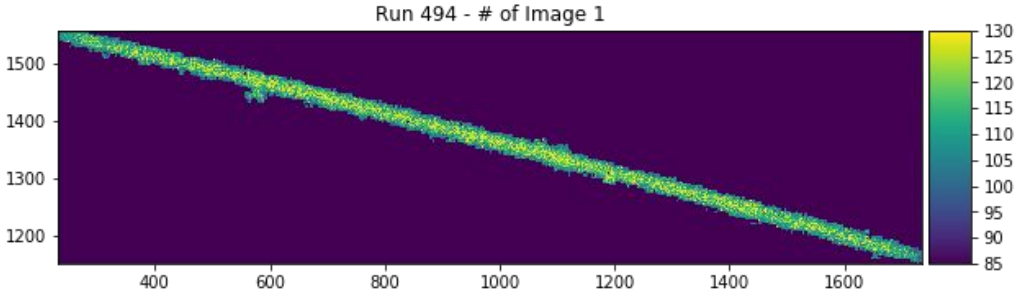
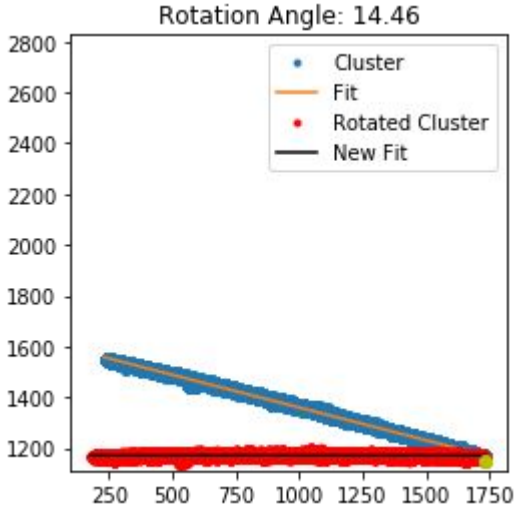
Analysis Long Tracks

02-21-2019

Characterize the ‘long’ tracks

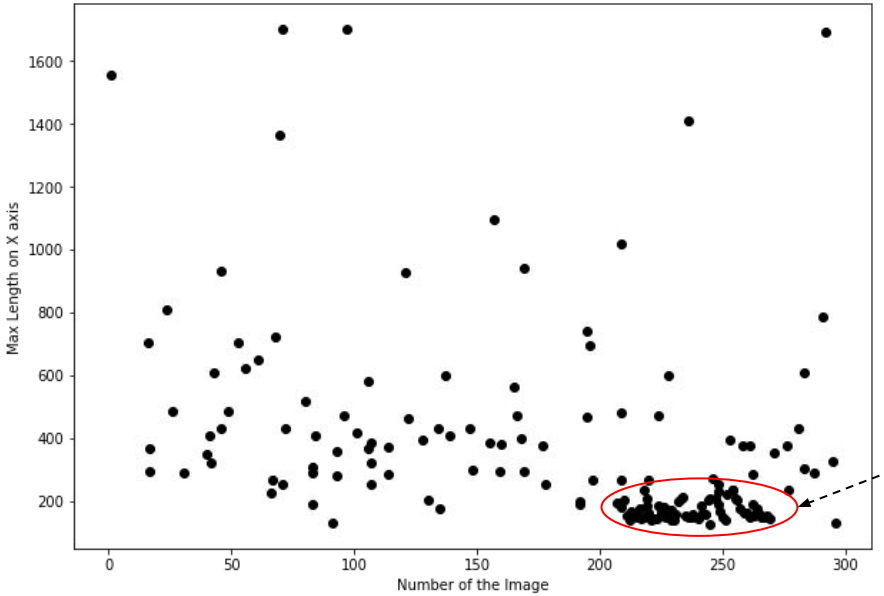
- The idea is:
  - Do a Fit on the the coordinates;
  - Use this Fit to get the angle;
  - Use the angle to calculate the Transformation matrix;
  - And then slice the cluster and calculate the intensity of light of each slice.

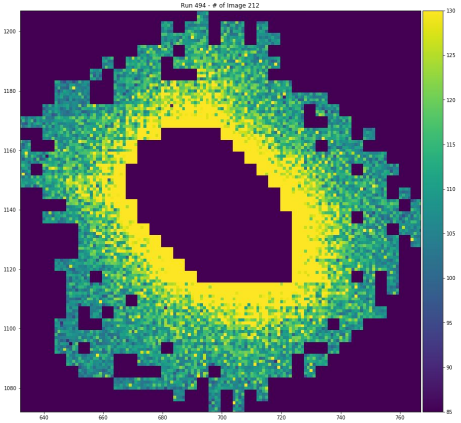


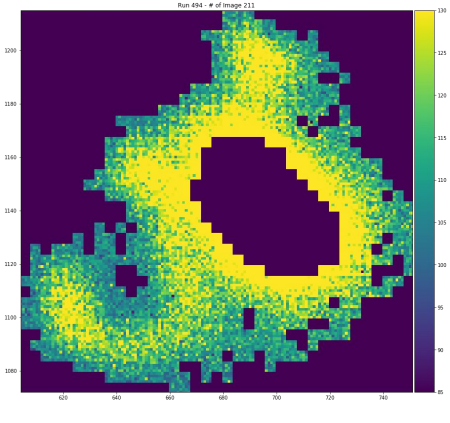


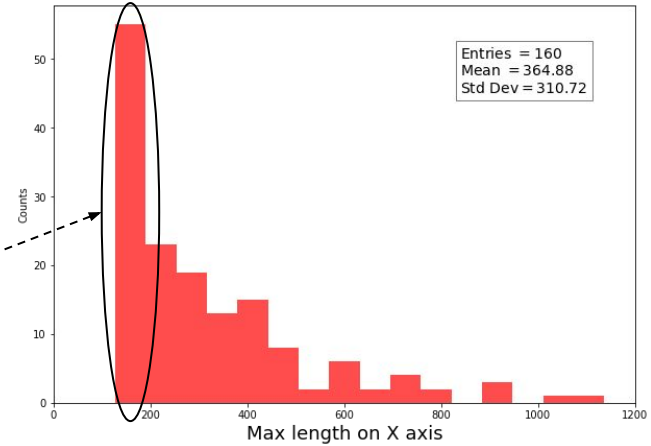
Sparks founded

Look at the Max Length on X profile we saw a ‘problem’ when it is justified by ~50 consecutively images with a spark.



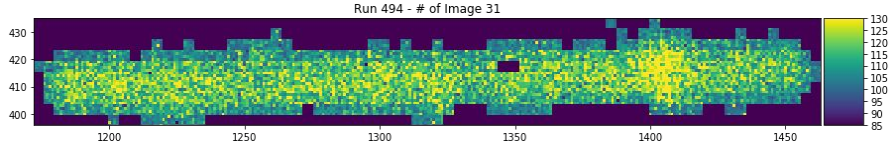




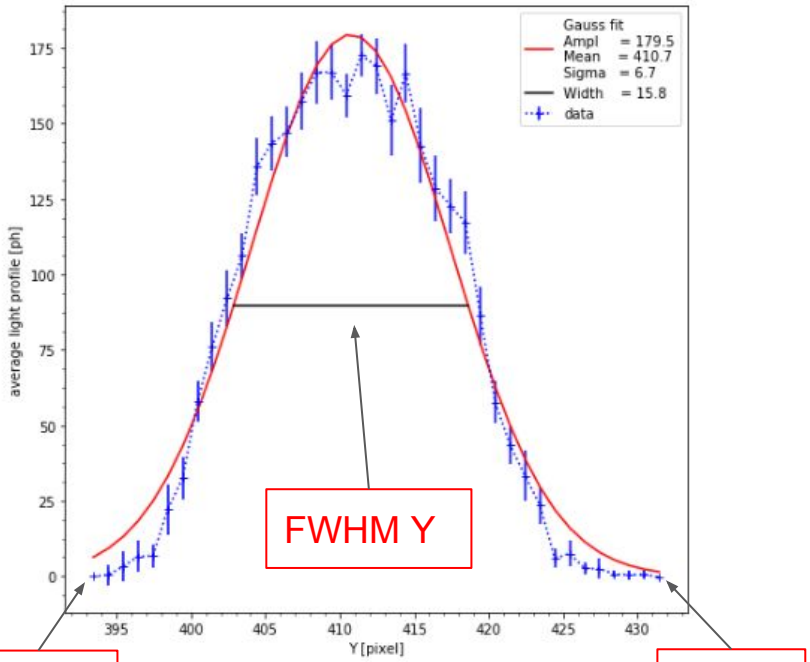


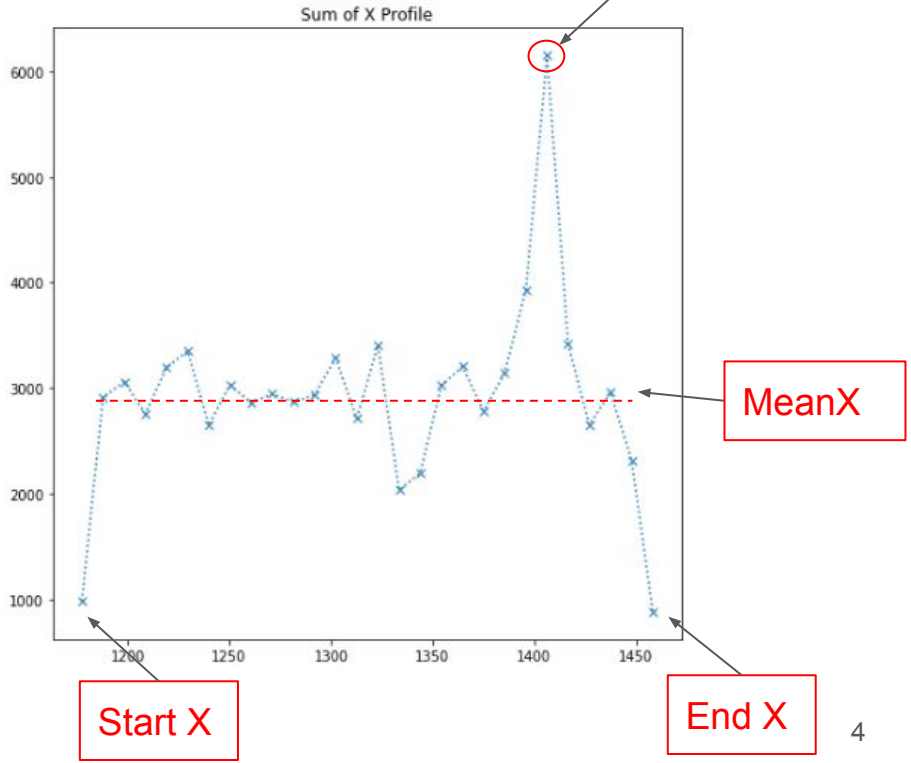
Question: What we do with a image that have spark?
Discard the entire image? Or ignore only the spark?

Example of cluster:



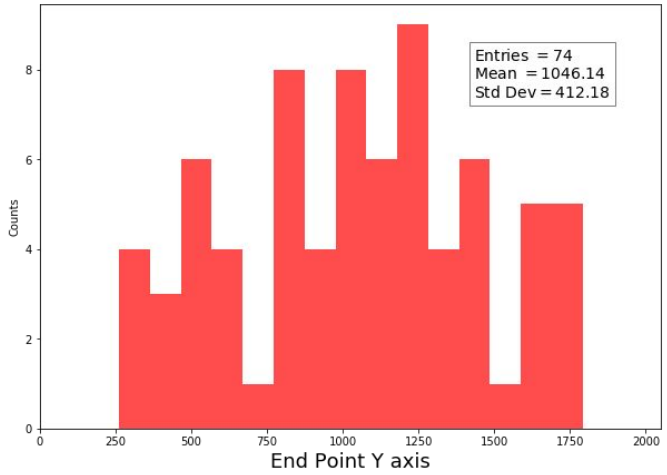
- Features that we are extracting from the cluster:
  - The slices has **10px** and **1px** of length on **X** and **Y profile**, respectively.

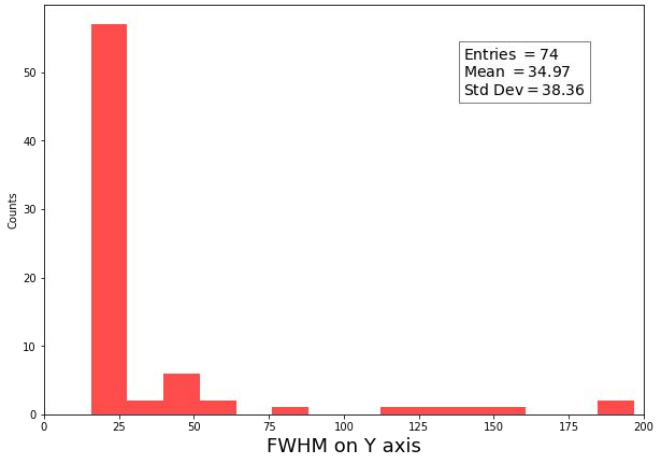


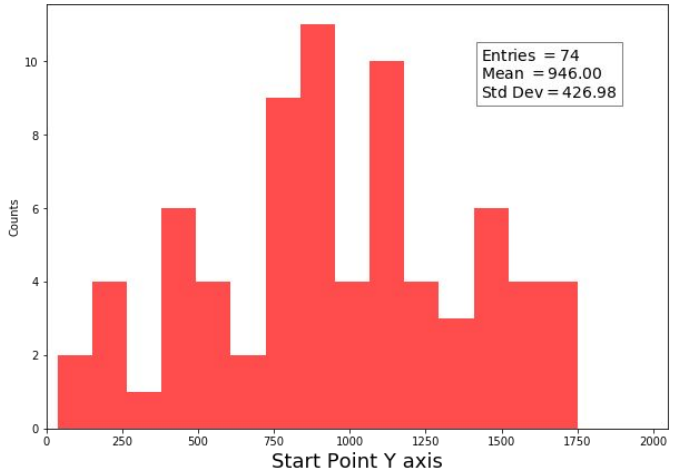


Characterize the ‘long’ tracks - Y Profile

- Looking at the histogramas we can see that the most probably value of FWHM on Y profile is around 25px (1px - 55µm).
- And we the start or end point is quite uniform.

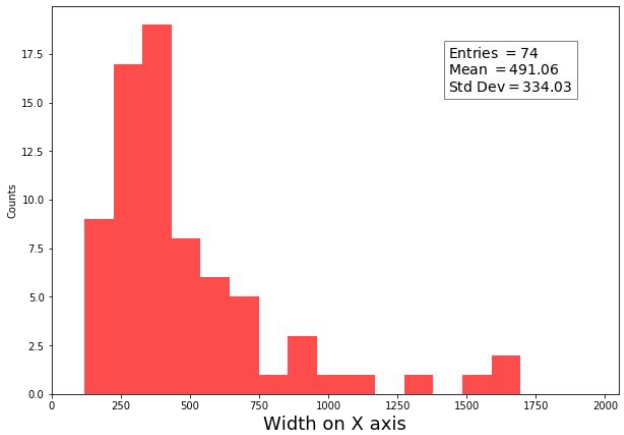


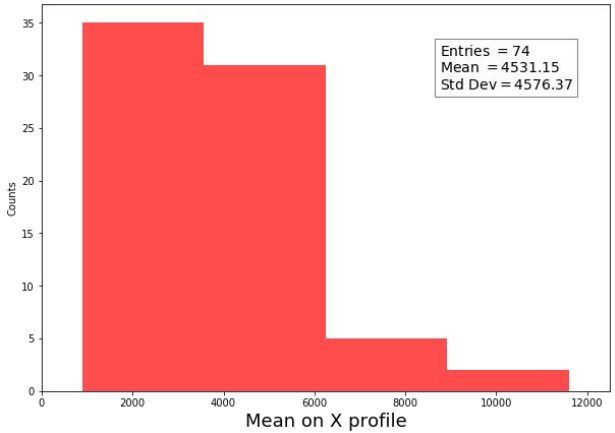


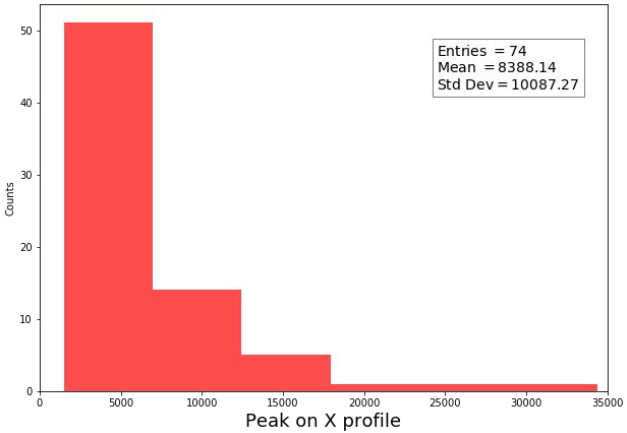


Characterize the ‘long’ tracks - X Profile

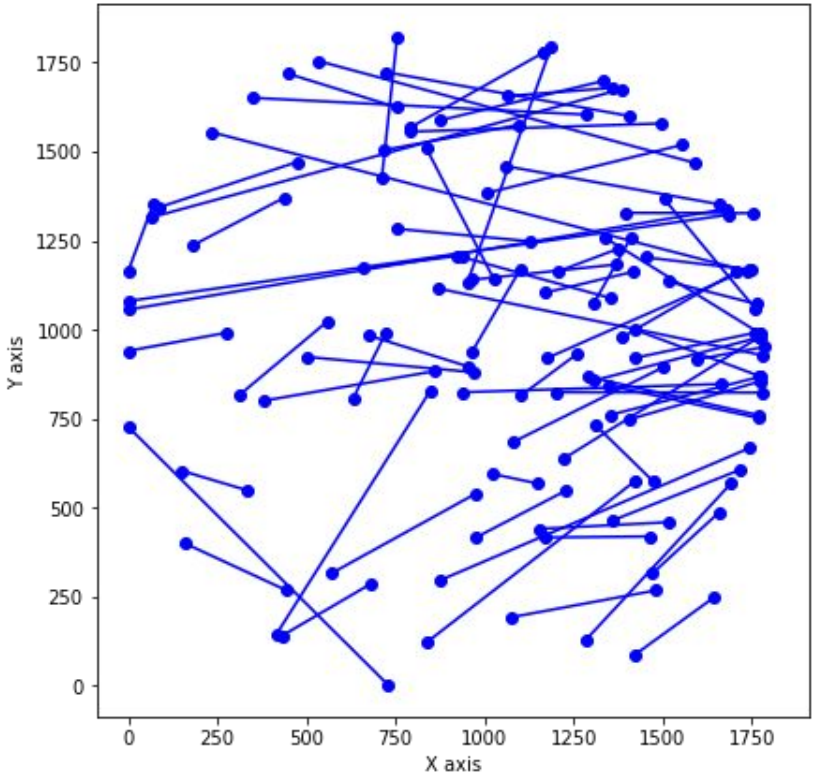
- Looking at the histogramas we can see that the most probably value of width on X profile is around 400px (1px - 55µm).

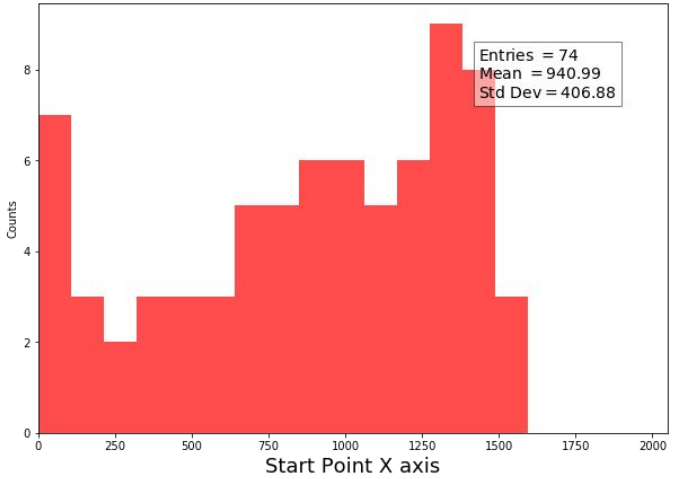


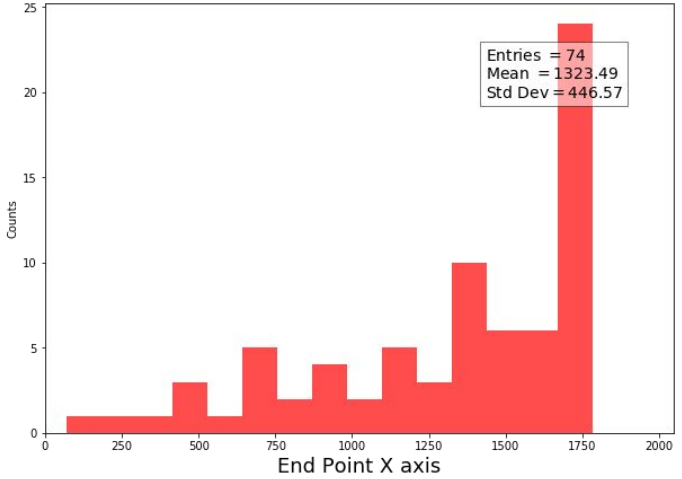




Start and End points on X profile







Variables for the ‘long’ tracks

- ❏ Sum Light
- ❏ Sum Pedestal
- ❏ SumPixels
- ❏ FWHM Y
- ❏ Start point Y
- ❏ End point Y
- ❏ Width X
- ❏ Start point X
- ❏ End point X
- ❏ Mean Y
- ❏ Peak X

Backup

Status Report

- Developed the algorithm to:
  - Run *i2DBSCAN* over the images (one or more runs);
  - Create a table with the necessary information;
  - The algorithm is taking 8-10 minutes to run over 300 images and save the **table**.
- We chose to not generate all the variables in the same time that we are doing the clusterization.
- But the ‘table’ has all the information needed to create the variables.
  - For now, it is better to have ‘checkpoints’ in this stage of developing the framework;

File name	Tens GEM (V)	He:CF4	T2 (kV/cm)	Drift field	esposizione (s)	Commenti
run494	440	60/40	2	1	2	neutroni from AmBe + CMOS + 1 panetto di Pb+piu' vicini (300 eventi)

- **On Going:**
  - We are using ‘Pandas’ to be able to analyze the clusterization output;
  - In this way, we are capable of create any variable easily;
  - And our focus is characterize the ‘long’ tracks.

Output table of iDBSCAN

Tag of each cluster
(Long, Medium or Small)

Coordinates of each pixel

Light of each pixel

Pedestal of each pixel

Index of each cluster

	Run	Image	Tag	X	Y	Light	Pedestal
0	494	0	m	[808, 808, 808, 808, 809, 809, 809, 809, 810, ...	[40, 41, 42, 43, 40, 41, 42, 43, 40, 41, 42, 4...	[103, 96, 98, 103, 97, 129, 99, 120, 100, 98, ...	[103, 102, 101, 104, 100, 115, 102, 104, 104, ...
1	494	0	m	[652, 652, 652, 652, 653, 653, 653, 653, 654, ...	[112, 113, 114, 115, 112, 113, 114, 115, 112, ...	[98, 102, 99, 104, 103, 102, 101, 98, 103, 99,...	[99, 102, 103, 102, 102, 102, 100, 102, 102, 1...
2	494	0	m	[404, 404, 404, 404, 405, 405, 405, 405, 406, ...	[160, 161, 162, 163, 160, 161, 162, 163, 160, ...	[96, 101, 102, 103, 100, 100, 100, 98, 104, 10...	[96, 102, 102, 102, 99, 103, 102, 99, 102, 103...
3	494	0	m	[1272, 1272, 1272, 1272, 1273, 1273, 1273, 127...	[488, 489, 490, 491, 488, 489, 490, 491, 488, ...	[103, 96, 100, 103, 100, 105, 100, 100, 103, 1...	[102, 103, 103, 103, 103, 103, 106, 102, 102, ...
4	494	0	m	[1208, 1208, 1208, 1208, 1209, 1209, 1209, 120...	[724, 725, 726, 727, 724, 725, 726, 727, 724, ...	[99, 108, 106, 102, 102, 106, 111, 100, 101, 1...	[102, 103, 102, 102, 104, 102, 110, 102, 102, ...
5	494	0	m	[1516, 1516, 1516, 1516, 1517, 1517, 1517, 151...	[780, 781, 782, 783, 780, 781, 782, 783, 780, ...	[103, 108, 116, 100, 113, 111, 111, 110, 109, ...	[102, 102, 105, 102, 109, 103, 102, 103, 104, ...
6	494	0	m	[1484, 1484, 1484, 1484, 1485, 1485, 1485, 148...	[856, 857, 858, 859, 856, 857, 858, 859, 856, ...	[109, 100, 103, 98, 104, 101, 101, 110, 114, 1...	[103, 104, 104, 107, 103, 103, 106, 104, 104, ...
7	494	0	m	[700, 700, 700, 700, 701, 701, 701, 701, 702, ...	[996, 997, 998, 999, 996, 997, 998, 999, 996, ...	[133, 106, 106, 101, 100, 120, 113, 103, 103, ...	[135, 103, 102, 102, 104, 110, 105, 102, 103, ...
8	494	0	m	[516, 516, 516, 516, 517, 517, 517, 517, 518, ...	[1060, 1061, 1062, 1063, 1060, 1061, 1062, 106...	[100, 103, 102, 111, 103, 98, 99, 101, 100, 11...	[105, 102, 104, 107, 103, 103, 99, 103, 102, 1...
9	494	0	m	[804, 804, 804, 804, 805, 805, 805, 805, 806, ...	[1240, 1241, 1242, 1243, 1240, 1241, 1242, 124...	[107, 102, 102, 104, 109, 109, 114, 107, 111, ...	[100, 105, 104, 103, 105, 107, 109, 103, 103, ...

New table with a few variables

Using this ‘Pandas’ tools is very simple to managed the information:

- ❑ Insert or remove variables;
- ❑ Filter by any feature.

	Run	Image	Tag	SumLight	SumPedestal	SumPixels	PhotonPPixels	XYCorrelation	LightPPixel
0	494	0	m	639904.0	605129.0	5888.0	5.906080	0.680871	108.679348
1	494	0	m	276976.0	261851.0	2544.0	5.945362	0.693469	108.874214
2	494	0	m	283543.0	271467.0	2640.0	4.574242	0.716070	107.402652
3	494	0	m	223004.0	208398.0	2016.0	7.245040	0.234880	110.617063
4	494	0	m	933856.0	886972.0	8544.0	5.487360	0.876593	109.299625
5	494	0	m	152000.0	143567.0	1376.0	6.128634	0.826296	110.465116
6	494	0	m	441261.0	413057.0	3968.0	7.107863	0.452934	111.204889
7	494	0	m	180194.0	167429.0	1616.0	7.899134	0.665383	111.506188
8	494	0	m	241204.0	228315.0	2208.0	5.837409	0.115408	109.240942
9	494	0	m	153964.0	150582.0	1456.0	2.322802	0.280943	105.744505

Simple Example

If we want to check the clusters that have:

- ❏ Tag = ‘L’
- ❏ Photon/Pixel < 10

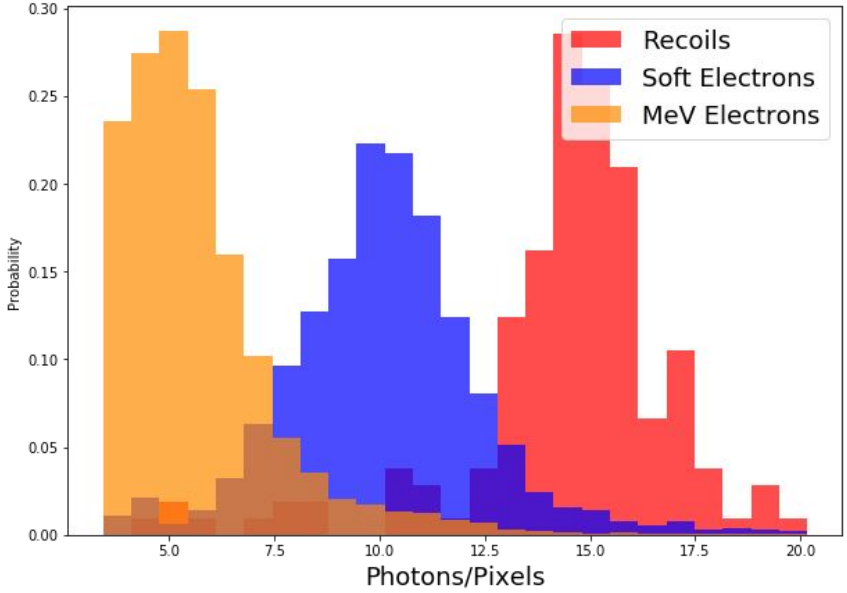
So, will be easy to plot the cluster and debug if it is necessary

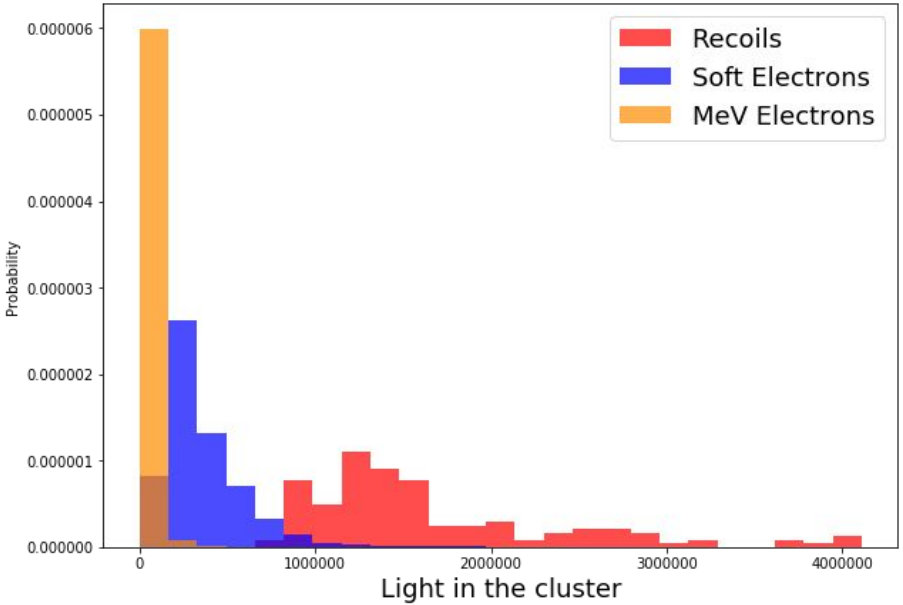
```
In [134]: variables[(variables.Tag == 'l') & (variables.PhotonPPixels < 5)]
```

Out[134]:

	Run	Image	Tag	SumLight	SumPedestal	SumPixels	PhotonPPixels	XYCorrelation	LightPPixel
6930	494	83	l	4066372.0	3992954.0	38944.0	1.885220	0.336163	104.415879
6931	494	83	l	1239130.0	1198118.0	11680.0	3.511301	0.242340	106.089897
7933	494	93	l	5170105.0	4956233.0	47984.0	4.457152	0.097981	107.746436
18080	494	209	l	13862470.0	13572413.0	132880.0	2.182849	0.798478	104.323224
18081	494	209	l	1765949.0	1742871.0	17104.0	1.349275	0.573098	103.247720
18082	494	209	l	8106981.0	7801219.0	75600.0	4.044471	0.049161	107.235198
18083	494	209	l	965660.0	948860.0	9280.0	1.810345	0.267739	104.058190
25661	494	271	l	7176089.0	6885724.0	66688.0	4.354082	0.233786	107.606901

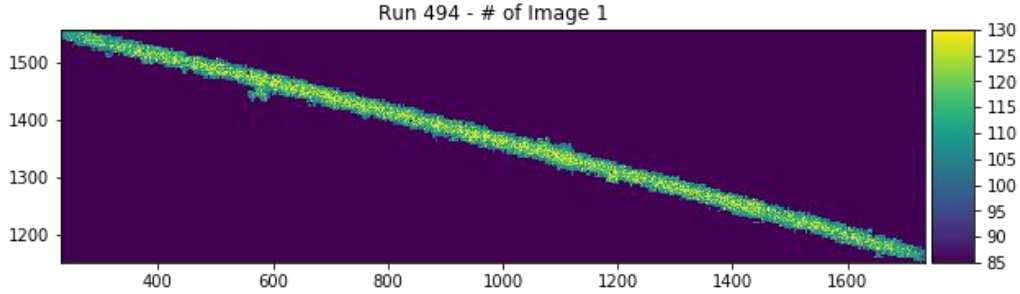
Preliminary Results - Variables

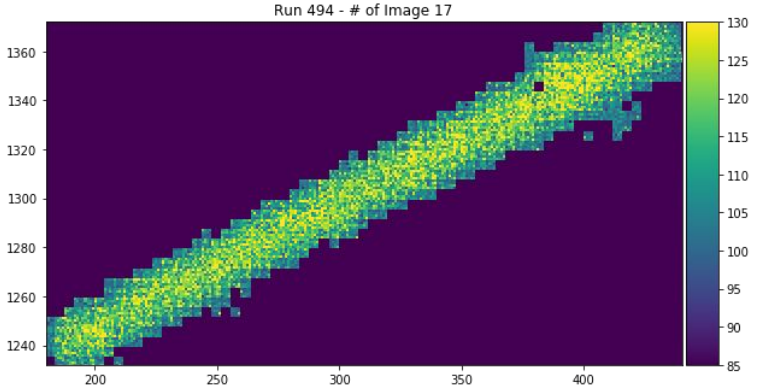


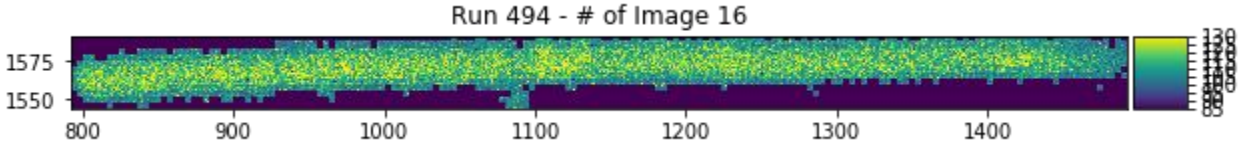


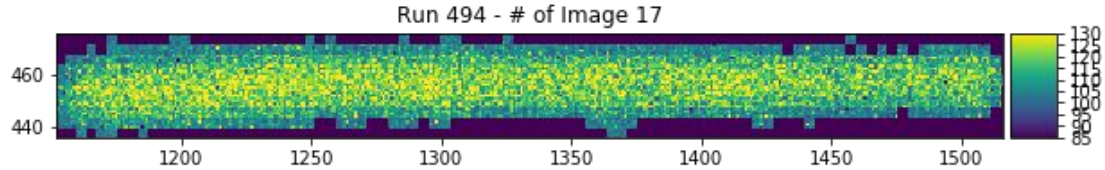
As we are creating the framework, we have not yet stopped to deeply analyze the results.

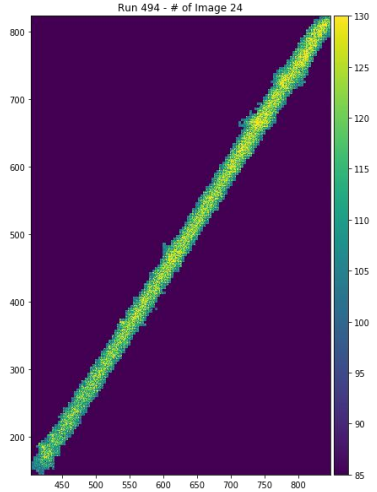
Examples of ‘L’ Clusters - 160 found on Run 494



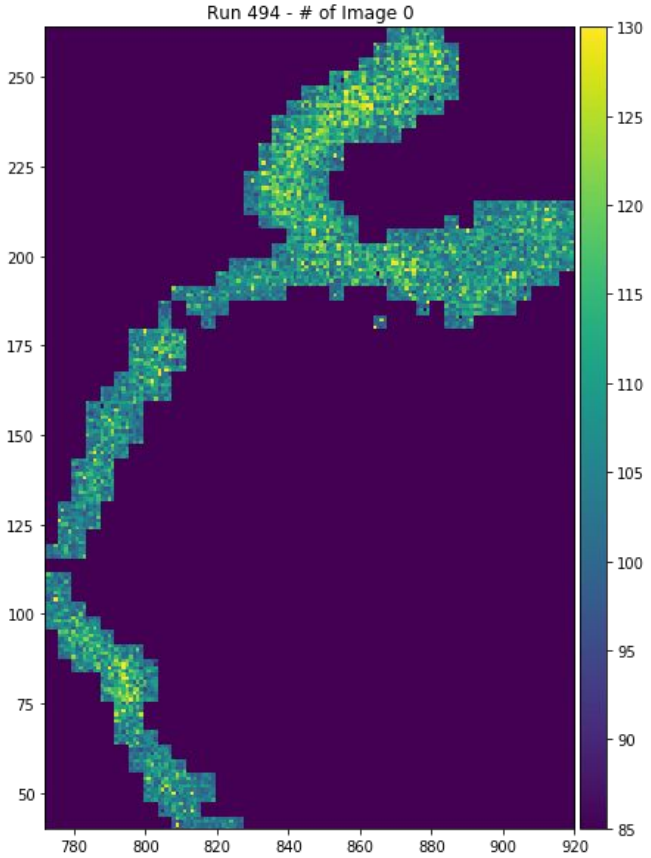


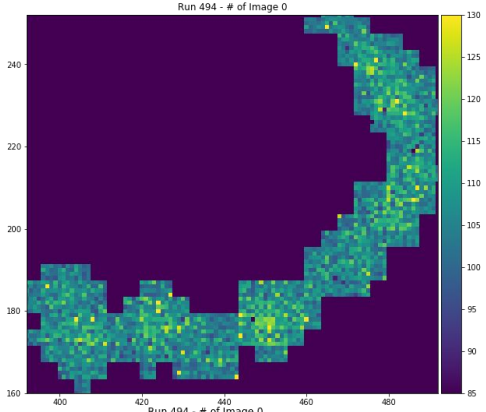


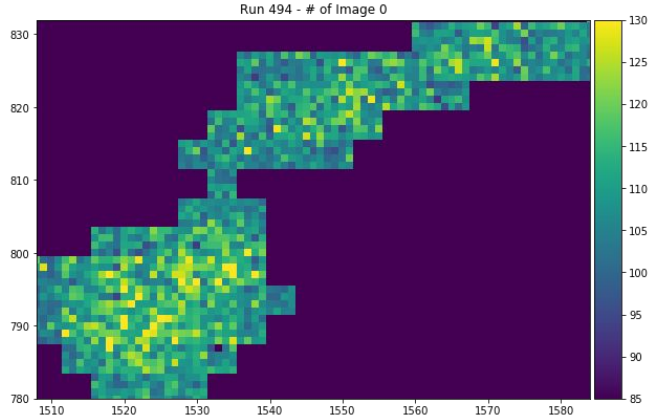


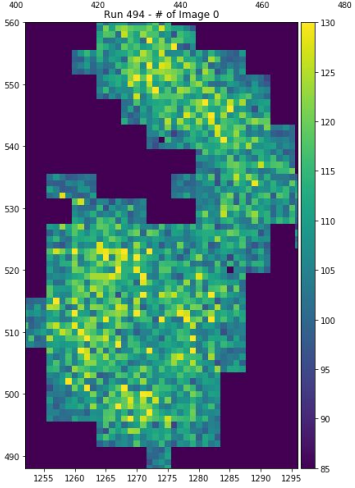


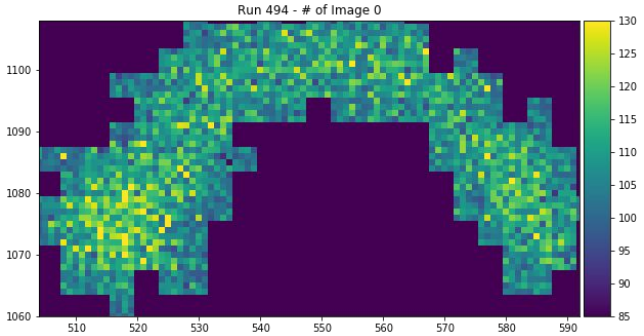
Examples of ‘M’ Clusters - 3328 found on Run 494











Examples of ‘S’ Clusters - 24964 found on Run 494