Study of the Impact of Pre-processing Applied to Images Acquired by the Cygno Experiment

G. S. P. Lopes[6]($\boxtimes$), E. Baracchini[2], F. Bellini[1,3], L. Benussi[4], S. Bianco[4], G. Cavoto[1,3], I. A. Costa[1,6], E. Di Marco[1], G. Maccarrone[4], M. Marafini[5], G. Mazzitelli[4], A. Messina[1,3], R. A. Nobrega[6], D. Piccolo[4], D. Pinci[1], F. Renga[1], F. Rosatelli[4], D. M. Souza[6], and S. Tomassini[4]

[1] Istituto Nazionale di Fisica Nucleare Sezione di Roma, 00185 Rome, Italy
davide.pinci@roma1.infn.it
[2] Gran Sasso Science Institute L'Aquila, 67100 L'Aquila, Italy
[3] Dipartimento di Fisica, Sapienza Università di Roma, 00185 Rome, Italy
[4] Laboratori Nazionali di Frascati,
Istituto Nazionale di Fisica Nucleare, 00040 Rome, Italy
[5] Museo Storico della Fisica e Centro Studi e Ricerche "Enrico Fermi",
Piazza del Viminale 1, 00184 Rome, Italy
[6] Electrical Engineering Department, Federal University of Juiz de Fora,
Juiz de Fora, Brazil
guilherme.lopes@engenharia.ufjf.br

Abstract. This work proposes to evaluate the effect of digital filters when applied to images acquired by the ORANGE prototype of the Cygno experiment. A preliminary analysis is presented in order to understand if filtering techniques can produce results that justify investing efforts in the pre-processing stage of those images. Such images come from a camera sensor based on CMOS technology installed in an appropriate gas detector. To perform the proposed work, a simulation environment was created and used to evaluate some of the classical filtering techniques known in the literature. The results showed that the signal-to-noise ratio of the images can be considerably improved, which may help in subsequent processing steps such as clustering and particles identification.

Keywords: Image processing · Digital filters · Particle physics experiment

1 Introduction

Digital image processing techniques emerged in the 1960s, being applied in fields such as medical science, observation of earth resources and astronomy. Since

then, its field of application has grown considerably, and particle physics experiments have been making intensive use of them [1].

Currently, gas detectors have proven to be a choice for detecting particles with low emission of energy. In these detectors, the light produced by the de-excitation of the gas molecules during the multiplication process of electrons can be captured by a camera based on CMOS (Complementary Metal-Oxide-Semiconductor) technology [2]. The acquired images can be processed for better pictorial information and for a more effective human interpretation. Moreover, additional information can be extracted and processed by classifiers for applications in pattern recognition and machine learning.

For improving the detection and classification of patterns in images, it might be necessary a good pre-processing, mainly on images that have a low signal-noise ratio. To evaluate the impact of any pre-processing technique, a simulated data-set may be essential to assess the potential of any proposed algorithm and, in general, as an aid in the search for solutions to mitigate the effect of noise on images.

This work proposes a study aiming at understanding the importance of filtering for the CYGNO experiment [3] regarding the improvement of the signal-to-noise ratio of its captured images, evaluating, in a first approach, the efficiency of detection of straight tracks using spatial filters. To accomplish this task, a simulation tool is also proposed.

The present paper is organized as follows: Sect. 2 describes the used data-set and the analysis methodology. The image generation procedure is explained in Sect. 3 and the filters definitions are presented in Sect. 4. Section 5 shows the relevant results and discusses the applicability of the method. Section 6 concludes this work.

2   Data-Set Definition and Analysis Methodology

In this section an overview of the data-set used in this work and of the analysis methodology are given.

2.1   Data Set

The Cygno experiment aims to develop a triple-GEM (Gaz Electron Multiplier) detector with combined read-out system by using an optical readout structure employing high granularity and low noise CMOS sensors to obtain a good-enough tracking performance for measuring low energy particles to search for Dark Matter massive particles. The detector is developed to read the light produced by the de-excitation of gas molecules during the processes of electron multiplication.

In this work we are using the data acquired by the ORANGE (Optically Readout GEm) prototype, which is described in details in [4–6]. A $10 \times 10\,\mathrm{cm}^2$ Triple GEM structure, with a 1 cm high drift gap, using a binary gas mixture He/CF_4 60/40), was readout by an Orca Flash 4 CMOS-based camera[1] equipped with a large aperture ($f/0.95$) lens, as shown in Fig. 1.

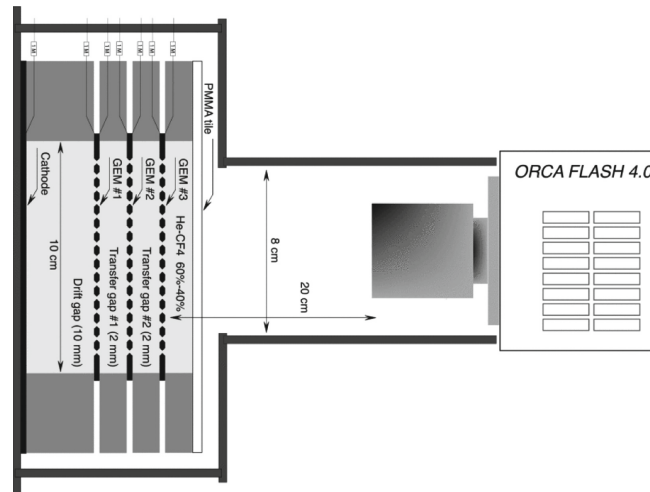[1] For more details visit the site www.hamamatsu.com.

Fig. 1. Drawing (not to scale) of the triple-GEM stack with the lens and the CMOS camera. The drift and transfer gaps are shown.

The images composing the data-set were acquired in free-running mode, without any trigger and a *AmBe* neutron source placed near to detector. Images produced by the GEM were recorded with an exposure of 2 s and all data were saved without any selection. Figure 2 shows two images generated by the detector for illustration.
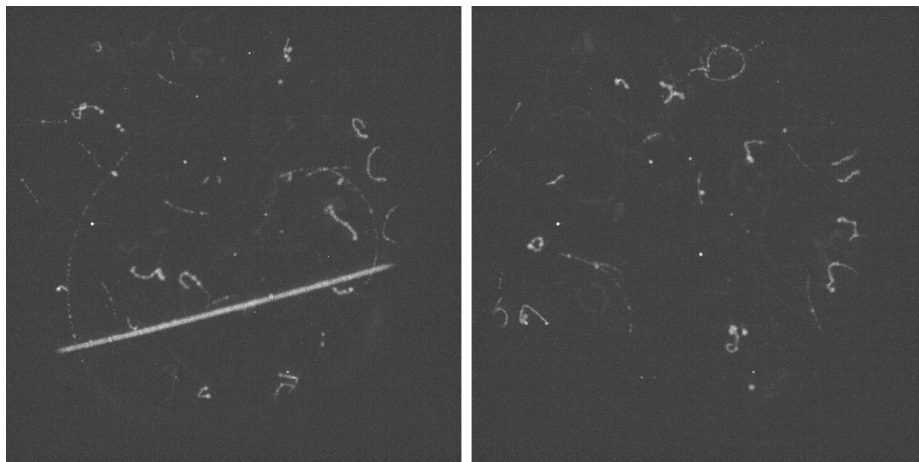

Fig. 2. Images acquired by the ORANGE prototype.

The tracks to be identified have different shapes and different intensities (luminosity). For this work, the main features of the straight tracks (signal) and of the noise have been extracted to study the images characteristics and to use them as input for the proposed simulation tool.

2.2   Analysis Methodology

The proposed analysis steps are summarized in Fig. 3. It can be divided into 4 steps, as follows:

– *Signal and background extraction*
  Using a real data-set acquired by ORANGE, the straight tracks and the image noise are studied and their main parameters characterized (see Sect. 3.1).
– *Image generation*
  In this step the tracks and noise are generated based on the parameters extracted from real data. For each track, its contrast is swept in order to produce a data-set which allows evaluating the filters performance for many levels of difficulty (see Sect. 3.2).
– *Image filtering*
  Filtering techniques are then applied to improve the signal-to-noise ratio of the images. For this work, some of the most used spatial filters (mean, median and Gaussian filters) are employed (see Sect. 4).
– *Performance evaluation*
  The last step is then, after performing a binarization of the image using a simple threshold level based on intensity, measure the number of signal pixels and the number of noise pixels with intensities above the threshold to produce two parameters: DE (detection efficiency) and FA (false alarm), respectively (false alarm might be seen as its counterpart known as background rejection, given by $BR = 1 - FA$).
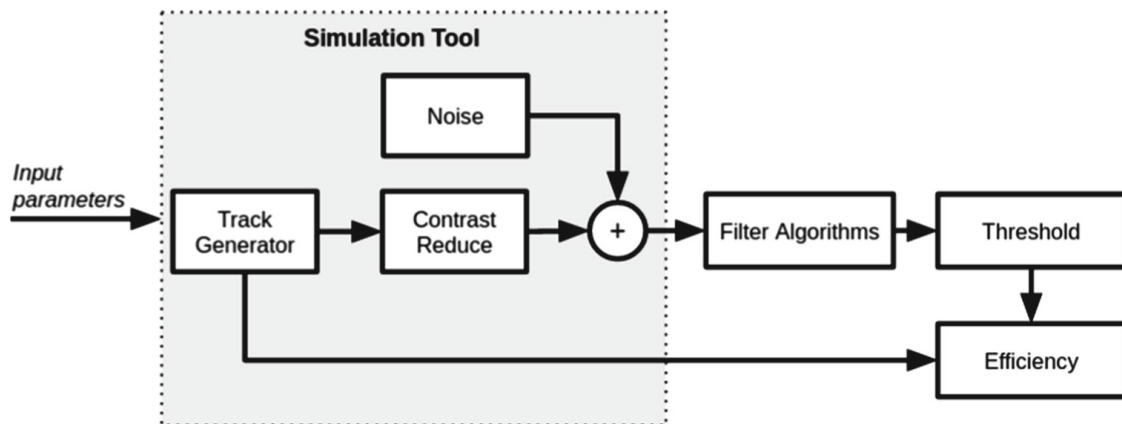

Fig. 3. Analysis steps.

3   Simulation Tool

In this section, the steps performed to obtain a first proposal of a simulation model based on the images acquired by the ORANGE prototype will be presented. The objective here would be only to create a tool that facilitates the

evaluation of any method of image processing that could be considered by the Experiment.

An image signal can be modeled as in Eq. 1, where I_t is the binary image representing the track signal whose length and width are given using Monte Carlo method, i is the track intensity (or contrast), and η represents the additive noise. In particular, i will be swept over the entire intensity range of the image to generate data-sets with tracks of varying intensities in order to evaluate the filtering algorithms at different levels of difficulty.

$$I_g(x, y) = iI_t(x, y) + \eta(x, y) \tag{1}$$

For the moment, only straight tracks are considered. Subsections 3.1 and 3.2 describe the procedure used to extract some of the main characteristics of the noise η and tracks I_t, respectively, and explain how they were employed to generate events within the simulation tool.

3.1   Noise Features Extraction and Generation

As usual, for any detection based experiment, to create a simulation tool, signal and noise need to be characterized. To extract the noise features, a data-set acquired without making use of any radioactive source was used. A luminosity histogram built using such data-set is shown in Fig. 4a. Studies on the tracks characteristics have shown that their minimum and maximum intensities are 85 and 135, respectively. Figure 4b shows the histogram considering only that interval. To model such noise, it was used the method known as KDE (Kernel Density Estimator) [7]. The resulting model was then used to generate noise values for the simulation tool. A histogram of a set of generated values are shown in the same Fig. 4b.

The luminosity values beyond the interval [85–135] were used to model a noise component known as salt-and-pepper [8]. A given pixel in the image has the probability of being part of the salt and pepper noise ($p_{s\&p}$). If it is the case, it
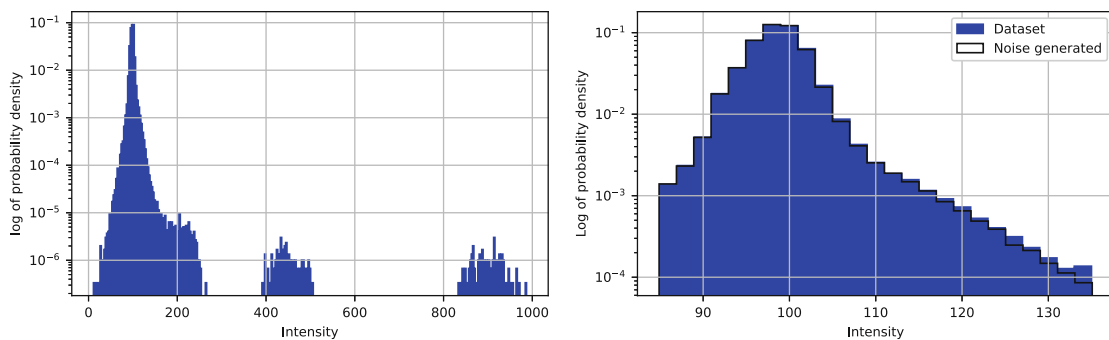
(a) Luminosity histogram from real data (b) Luminosity histograms from real and simulated data

Fig. 4. Luminosity histograms from real and simulated images.

can be modeled by the Bernoulli distribution where the parameter p_s represents the probability of the pixel to be salt and $p_p = (1 - p_s)$ the probability to be pepper. When a pixel is salt (pepper), its intensity value assumes the maximum (minimum) value on the considered image scale. The probability of a pixel to be a salt-and-pepper noise was measured by summing the number of events out of the interval [85–135] divided by total number of events in the data-set. The probably of a pixel to be salt (pepper) was given by the number of events over (under) 135 (85) divided by the total number of events in the data-set, normalized so that the sum of the two components equals one. The estimated values were: $p_{s\&p} = 0.0083$, $p_s = 0.16$, $p_p = 0.84$, in other words, the probability of a given pixel being degraded by salt-and-pepper noise is 0.83%, and given that it has occurred, the probability of that pixel being salt is 16% and of being pepper is 84%.

3.2   Signal Features Extraction and Generation

As mentioned before, for a first approach only straight tracks have been considered. The considered features were length and width, to be used as input for the simulation tool. Contrast will be given by sweeping the tracks intensities within the range [85–135]. The length and width distributions are presented in Fig. 5. The marginal of the measurements were modeled using KDE, considering that the dependence between length and width is low enough not to justify, at this stage, a two-dimensional density estimation. A Gaussian kernel has been chosen and the KDE bandwidth was determined by Silverman's formula [9].
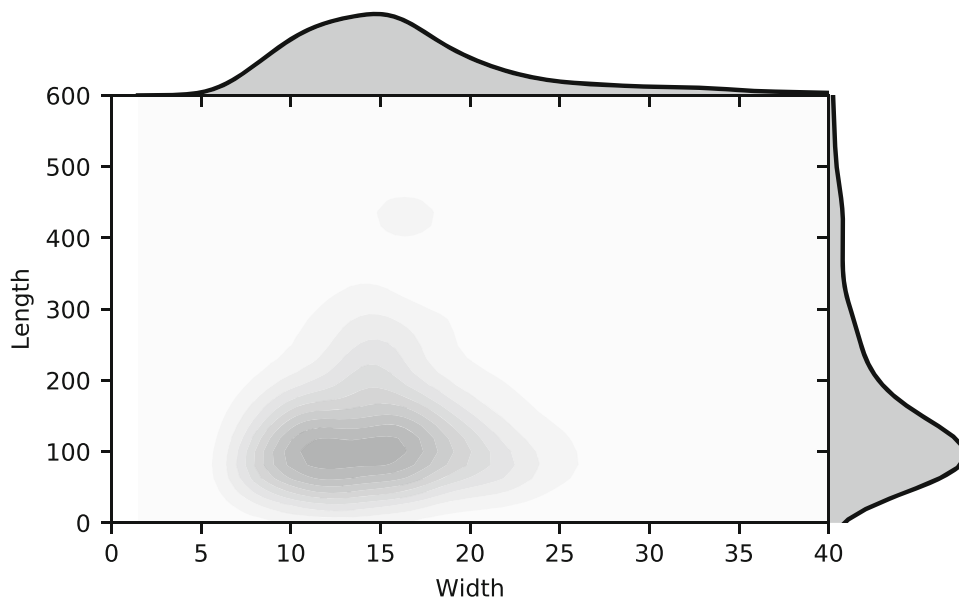

Fig. 5. Histograms of length and width of tracks

The next step is to generate the tracks using the length and width extracted from the estimated pdfs. The track generation is divided into 3 parts. The first

is to define a straight segment with length obtained by Monte Carlo and divide it into n equally spaced points.

The second part is to rotate these points by a random angle and centralize it in a random position. This is necessary due to the random orientation and position of the tracks. It is assumed that they are uniformly distributed within the image area. Then, after having drawn an angle θ from 0 to 2π and a central position point (x_0, y_0), the rotation/translation matrix showed in Eq. 2 is applied.

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} cos(\theta) & -sen(\theta) & x_0 \\ sen(\theta) & cos(\theta) & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{2}$$

The third part is to join these points to get a straight track. For that, a Gaussian radial base function [10] $K(\mathbf{x}, \mathbf{x}')$ given by Eq. 3 is used for each point defined previously.

$$K(\mathbf{x}, \mathbf{x}') = e^{-\frac{||\mathbf{x}-\mathbf{x}'||^2}{2\sigma^2}} \tag{3}$$

where the vector $\mathbf{x}$ defines positions on a Cartesian plane (x, y) and the vector $\mathbf{x}'$ defines the position of each of the n points that form the straight segment. The output K are made of values from ≈ 0 (far from $\mathbf{x}'$) to 1 (near to $\mathbf{x}'$). Each point has a function K, so the final output, after processing all the n points, is given by the sum of all functions K as can be shown in Eq. 4. As the Gaussian function is infinite, a threshold is necessary to define the track region (defined as 4σ from $\mathbf{x}'$).

$$\sum_{j=1}^{n} K_j(\mathbf{x}, \mathbf{x}'_j) = e^{-\frac{||\mathbf{x}-\mathbf{x}'_j||^2}{2\sigma^2}} \tag{4}$$

For simplifying the adjustment of track intensity, a variable normalized from 0 to 1 has been defined as shown in Eq. 5, where $i_{max} = 135$ (defined in Sect. 3.1) and $i_{med} = 99$ (which is the image noise baseline).

$$i = \alpha(i_{max} - i_{med}) + i_{med} \tag{5}$$

4   Filters Definition

Spatial filters act directly on the pixels of an image [11] by moving a filter mask thought out its full region. The response of a spatial filter for each point (x, y) is obtained by the relationship between the central pixel and its neighbors. A spatial filter can be linear or non-linear. For this work, two linear filters (mean and Gaussian) and a non-linear filter (median) are considered.

Linear Filter. In general, the output image $g(x,y)$ is given by a 2D-convolution between the system, defined by its impulse response $w(x,y)$ (also known as filter mask), and the input image $f(x,y)$, as given by Eq. 6 [1]. The sizes of w and f are defined as $a \times b$ and $M \times N$, respectively.

$$g(x,y) = \sum_{s=-a+1}^{a-1} \sum_{t=-b+1}^{b-1} w(s,t) f(x+s, y+t) \tag{6}$$

For the mean filter, the mask is defined by Eq. 7, where W is the size of the window. This filter is used to soften the image, reducing the effects caused by the presence of high-frequency components, reducing the variance of the image.

$$w(x,y) = \frac{1}{W} \tag{7}$$

For the Gaussian filter, the mask is described by Eq. 8 and its size is usually defined as a function of σ (e.g. $= 5\sigma$).

$$w(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \tag{8}$$

Non-linear Filter. This type of filter uses non-linear operations between mask and image. Some examples are the *max* and *min* operators that get the maximum and minimum values of a pixel neighborhood, respectively. The median filter replaces a given pixel by the median of all pixels in its neighborhood w, as given by Eq. 9 [1].

$$g(x,y) = median\{f(x,y), (x,y) \in w\} \tag{9}$$

5   Results

The main objective of this work is to verify if digital filters have the potential to produce relevant improvement in the signal-to-noise ratio on the ORANGE images. To generate the results, the following items were considered:

– Simulation data will be generated as described in Sect. 3;
– Track intensity will be swept by varying the parameter α used in Eq. 5 through out the interval $[0, 1]$, allowing to measure the performance of the applied filters for various levels of difficulty.;
– Tracks were divided into three categories according to their width values: slim (width < 10 pixels), medium ($10 \leq$ width ≤ 20) and thick (width > 20 pixels);
– The filters known as mean, Gaussian and median will be tested for a given range of mask window size;
– The best mask window size will be found for each filter given a fixed false alarm level of 1% (or, equivalently, a background rejection of 99%).

– Finally, the results will be given in terms of detection efficiency for a background rejection fixed to 99%, as mentioned above.

The left plot of Fig. 6 shows the detection efficiency for many values of α and window size for slim tracks only. As it is possible to notice, when the windows size gets large, the linear filters have their performance degraded, not happening the same with the median filter.

The right plot of Fig. 6 shows the best detection efficiency curves achieved by the filters (which means that their best window sizes are used) and the one achieved when raw-data is directly used, without any image processing. The detection improvement offered by the filters are clear and the median filter has presented the best results since it is immune to outliers and salt-and-pepper noise, followed by the Gaussian filter.
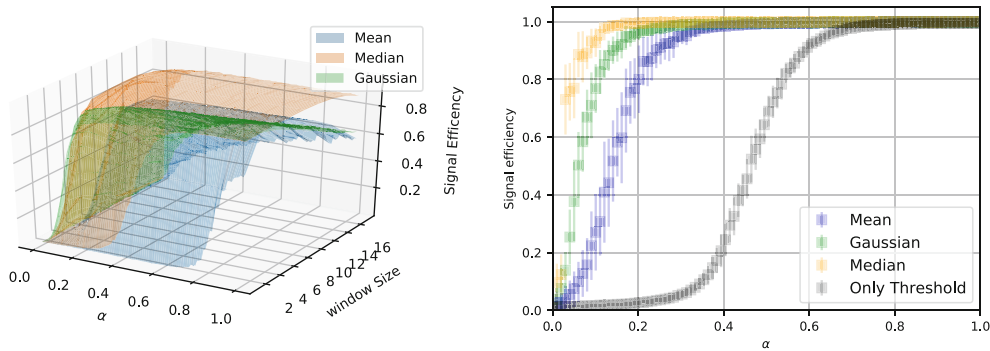

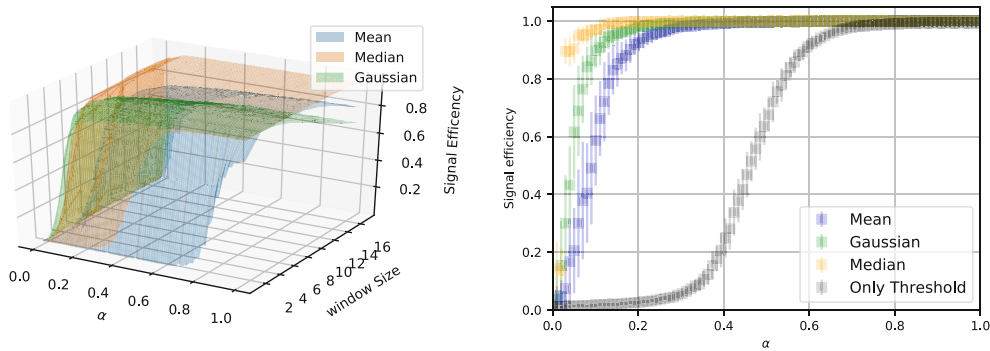**Fig. 6.** Detection efficiency for slim tracks.

Fig. 7. Detection efficiency for medium-width tracks.

The same analysis has been done for the medium-width and thick-width tracks as shown by Figs. 7 and 8. Similar conclusions can be made with slight differences:
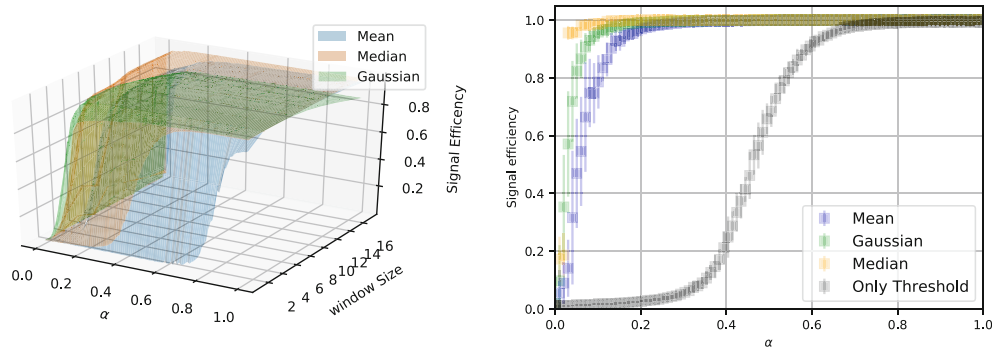
Fig. 8. Detection efficiency for thick-width tracks.

– The left plots show that the mean and Gaussian filters are less effected by the filter window size due to the thicker tracks width, but yet their performance are considerably degraded, differently from the median filter;
– The right plots show that the best curves offered by the filters get better (moving to the left and getting more pronounced) for larger track widths;
– The raw-data based method (only threshold) yields always the same performance, independent of track width.
– The detection efficiency curves achieved by the linear filters get closer.

6   Conclusions

This work presents an initial proposal for constructing a simulation tool for modeling the ORANGE images for evaluation of image processing algorithms. The simulation tool was used to verify the importance of applying digital filters to the CYGNO images for improving their signal-to-noise ratio.

Looking at the achieved results, it was possible to observe that the application of a pre-processing algorithm to the CYGNO image may be a way to improve the signal efficiency, even using a high background rejection requirement. These results justify further studies on the subject. Nonetheless, since this was a first attend, the simulation tool is been studied and upgraded to generated tracks even more closed to the reality of the experiment, in order to keep evaluating the already implemented filtering algorithm and add more complex algorithms.

References

1. Gonzalez, R.C., Woods, R.E., et al.: Digital Image Processing, vol. 141, no. 7. Publishing House of Electronics Industry (2002)
2. Marafini, M., Patera, V., Pinci, D., Sarti, A., Sciubba, A., Torchia, N.M.: Study of the performance of an optically readout triple-gem. IEEE Trans. Nucl. Sci. **65**(1), 604–608 (2018)
3. Pinci, D., et al.: Cygnus: development of a high resolution TPC for rare events. PoS, EPS-HEP2017, p. 077 (2017)
4. Marafini, M., Patera, V., Pinci, D., Sarti, A., Sciubba, A., Spiriti, E.: High granularity tracker based on a Triple-GEM optically read by a CMOS-based camera. JINST **10**(12), P12010 (2015)
5. Marafini, M., Patera, V., Pinci, D., Sarti, A., Sciubba, A., Spiriti, E.: ORANGE: a high sensitivity particle tracker based on optically read out GEM. Nucl. Instrum. Methods **A845**, 285–288 (2017)
6. Marafini, M., Patera, V., Pinci, D., Sarti, A., Sciubba, A., Spiriti, E.: Optical readout of a triple-GEM detector by means of a CMOS sensor. Nucl. Instrum. Methods **A824**, 562–564 (2016)
7. Scott, D.W., Sheather, S.J.: Kernel density estimation with binned data. Commun. Stat.-Theory Methods **14**(6), 1353–1359 (1985)
8. Chan, R.H., Ho, C.-W., Nikolova, M.: Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization. IEEE Trans. Image Process. **14**(10), 1479–1485 (2005)
9. Silverman, B.W.: Density Estimation for Statistics and Data Analysis. Routledge, Boca Raton (2018)
10. Fornberg, B., Larsson, E., Flyer, N.: Stable computations with Gaussian radial basis functions. SIAM J. Sci. Comput. **33**(2), 869–892 (2011)
11. Lim, J.S.: Two-Dimensional Signal and Image Processing, 710 p. Prentice Hall, Englewood Cliffs (1990)