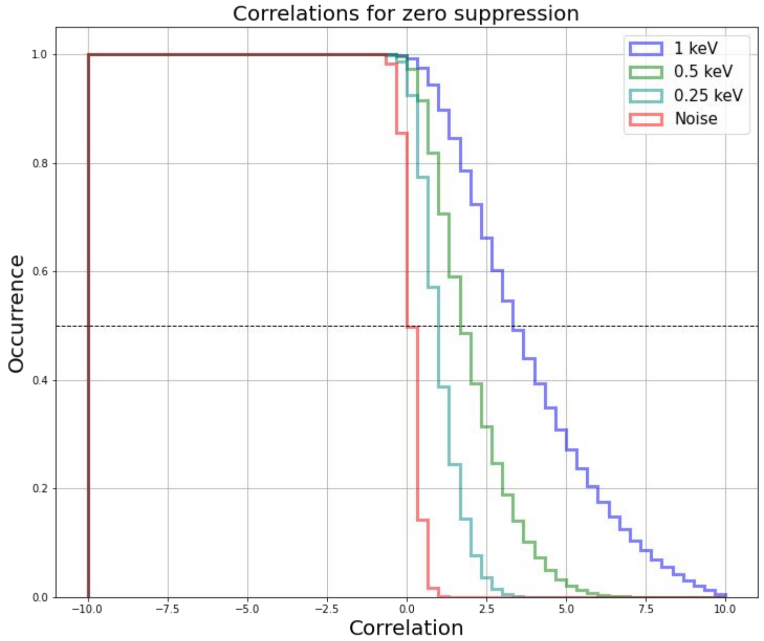
- In average, the following **number of noise pixels** are **above this threshold** for each **energy:**
  - **0.25 keV: 4017**
  - **0.5 keV: 1083**
  - **1 keV: 92**

- Every **noise** image would be **triggered.**



Zero suppression filter





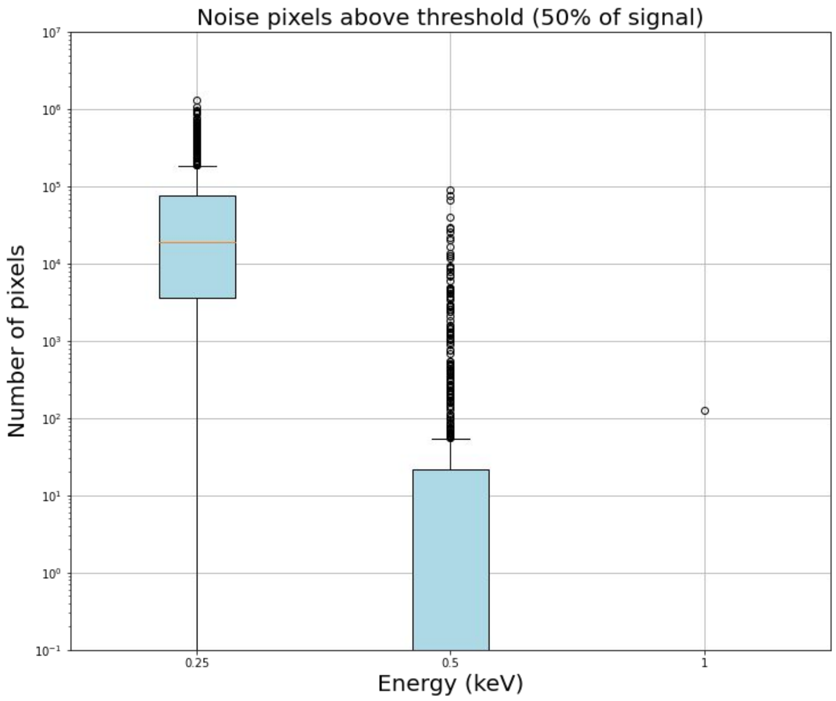
- Applying the Gaussian filter pushes the histograms away.
- The highest intensity pixel of the signals is generally above the noise.

Zero suppression filter (pixel level)

- A **threshold** capable of **detecting half** of the **signal pixels** was used to test this method as a **pixel level trigger.**

- In average, the following **number of noise pixels** are **above these thresholds** for each **energy:**
  - **0.25 keV: 86412 (reject ~98%)**
  - **0.5 keV: 1051 (reject ~99.9%)**
  - **1 keV: 0 (reject ~100%)**

- A **better rejection** compared to the **previous method**, but **not enough.**

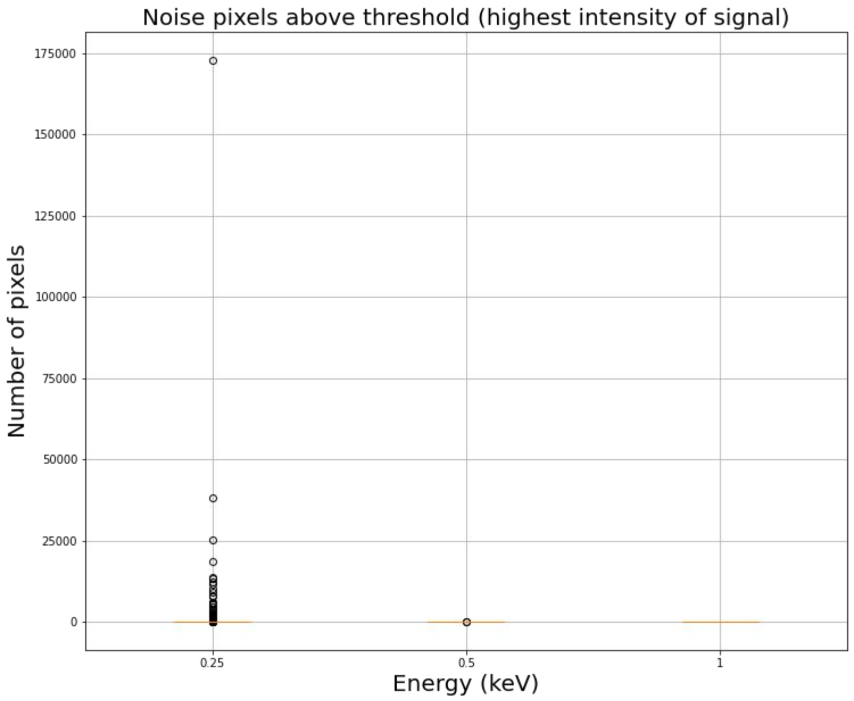


Zero suppression filter (image level)

- A **threshold** capable of **detecting** the **highest intensity pixel** from a **signal** is **enough** for an **image level trigger.**

- In average, the following **number of noise pixels** are **above this threshold** for each **energy:**
    - **0.25 keV: 663**
    - **0.5 keV: 0**
    - **1 keV: 0**

- The training dataset contains some signals with **very low ADC counts energy** compared to the others (highly affecting average of noise pixels above the threshold for 0.25 keV.)



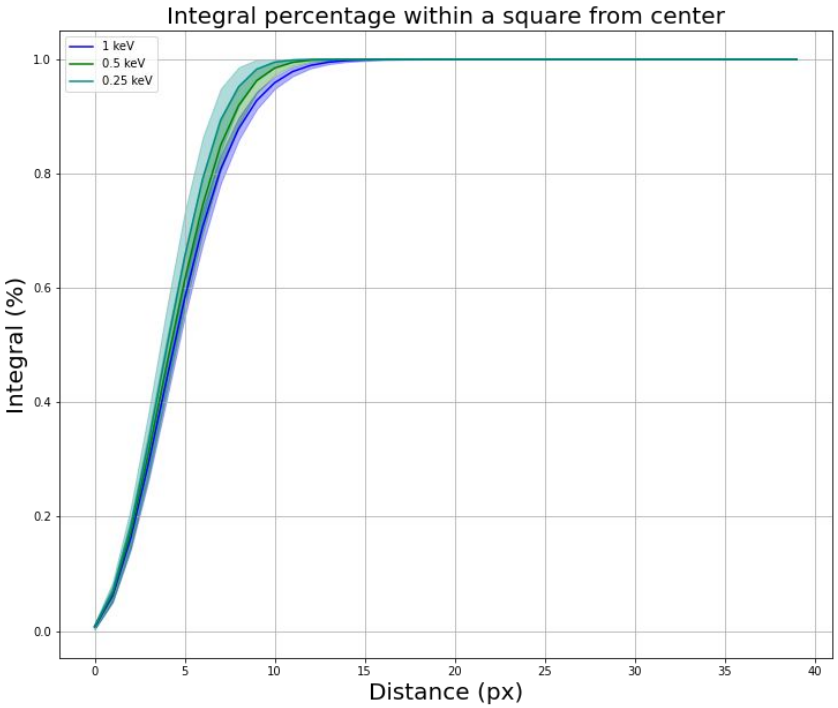
Pixel level trigger using centroids

- An alternative approach for a **pixel level trigger** involves **identifying pixels** with a **high likelihood** of belonging to a signal **(centroids)** and preserving the **surrounding region.**
  - Use a **high threshold** to **reject most part** of the **noise maintaining** at least **one pixel** of the **signal (threshold from correlation method).**
  - Study the **smallest radius** to save the **entire signal** around the **centroid (next slide).**
  - Measure overall performance (to be done).

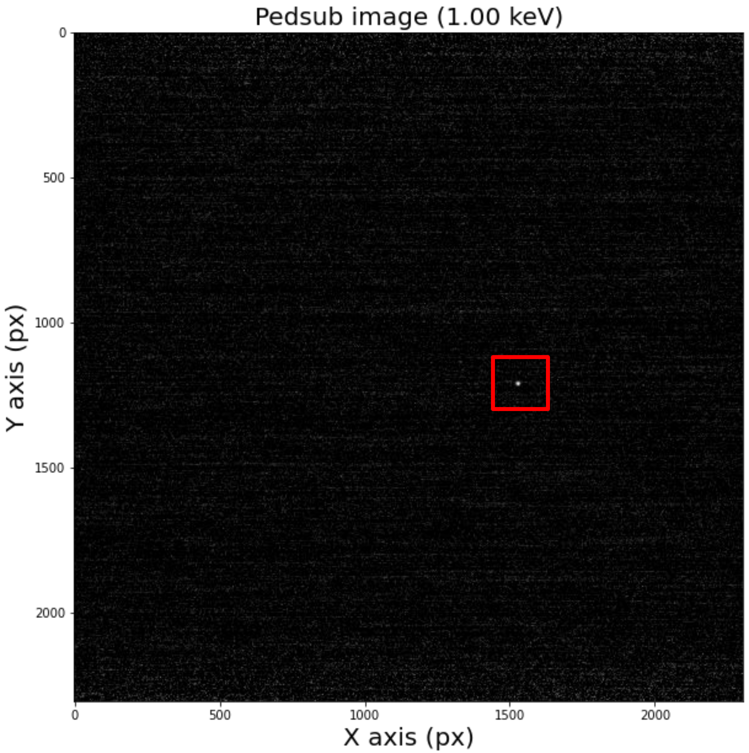
Pixel level trigger using centroids

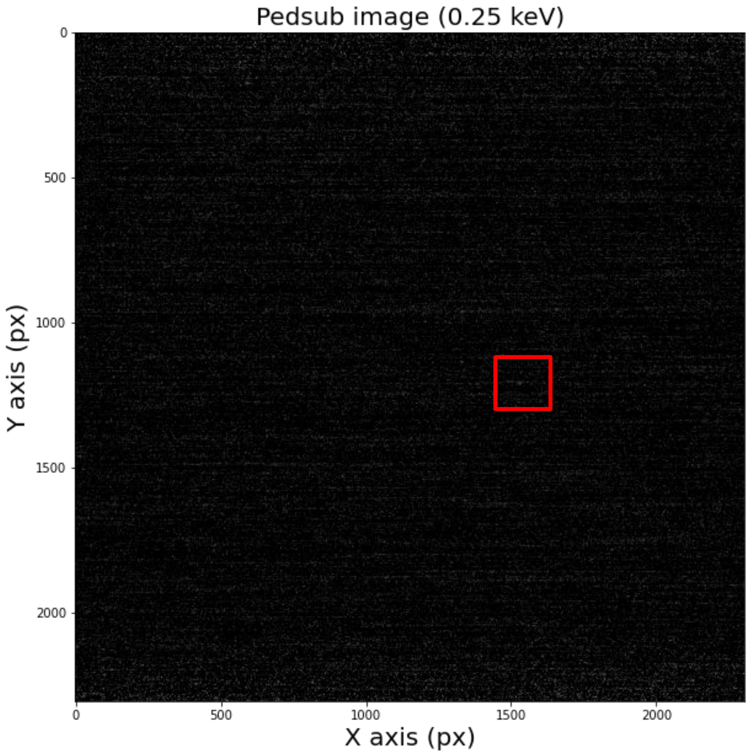
- A study of the **window necessary** to save the **signal pixels** was done using **100 simulated NR** events.

- A **radius** of **10** px from the **center** of the **signal** is **enough** to **detect** almost all of the 0.25 keV signal **energy** (**21x21 square**).
  - This radius is increased to ~15 for 1 keV.
  - **More than one pixel** from **1 keV events** are expected to be **above the threshold.**

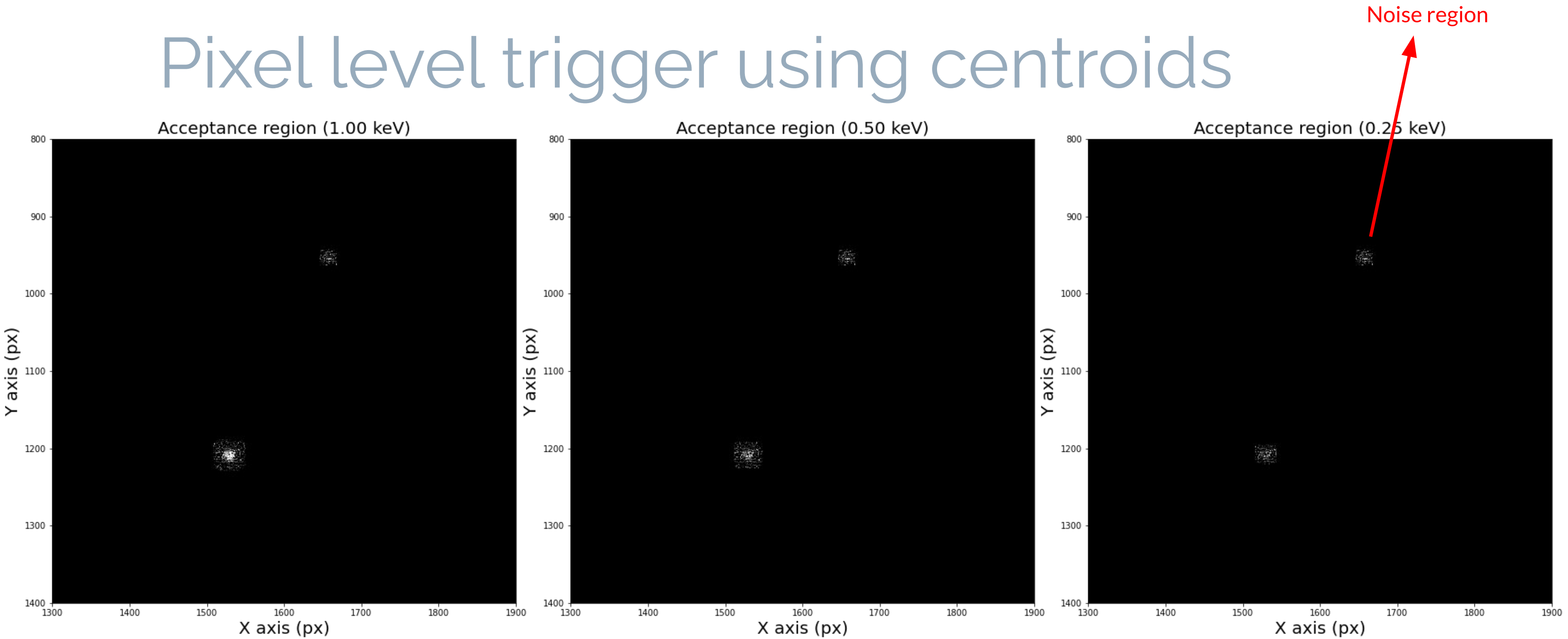


Pixel level trigger using centroids



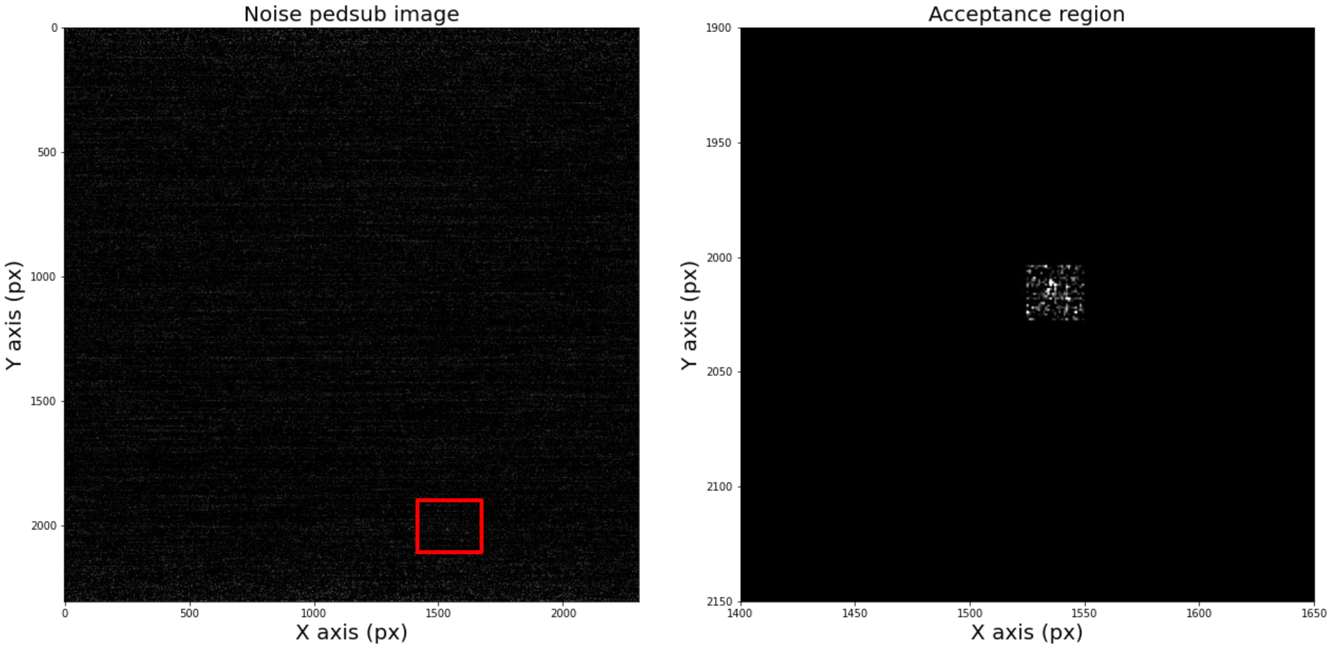


Pixel level trigger using centroids



- 1 keV: : **360** centroids -> **2060** pixels on the acceptance region **(100% of the signal energy)**. (0.0085 seconds)
- 0.5 keV: **195** centroids -> **1675** pixels on the acceptance region **(99.93% of the signal energy)**. (0.0074 seconds)
- 0.25 keV: **21** centroids -> **1121** pixels on the acceptance region **(99.15% of the signal energy)**. (0.0064 seconds)

Pixel level trigger using centroids



- Noise: 18 centroids -> 599 pixels on the acceptance region. (0.0176 seconds)

Conclusions and next steps

- The **zero suppression** alone is **not enough** to work neither as **image** nor **pixel level trigger.**

- A **pixel level trigger** based on **centroid detection** through **filtering (high correlation points)** seems **promising.**
  - Choose **hyperparameter** (**threshold** for **centroids** and **radius**).
  - **Efficiency** measures using the **datasets.**
  - **Processing time** measure.

2.
Anomaly detection

Anomaly detection

- **Anomaly detection** algorithms are specialized in **detecting outliers** on data.
  - Technically, **both trigger methods** proposed may be **considered** as **anomaly detection** (**supervised**).
  - Training a model with **labeled data** will **limit** the **detection of anomalous events** in CYGNO's case?
  - Can we consider a **track** as a **sum of various smaller tracks? (tested but not happen)**

- A possible solution would be to train a model using **only noise images**.
  - A promising approach involves using **autoencoders.**

Thanks!