IMAGE DENOISING TECHNIQUES FOR ASTRONOMICAL IMAGES

VALERIO ROSCANI
PHD STUDENT

COLLABORATORS:
CASTELLANO M.
FALCONE M.
FONTANA A.
MERLIN M.
TOZZA S.

EMAIL: VALERIO.ROSCANI@OA-ROMA.INAF.IT
INSTITUTION: ISTITUTO NAZIONALE DI ASTROFISICA
SAPIENZA - UNIVERSITÀ DI ROMA

FEBRUARY 6, 2018

OUTLINE

- Introduction

  From Analogic to Digital, Image type, CCD camera, PSF, Noise

- Denoising Algorithms

  Linear, Non-linear, Wavelets, Total Variation, , etc.

- Comparison Method

  Detection, Completeness and Purity functions, Comparison between algorithms

- Future Developments

  Other possible algorithms, Extended and Enhanced comparison method

INTRODUCTION – FROM ANALOGIC TO DIGITAL

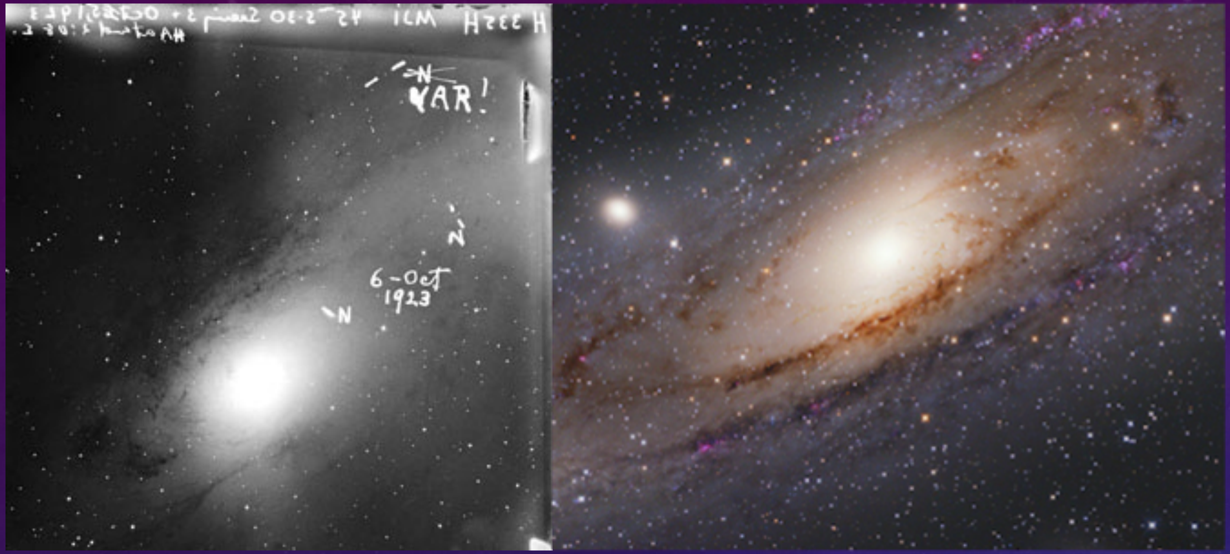
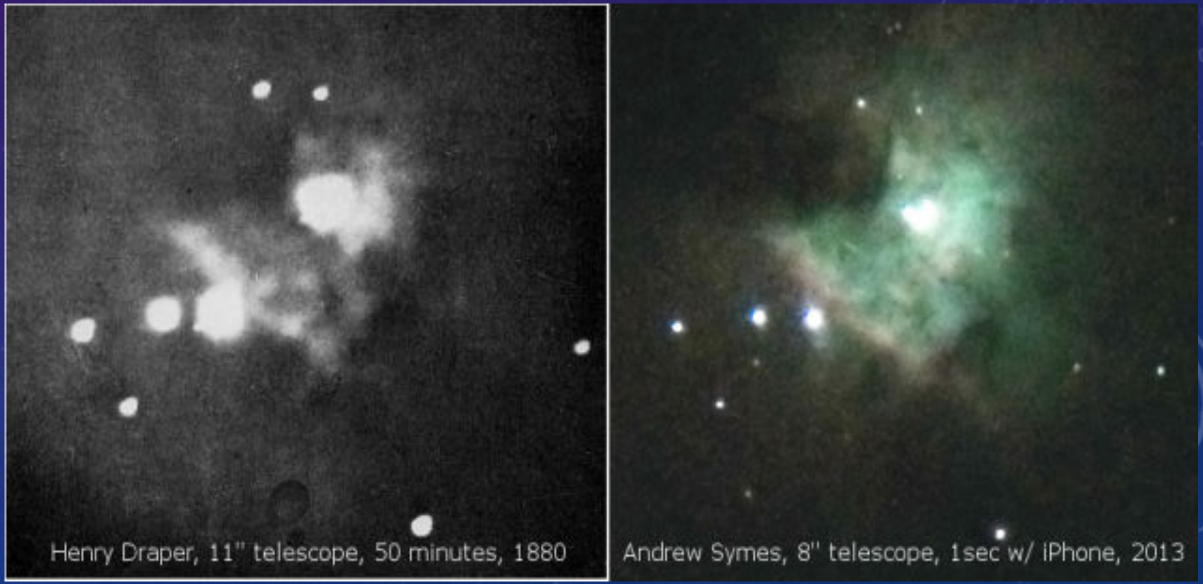


Image comparison: An analogic image of Andromeda galaxy, taken by Edwin Hubble in 1923, which totally revolutionized our idea of universe, followed by a modern digital image of the same galaxy taken by the Hubble Space Telescope.

Image comparison: "The First Photograph of a Nebula", 1880, photographic plate, followed by a digital image of the same nebula, taken by an iPhone camera + telescope 8''.



The evolution from analogic to digital provided a lot of advantages: more sensibility for each wavelength, better calibration of the instruments and an nearly full-automation of the analysis, no material decay, etc.

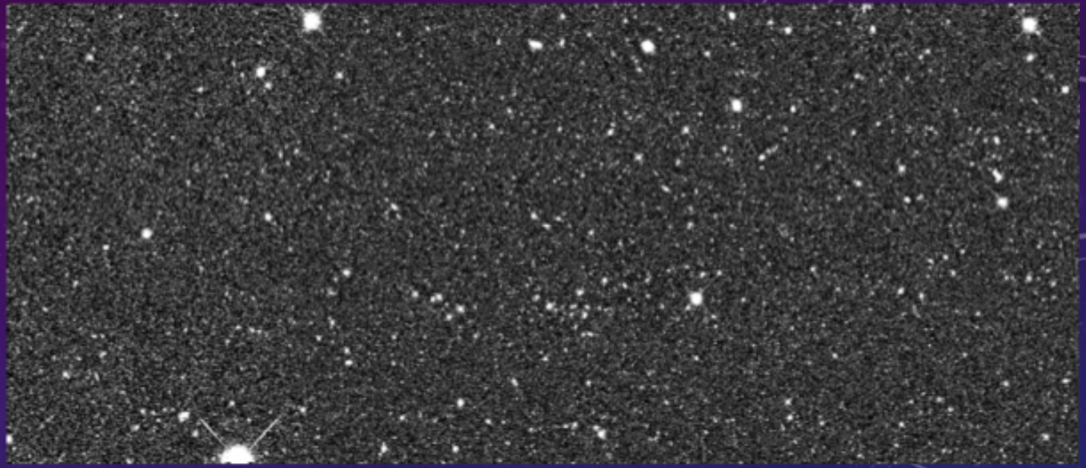
INTRODUCTION – IMAGE TYPE

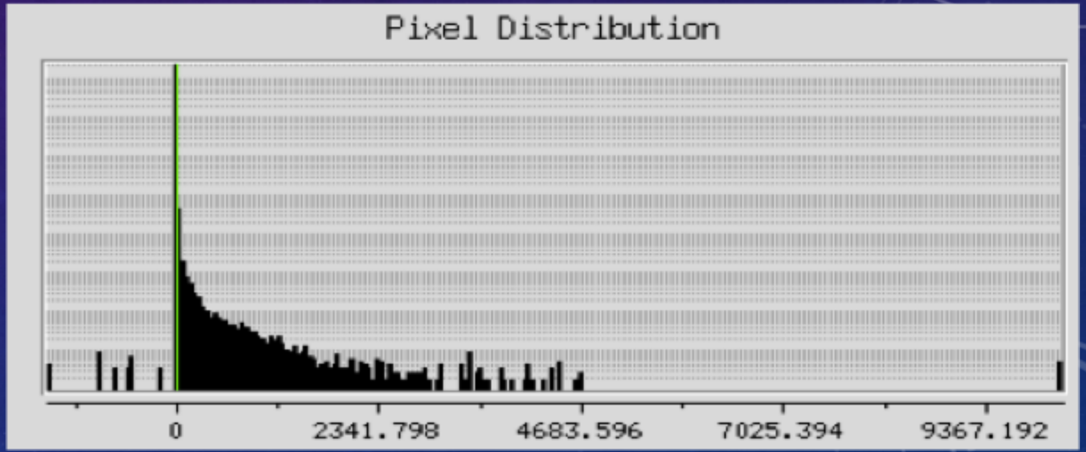
Flexible Image Transport System (FITS) Images is the standard format in astronomy, the header of the image can store useful information: coordinate systems, exposure time, wavelength, etc.

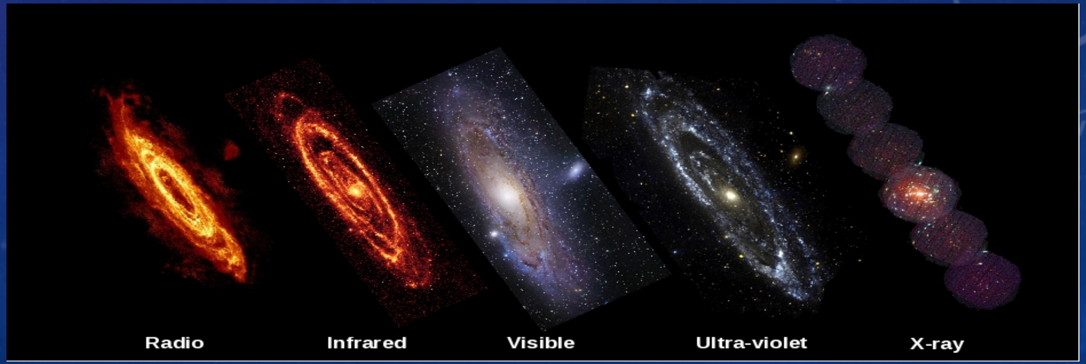
We are mainly interested in deep-field extragalactic images, taken by space telescopes and earth-based observations, studying this kind of images is fundamental in order to understand the origin of the universe, how the galaxies were formed and why the universe looks like the one we observe!

Extragalactic images principal characteristics:

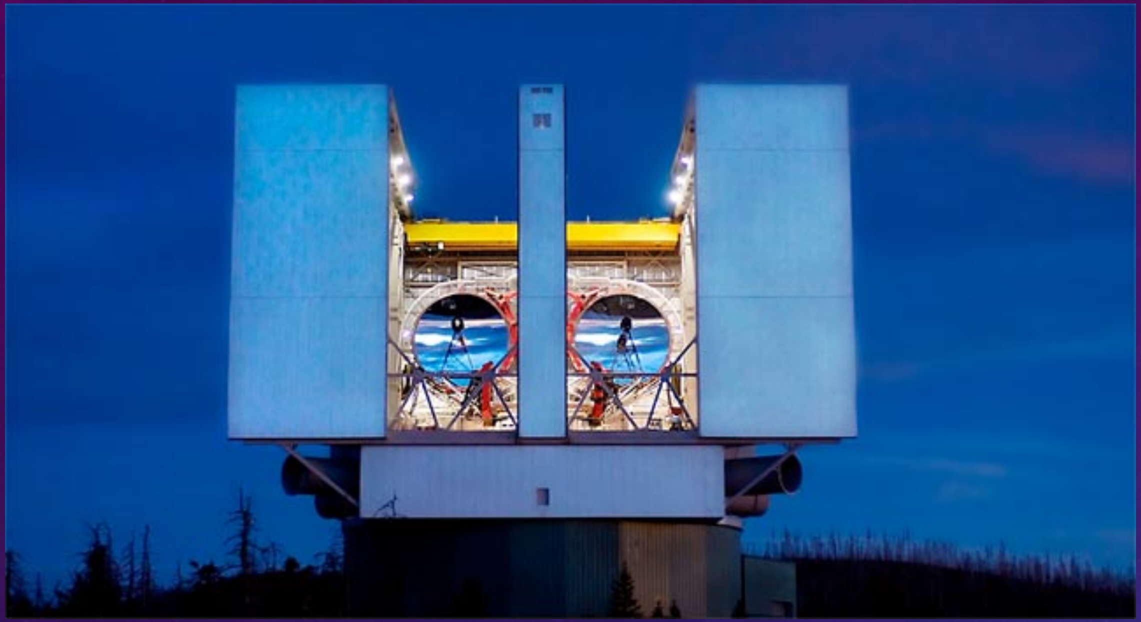
- Large number of objects in the image
- Large dynamical range of fluxes
- Large dynamical range of dimensions
- Multi-wavelength photons

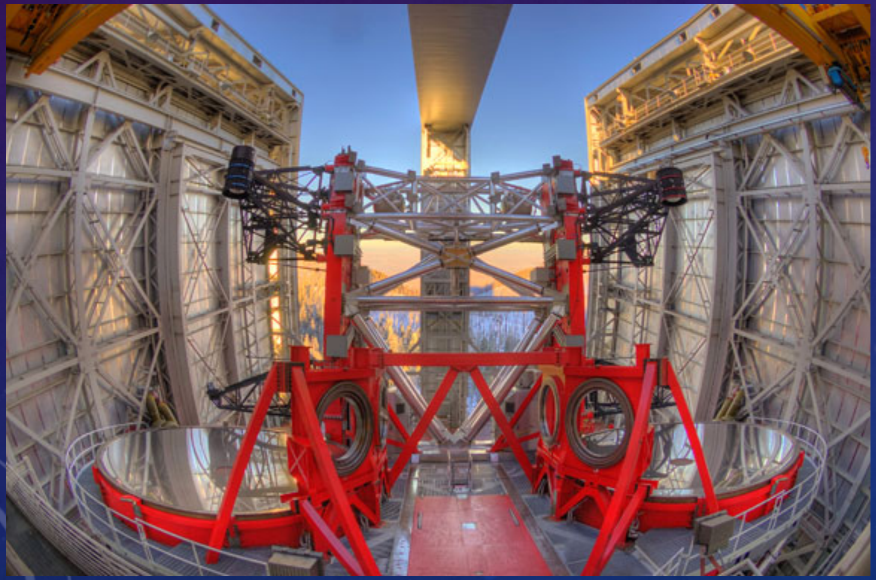






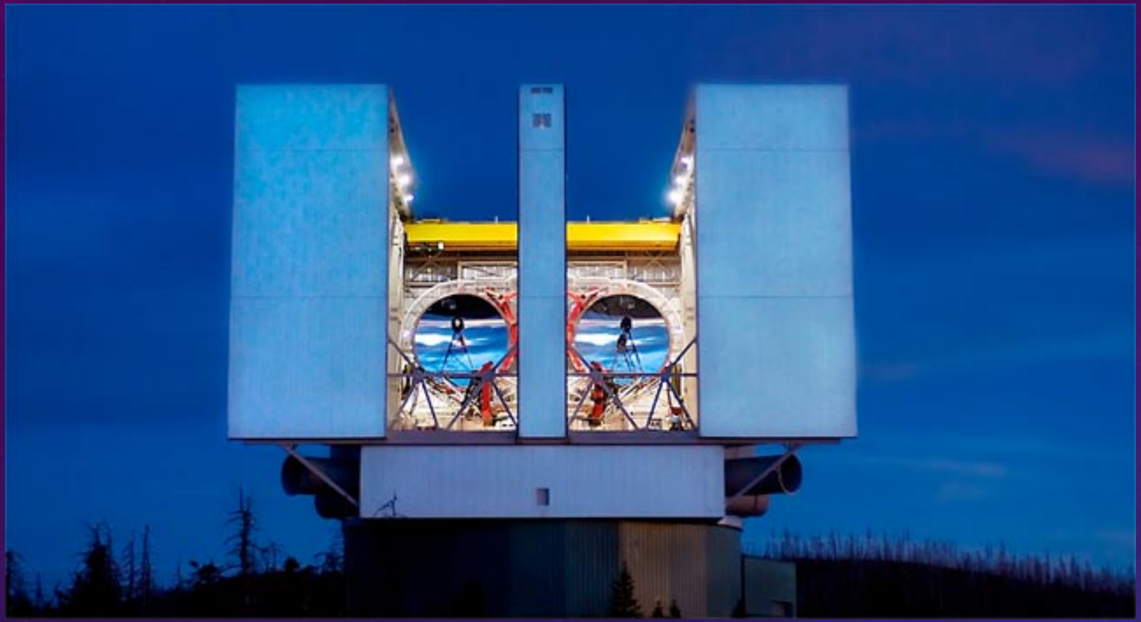
INTRODUCTION – IMAGE ACQUISITION



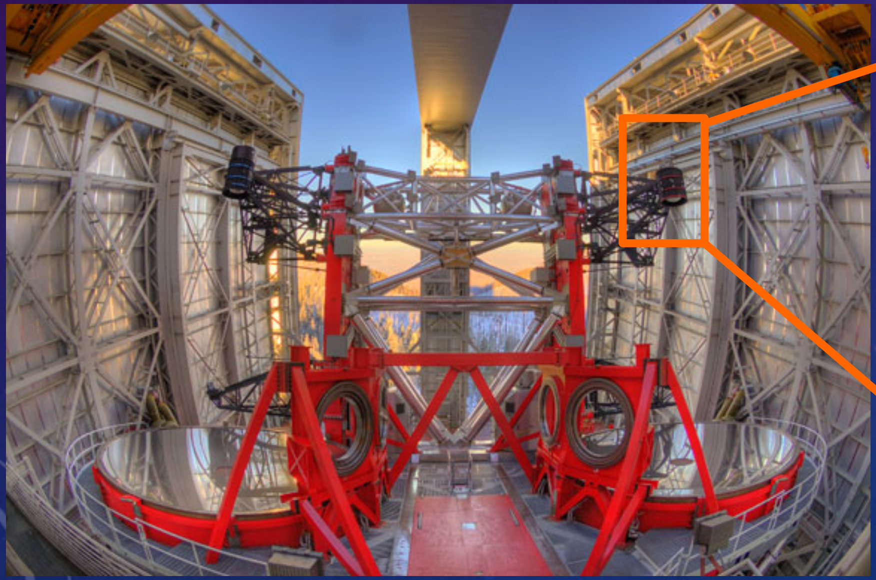


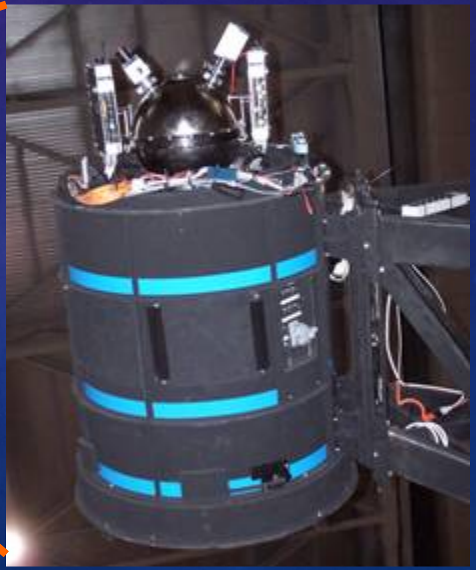
Large Binocular Telescope
Mt. Graham, Arizona
(Italy-INAF, USA, Germany)

INTRODUCTION – IMAGE ACQUISITION

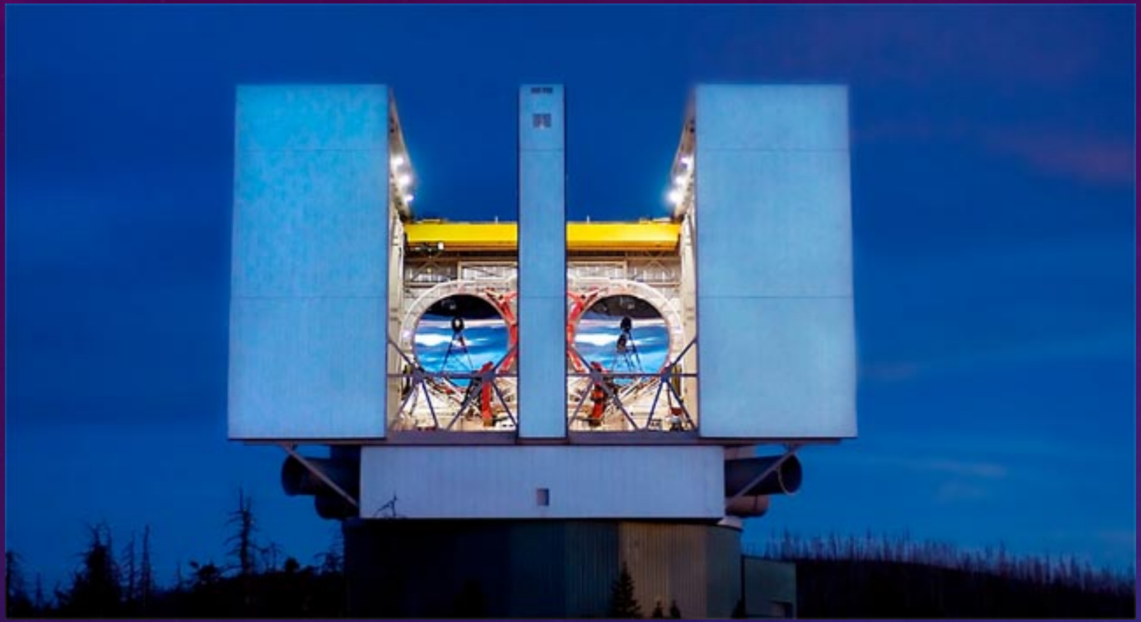


Large Binocular Telescope
Mt. Graham, Arizona
(Italia-INAF, USA, Germania)

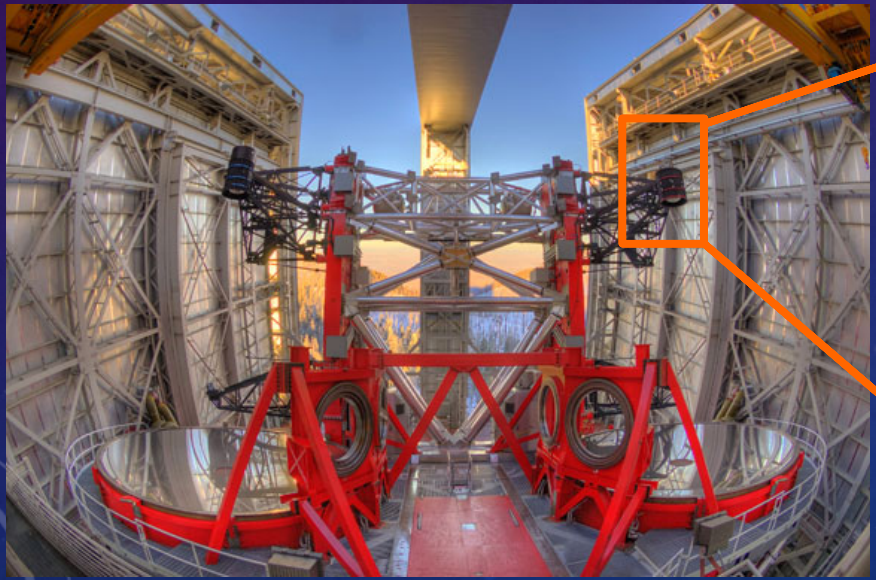


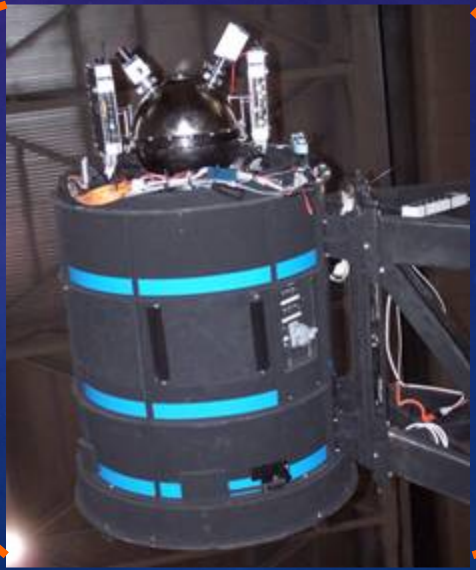


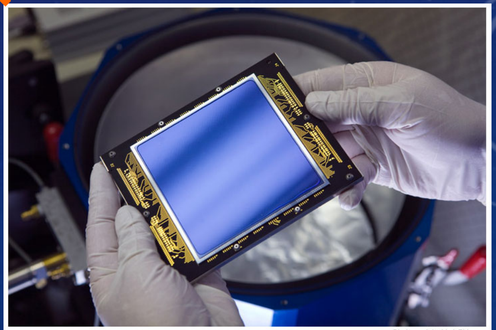
INTRODUCTION – IMAGE ACQUISITION



Large Binocular Telescope
Mt. Graham, Arizona
(Italia-INAF, USA, Germania)





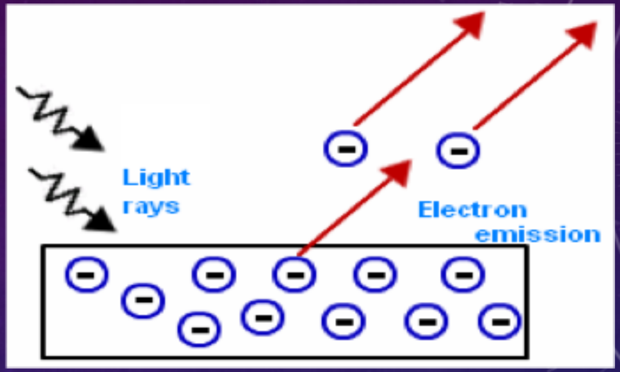


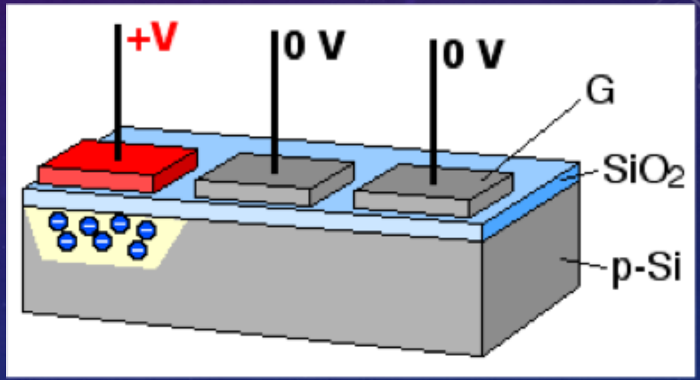
Photograph by Mark Thiessen

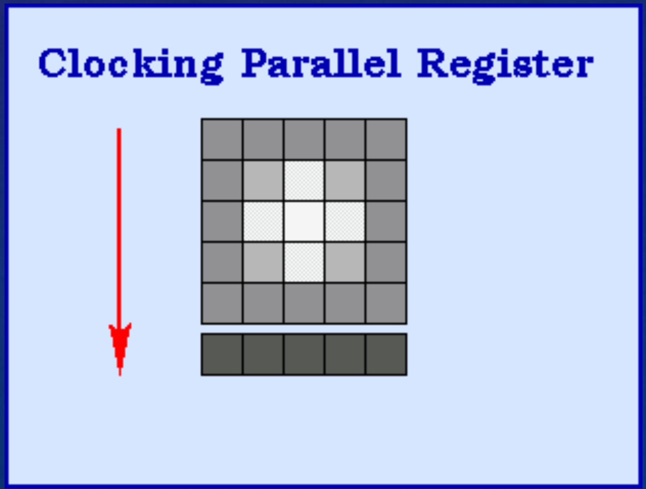
INTRODUCTION – CCD CAMERA

Astronomical digital images are produced by modern telescopes through a Charge-Couple Device (CCD)

- Photons striking a silicon surface create free electrons through the photoelectric effect.
- The electrons are gathered from the place they were generated by positively biasing discrete areas to attract them.
- A charge amplifier converts the charge into a signal, which is sampled and digitized.
- The signal is used to reconstruct the pixel matrix which corresponds to the image projected on the CCD surface.







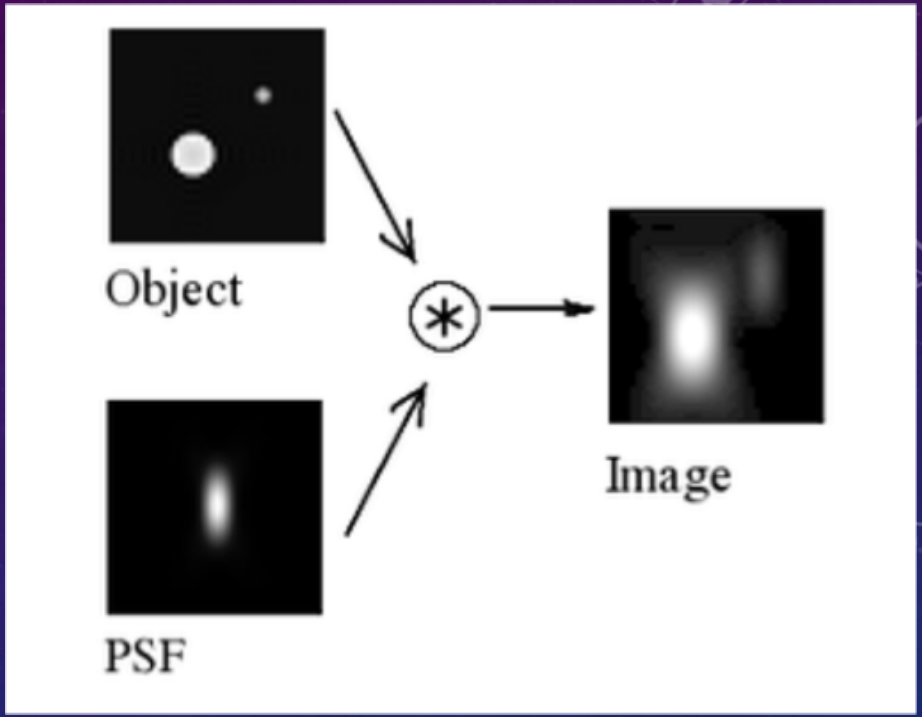
INTRODUCTION – PSF

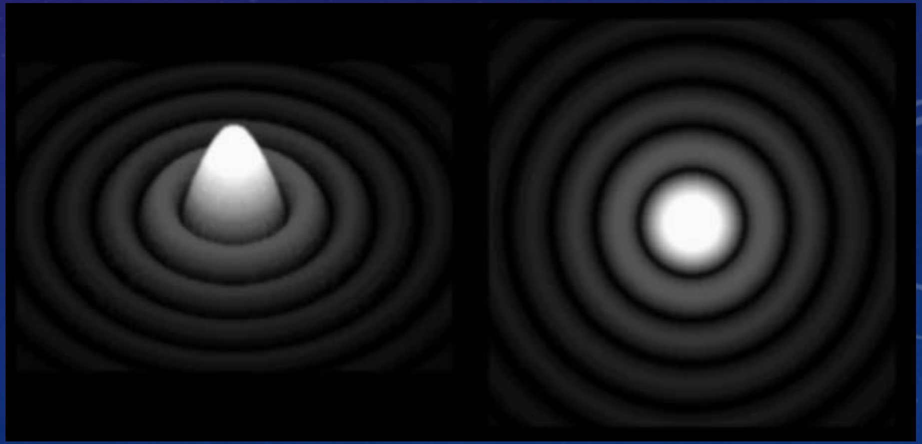
The **point spread function** (**PSF**) describes the response of an imaging system to a point-like object.

The image of an object can then be seen as a convolution of the true object and the PSF

The experimental determination of a PSF is often very straightforward due to the ample supply of point sources (stars and quasars). The form and source of the PSF may vary widely depending on the instrument and the context in which it is used.

The PSF for a perfect optical system, based on circular elements, would be an "Airy Pattern," which is derived from Fraunhofer diffraction theory

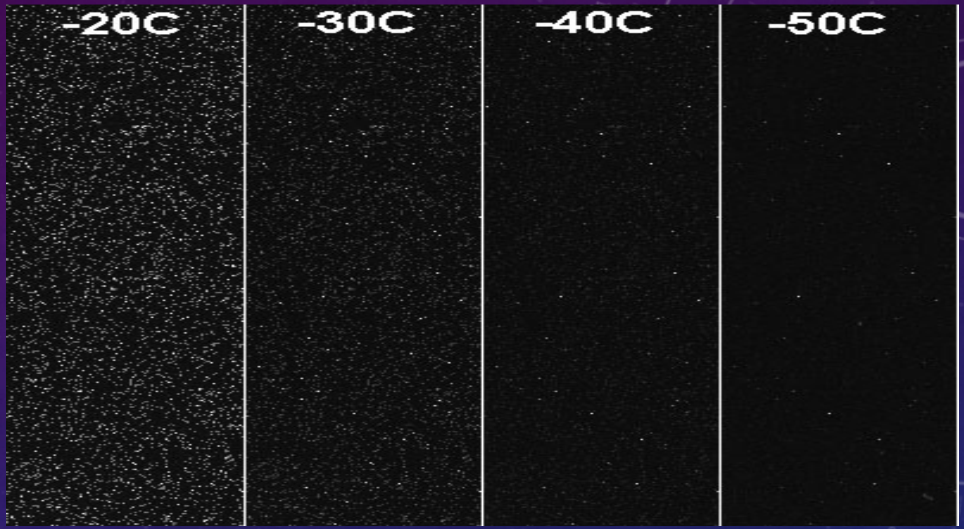




INTRODUCTION – NOISE IN ASTRONOMICAL IMAGES

- Thermal Noise: Some electrons can be generated by temperature oscillations, not correlated to any photon striking the CCD (Dark Current). Poisson Distribution.

  It can be subtracted to the final image, taking images with the frame closed.



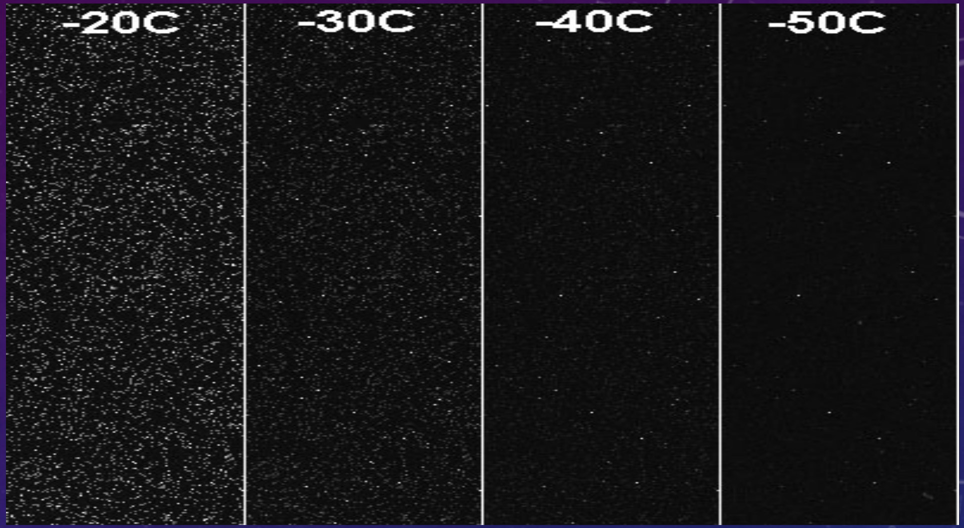
INTRODUCTION – NOISE IN ASTRONOMICAL IMAGES

- Thermal Noise: Some electrons can be generated by temperature oscillations, not correlated to any photon striking the CCD (Dark Current). Poisson Distribution.

  It can be subtracted to the final image, taking images with the frame closed.

- Pixel non-uniformity
  Each pixel has a slightly different sensitivity to light, typically within 1% to 2% of the average signal

  Pixel non-uniformity can be handled by calibrating the instrument with a flat-field image.



INTRODUCTION – NOISE IN ASTRONOMICAL IMAGES

- Thermal Noise: Some electrons can be generated by temperature oscillations, not correlated to any photon striking the CCD (Dark Current). Poisson Distribution.

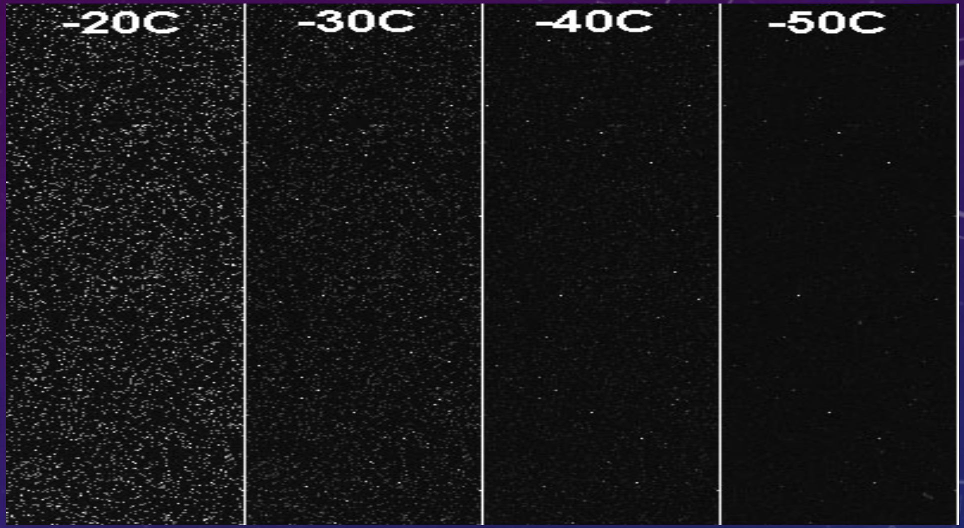
  It can be subtracted to the final image, taking images with the frame closed.

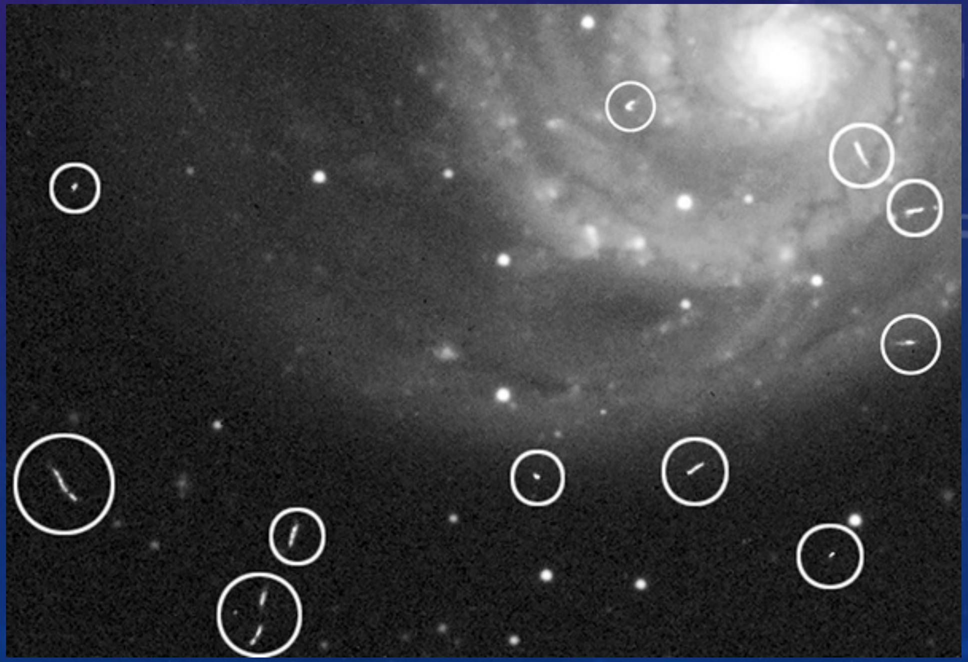
- Pixel non-uniformity
  Each pixel has a slightly different sensitivity to light, typically within 1% to 2% of the average signal

  Pixel non-uniformity can be reduced by calibrating an image with a flat-field image.

- Cosmic Rays: High energy radiation mainly originating from outside the solar system. Salt&"NoPepper"

  Easily recognizable, sharper shape than stars and galaxies. (Not properly noise)





INTRODUCTION – NOISE IN ASTRONOMICAL IMAGES

- Shot Noise: It is caused by the random arrival of photons. Each photon is an independent event, the probability of its arrival in a given time period is governed by a Poisson distribution

Shot Noise can be reduced by collecting more photons, either with a longer exposure or by combining multiple frames

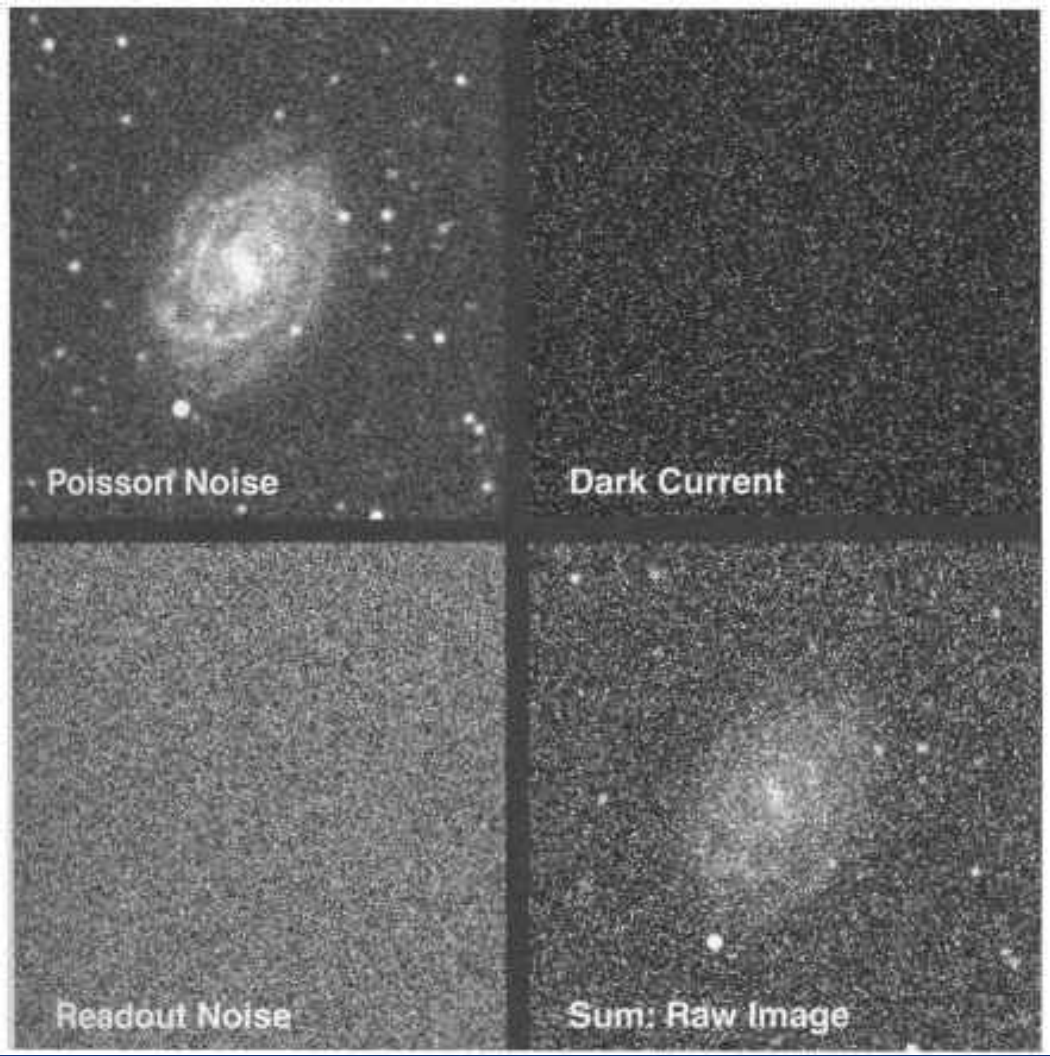
INTRODUCTION – NOISE IN ASTRONOMICAL IMAGES

- Shot Noise: It is caused by the random arrival of photons. Each photon is an independent event, the probability of its arrival in a given time period is governed by a Poisson distribution

  Shot Noise can be reduced by collecting more photons, either with a longer exposure or by combining multiple frames

- Readout Noise : It is a consequence of the imperfect operation of physical electronic devices. Basically, detectors which use an amplifier introduce this type of noise due to their non-perfect measurement of the charge in a clump of electrons.

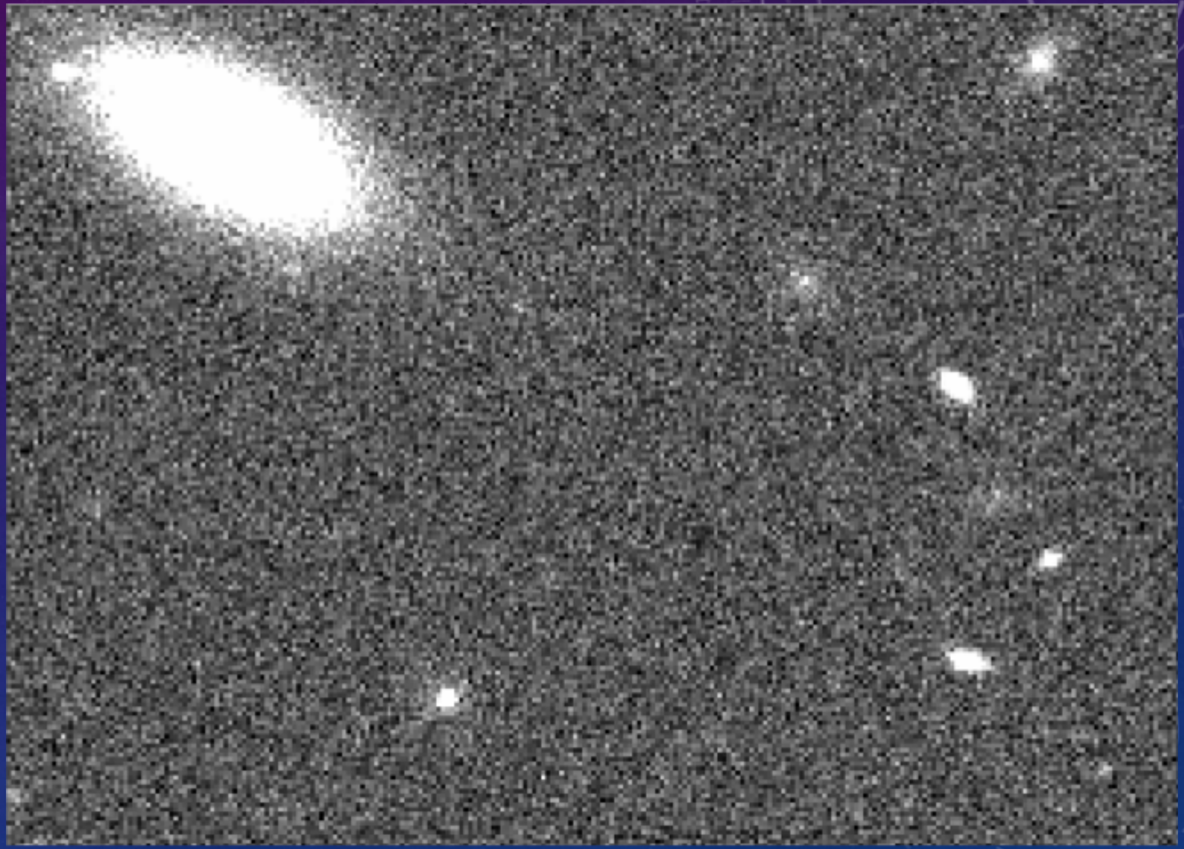
  Readout Noise exhibits a Gaussian distribution, can be reduced combining frames or with denoising techniques



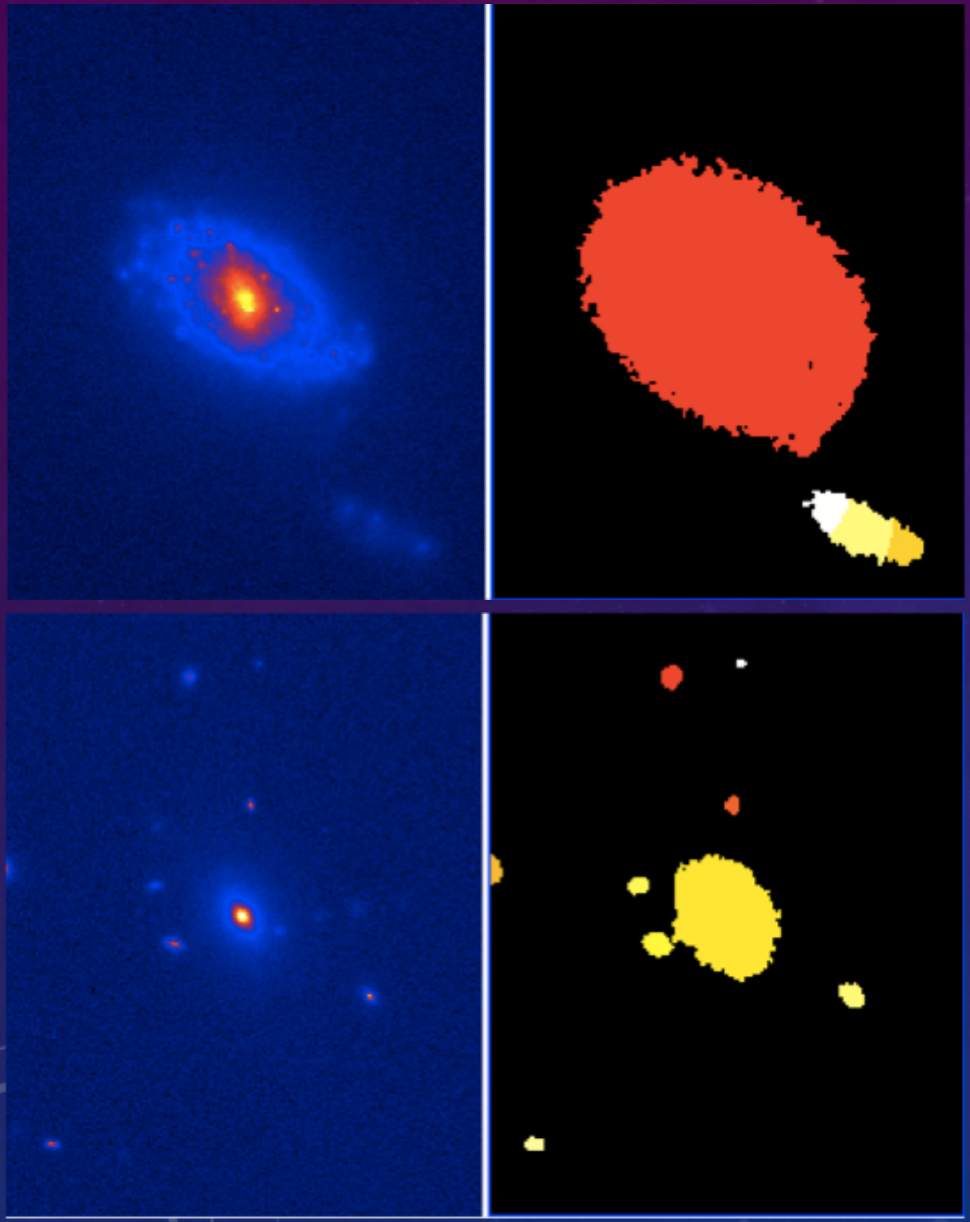
INTRODUCTION – NOISE IN ASTRONOMICAL IMAGES

- Background Noise: Produced by incoming light from an apparently empty part of the night sky. The great number of sources contribute to obscure the sources signals, setting a ”level of background”. The background noise is not simply approximated by a noise model, it is affected not only by faintest sources in the night sky, but also by distortions and emissions produced by the atmosphere.

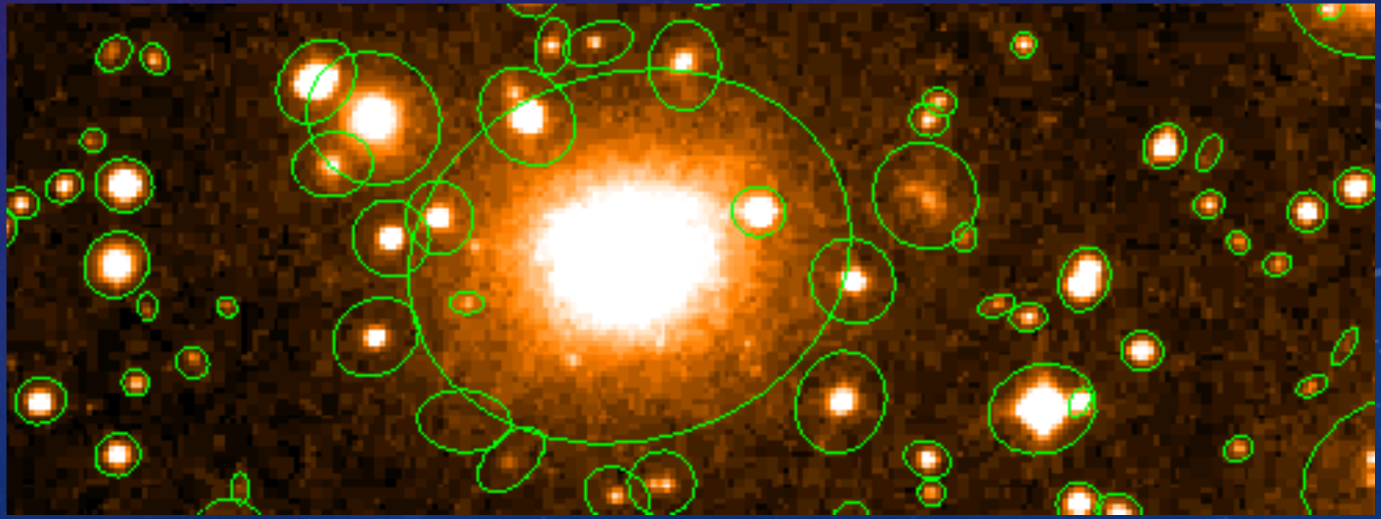
Background noise, for extragalactic images, is typically approximated by a Gaussian distribution, in absence of strong sources (e.g. stars).



SOURCE DETECTION AND IMAGE SEGMENTATION



Algorithms of source detection are extremely useful in astronomy in order to build catalogues of objects, with related features, fluxes, coordinates etc.
These algorithms estimate the background noise producing a root mean square map (rms) of the image.
A threshold for the detection is set to be N*σ times above the background rms.
Only the pixels which are above this threshold are considered real objects, (a minimum area of connected-above-threshold pixels is also considered, in order to avoid spurious detections).



DENOISING ALGORITHMS

Many methods of noise reduction exist. They are based on different approaches, and they behave differently depending on the kind of noise in the image.

- Linear Filters
- Non-linear Filters
- Statistical methods
- Non-local Means filter
- Total Variation algorithms
- Wavelet Transform
- Deep Learning
- Etc.

DENOISING ALGORITHMS

Gaussian Filter: Using the Gaussian function for calculating the transformation to apply to each pixel in the image

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

DENOISING ALGORITHMS

Gaussian Filter: Using the Gaussian function for calculating the transformation to apply to each pixel in the image

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

Unfortunately, this filter doesn’t prevent lose of contours, not only wipes out noise, but also image details.

DENOISING ALGORITHMS

Gaussian Filter: Using the Gaussian function for calculating the transformation to apply to each pixel in the image

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

Unfortunately, this filter doesn't prevent lose of contours, not only wipes out noise, but also image details.

Perona-Malik: An anisotropic diffusion algorithm, with partial differential equations. This technique aims at reducing image noise without removing significant parts of the image content, typically edges, lines or other details in the image

$$\begin{cases} \frac{\partial u}{\partial t} = \nabla \cdot (g(|\nabla u|)\nabla u) & \Omega \times (0, T) \\ \frac{\mathrm{d}u}{\mathrm{d}t} = 0 & \partial\Omega \times (0, T) \\ u(x, 0) = u_0(x) & \Omega \end{cases}$$

DENOISING ALGORITHMS

Gaussian Filter: Using the Gaussian function for calculating the transformation to apply to each pixel in the image

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

Unfortunately, this filter doesn't prevent lose of contours, not only wipes out noise, but also image details.

Perona-Malik: An anisotropic diffusion algorithm, with partial differential equations. This technique aims at reducing image noise without removing significant parts of the image content, typically edges, lines or other details in the image

$$\begin{cases} \frac{\partial u}{\partial t} = \nabla \cdot (g(|\nabla u|)\nabla u) & \Omega \times (0,T) \\ \frac{\mathrm{d}u}{\mathrm{d}t} = 0 & \partial\Omega \times (0,T) \\ u(x,0) = u_0(x) & \Omega \end{cases}$$

$$g(|\nabla u|)$$

Very important are the edge-stopping functions, using the gradient of the image can partially preserve contours

DENOISING ALGORITHMS

Edge-stopping functions that we have considered an tested:
c_1 and c_2, proposed by (Perona and Malik, 1990)

$$c_1(|\nabla I|) = \exp\left(-\left(\frac{|\nabla I|}{K}\right)^2\right)$$

$$c_2(|\nabla I|) = \frac{1}{1+\left(\frac{|\nabla I|}{K}\right)^2}$$

Tukey's biweight function (c_3) proposed by (Black et al.1998)

$$c_3(|\nabla I|) = \begin{cases} \frac{1}{2}\left[1-\left(\frac{|\nabla I|}{S}\right)^2\right]^2 & ; \ |\nabla I| \le S \\ 0 & ; \ \text{otherwise} \end{cases}$$

where $S=K\sqrt{2}$.

c_4 proposed by (Zhichang Guo et.al 2012)

$$c_4(|\nabla I|) = \frac{1}{1+\left(\frac{|\nabla I|}{K}\right)^{\alpha(|\nabla I|)}}$$

$$\alpha(|\nabla I|) = 2 - \frac{2}{1+\left(\frac{|\nabla I|}{K}\right)^2}$$

c_5 proposed by (J.Weickert, 1998)

$$c_5(|\nabla I|) = \begin{cases} 1-\exp(-3.31488 * K^8 / (|\nabla I|)^8) & ; \text{If} \ (|\nabla I|) \neq 0 \\ 1 & ; \text{Otherwise} \end{cases}$$

DENOISING ALGORITHMS

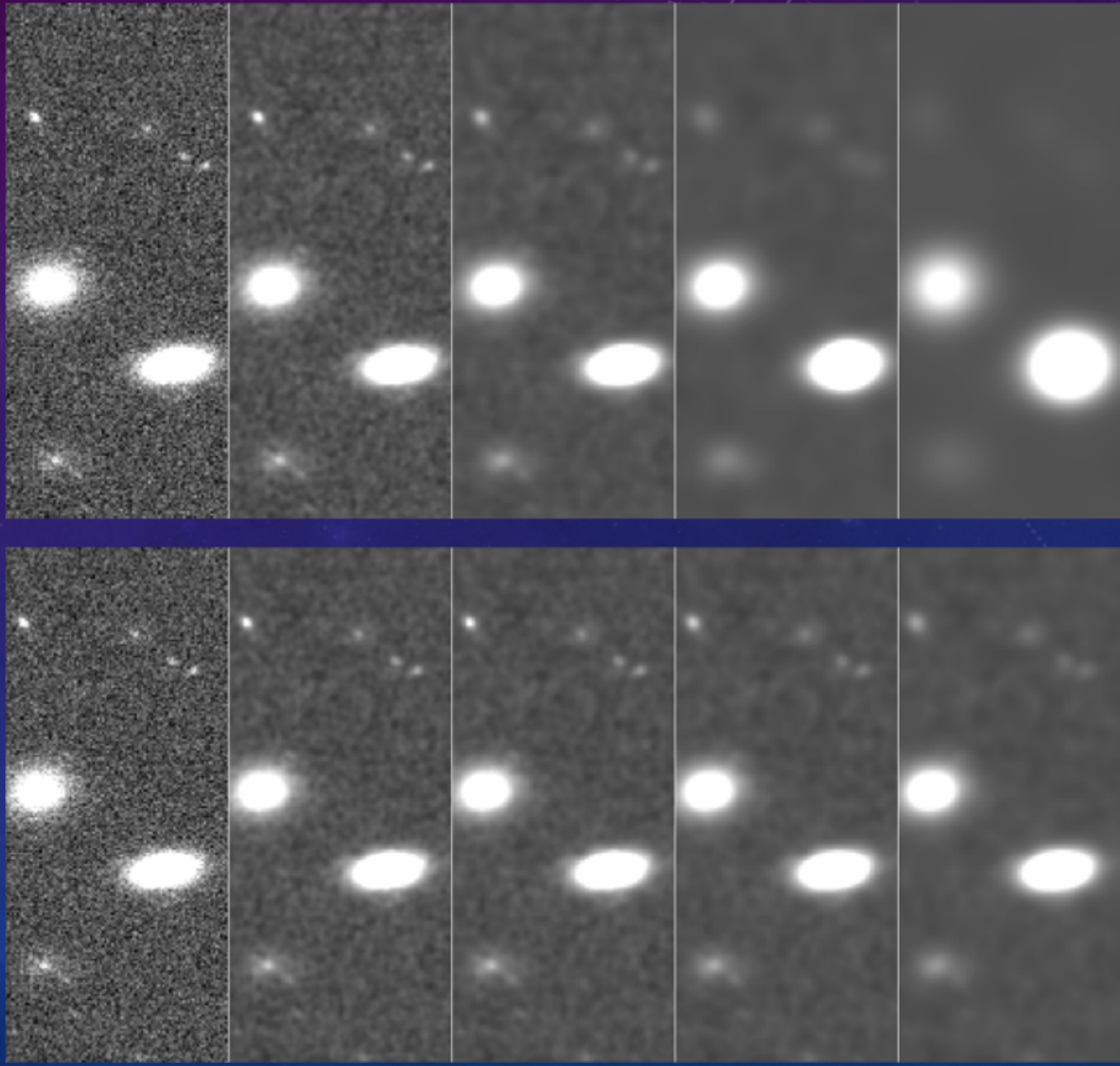
Gaussian Filter: Using the Gaussian function for calculating the transformation to apply to each pixel in the image

Unfortunately, this filter doesn't prevent lose of contours, not only wipes out noise, but also image details.

Perona-Malik: An anisotropic diffusion algorithm, with partial differential equations. This technique aims at reducing image noise without removing significant parts of the image content, typically edges, lines or other details in the image

Very important are the edge-stopping functions, using the gradient of the image can partially preserve contours

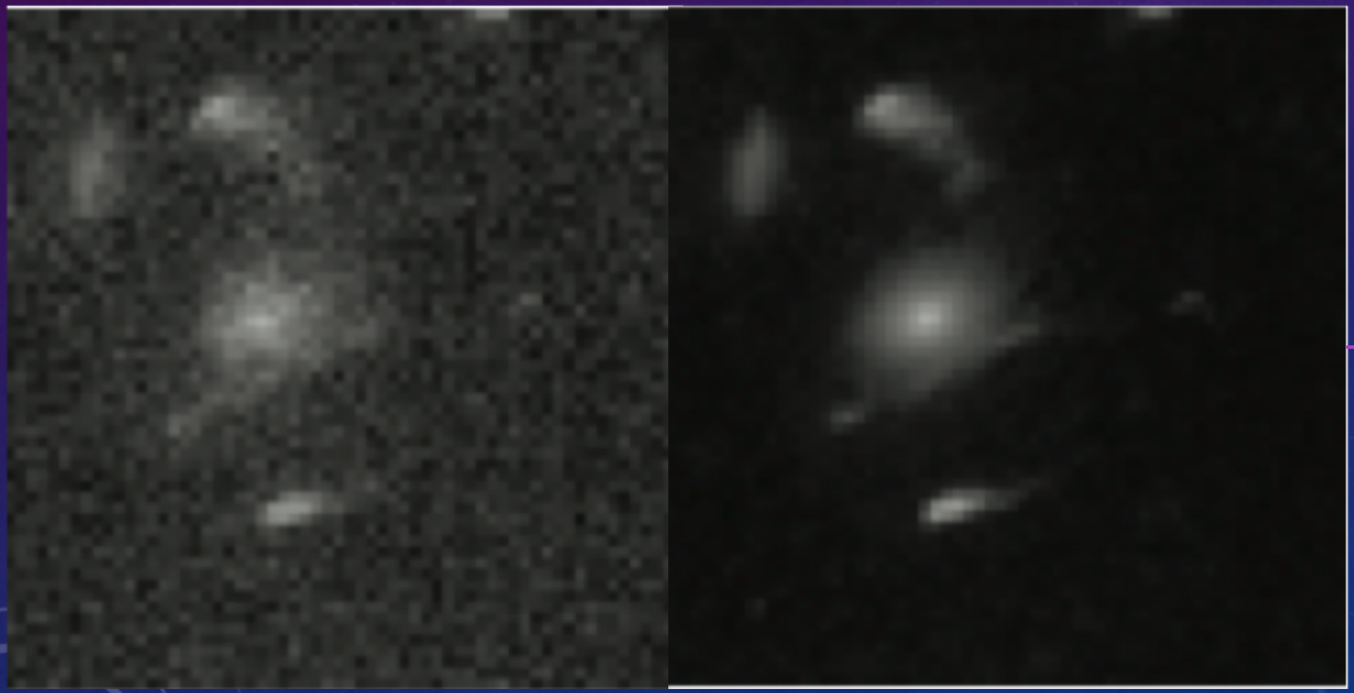
Integration Time →



DENOISING ALGORITHMS

Wavelet transform: It is a multi-scale approach, an image can be decomposed into components at different scales, well-adapted to the study of astronomical data.
It has been widely used in astronomical data analysis during the last ten years.

Works well independently on the type of noise
It represents well isotropic features, but it is far from optimal for analyzing anisotropic objects



DENOISING ALGORITHMS

Astro-Total Variation Denoiser (ATVD)

D.Ottaviani   Castellano et al. (2015)

$$inf\left(\int |Du| + \lambda||v||_X^p\right)$$

u is the structural part

v is the texture part

λ is the splitting parameter

X is a function space (e.g. L^2, L^1, L^p)

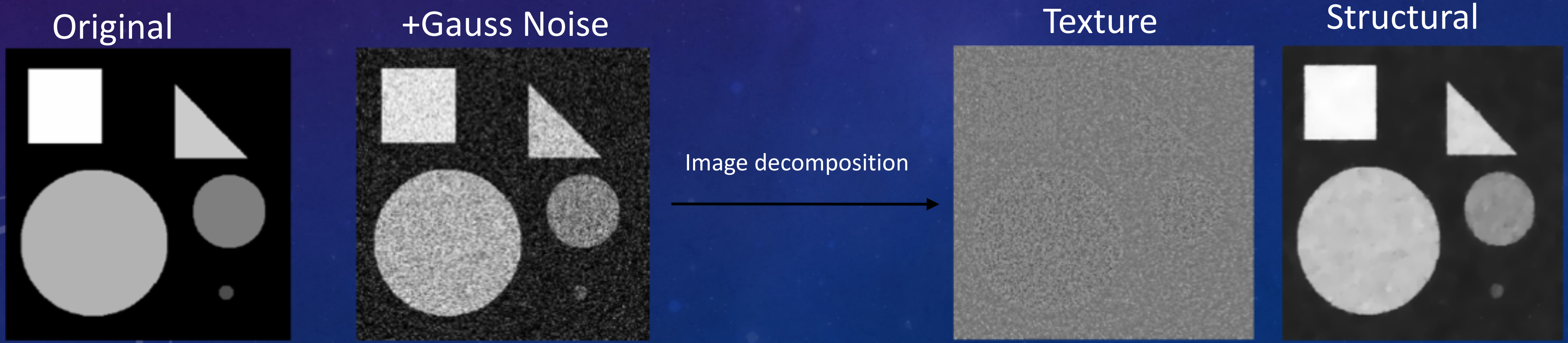
DENOISING ALGORITHMS

Astro-Total Variation Denoiser (ATVD)

D.Ottaviani   Castellano et al. (2015)

$$inf\left(\int |Du| + \lambda||v||_X^p\right)$$

u is the structural part
v is the texture part
λ is the splitting parameter
X is a function space (e.g. L^2, L^1, L^p)



DENOISING ALGORITHMS

Astro-Total Variation Denoiser (ATVD)

D.Ottaviani Castellano et al. (2015)

$$inf\left(\int |Du| + \lambda ||v||_X^p\right)$$

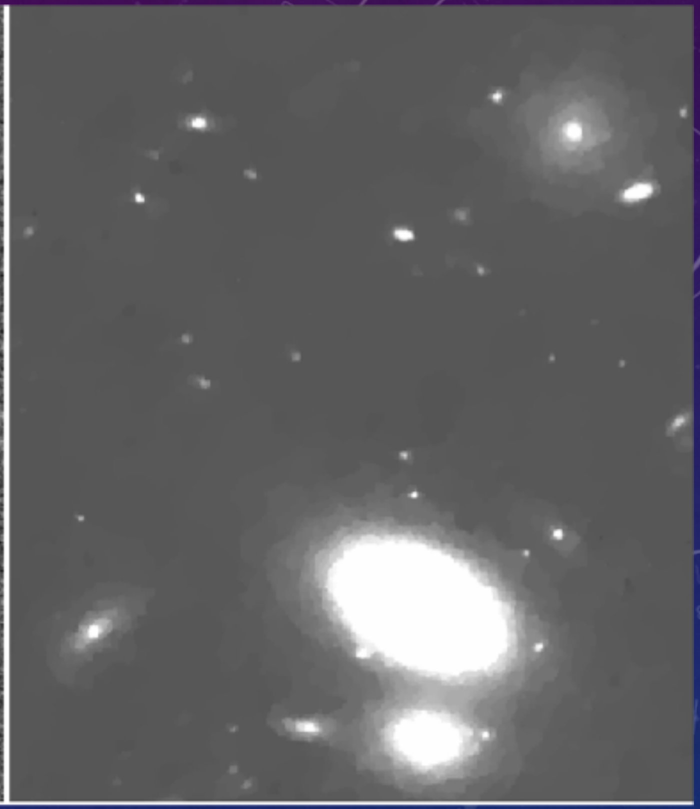
u is the structural part

v is the texture part

λ is the splitting parameter

X is a function space (e.g. L^2, L^1, L^p)

Original



Denoised

Original

+Gauss Noise

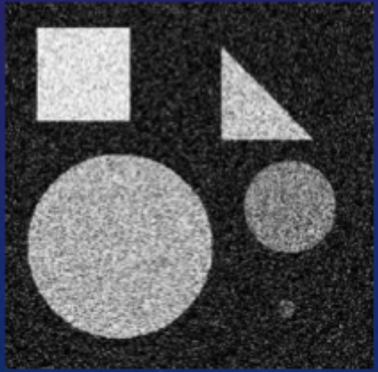
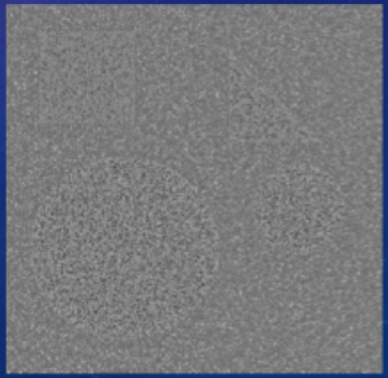


Image decomposition

Texture



Structural

DENOISING ALGORITHMS

Astro-Total Variation Denoiser (ATVD)

D.Ottaviani Castellano et al. (2015)

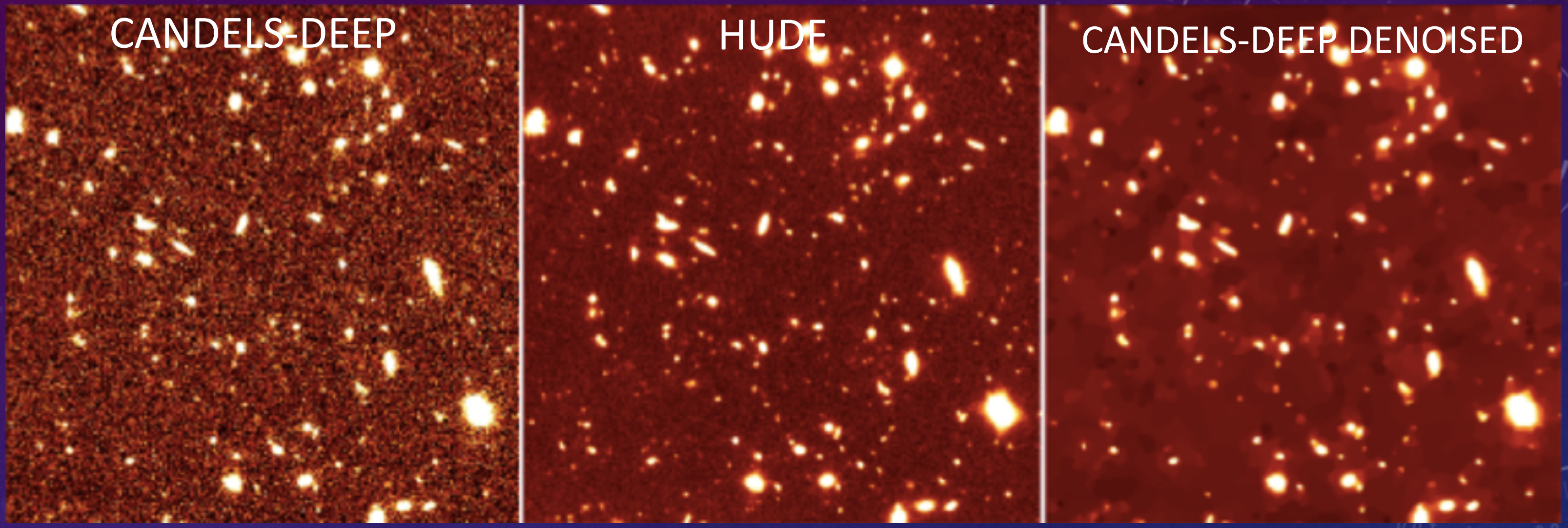
$$inf\left(\int |Du| + \lambda||v||_X^p\right)$$

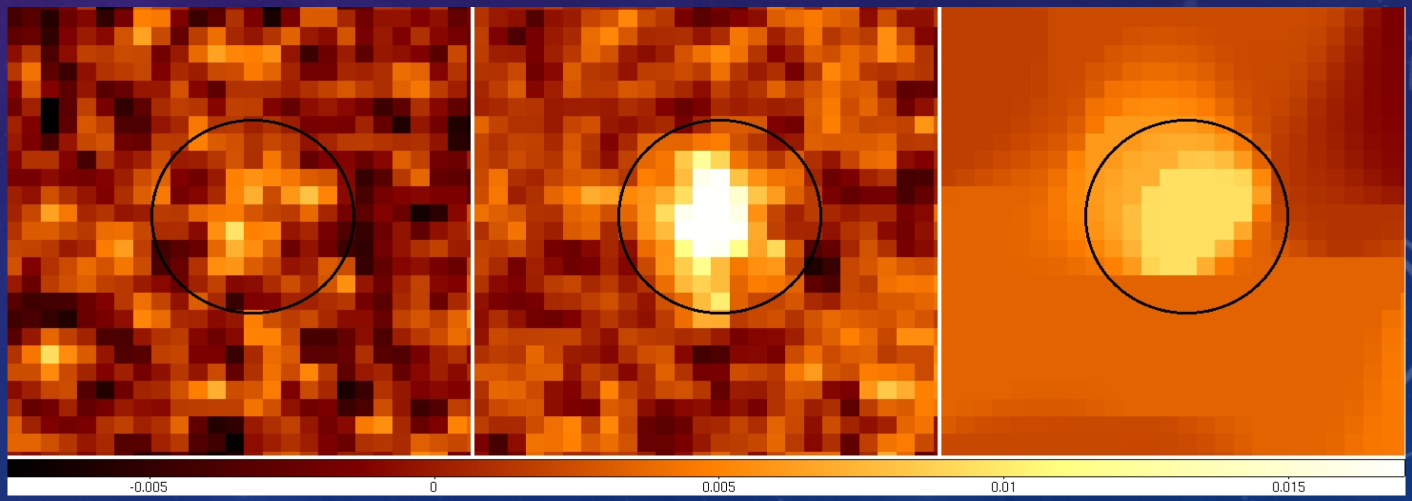
u is the structural part

v is the texture part

λ is the splitting parameter

X is a function space (e.g. L^2, L^1, L^p)



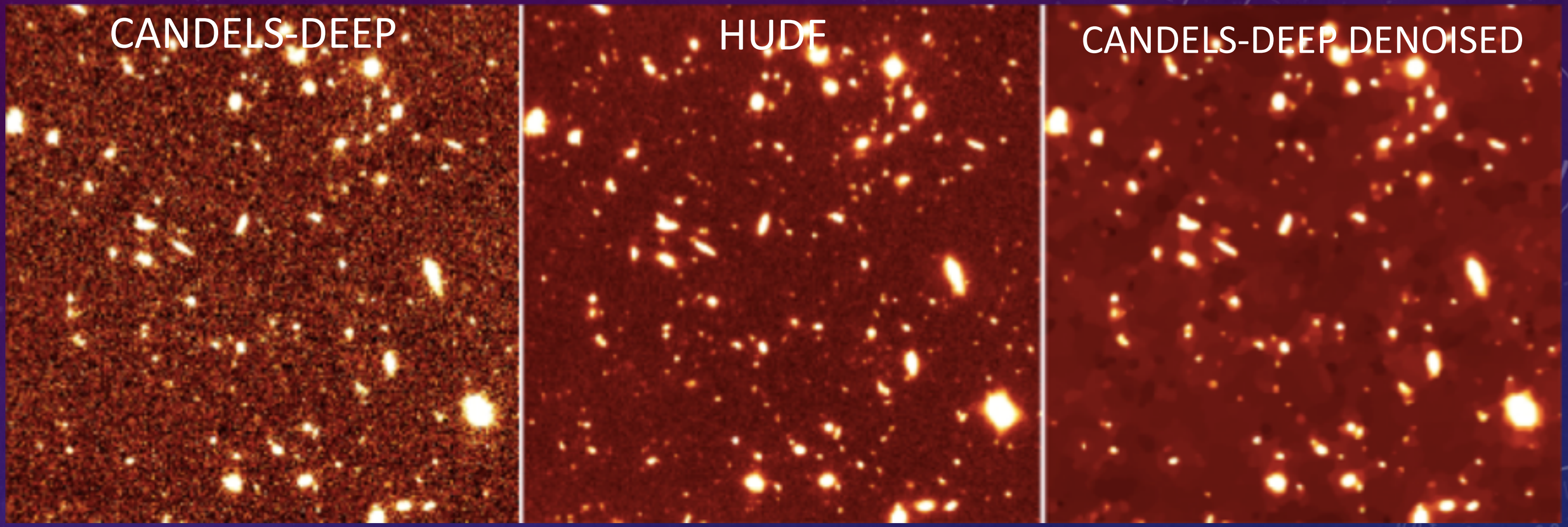


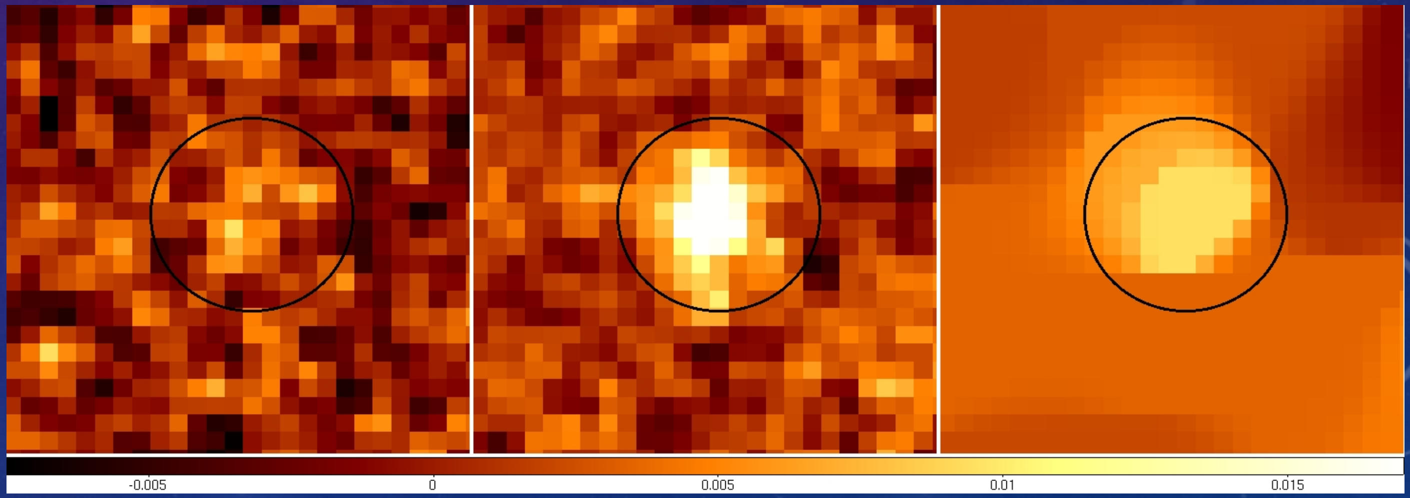
DENOISING ALGORITHMS

Astro-Total Variation Denoiser (ATVD)

D.Ottaviani   Castellano et al. (2015)

$$inf\left(\int |Du| + \lambda||v||_X^p\right)$$





u is the structural part

v is the texture part

λ is the splitting parameter

X is a function space (e.g. L^2, L^1, L^p)

Results are really interesting, although it is much slower than a Perona-Malik

COMPARISON METRIC

How do we compare the efficiency of these denoising algorithms?

COMPARISON METRIC

How do we compare the efficiency of these denoising algorithms?

Not a trivial issue: A good metric should consider an improvement in the number of detected objects without introducing spurious sources.

In real astronomical images, we are not able distinguish with extreme precision what is a real object and what is just a fluctuation of the background.

COMPARISON METRIC

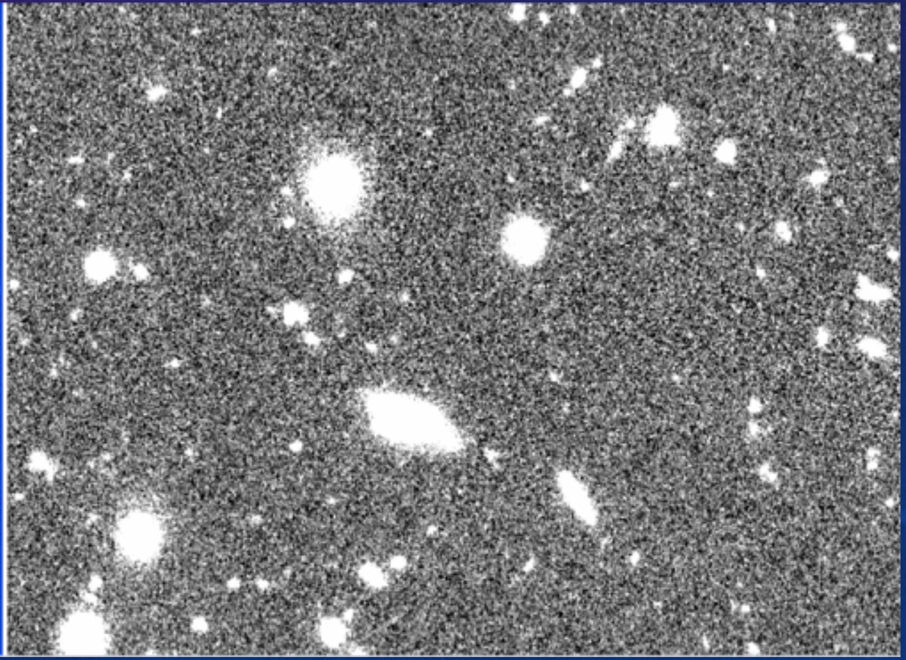
How do we compare the efficiency of these denoising algorithms?

Not a trivial issue: A good metric should consider an improvement in the number of detected objects without introducing spurious sources.

In real astronomical images, we are not able distinguish with extreme precision what is a real object and what is just a fluctuation of the background.

For this reason we need to consider simulated images.

Simulated



Real

COMPARISON METRIC

How do we compare the efficiency of these denoising algorithms?

Using a detection algorithm on a simulated image, can provide us useful information:

1. We know the exact number of objects
2. We know the coordinates of these objects
3. We know the exact flux of every object

COMPARISON METRIC

How do we compare the efficiency of these denoising algorithms?

Using a detection algorithm on a simulated image, can provide us useful information:

1. We know the exact number of objects
2. We know the coordinates of these objects
3. We know the exact flux of every object

Let’s define these quantities:

- N_{real} = real number of simulated objects
- $N_{detected}$ = number of objects detected by a detection algorithm
- $N_{spurious}$ = number of objects detected by a detection algorithm, that we know for sure are not real

In order to define $N_{spurious}$ we need define methods, which allow us to understand if the object detected is spurious or not

COMPARISON METRIC

How do we compare the efficiency of these denoising algorithms?

Association Method: We know which are real objects, every detected object has to be associated to a real object, if not: that object is considered spurious

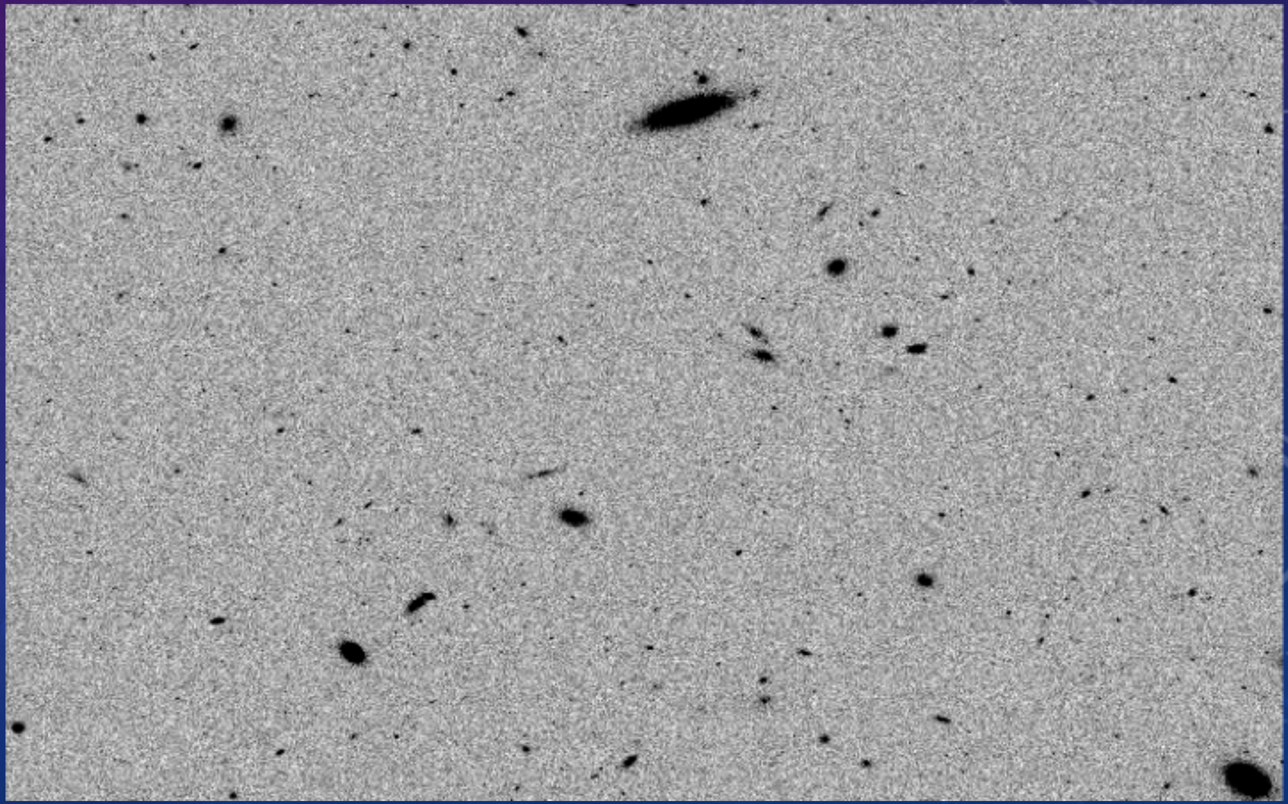
COMPARISON METRIC

How do we compare the efficiency of these denoising algorithms?

Association Method: We know which are real objects, every detected object has to be associated to a real object, if not: that object is considered spurious

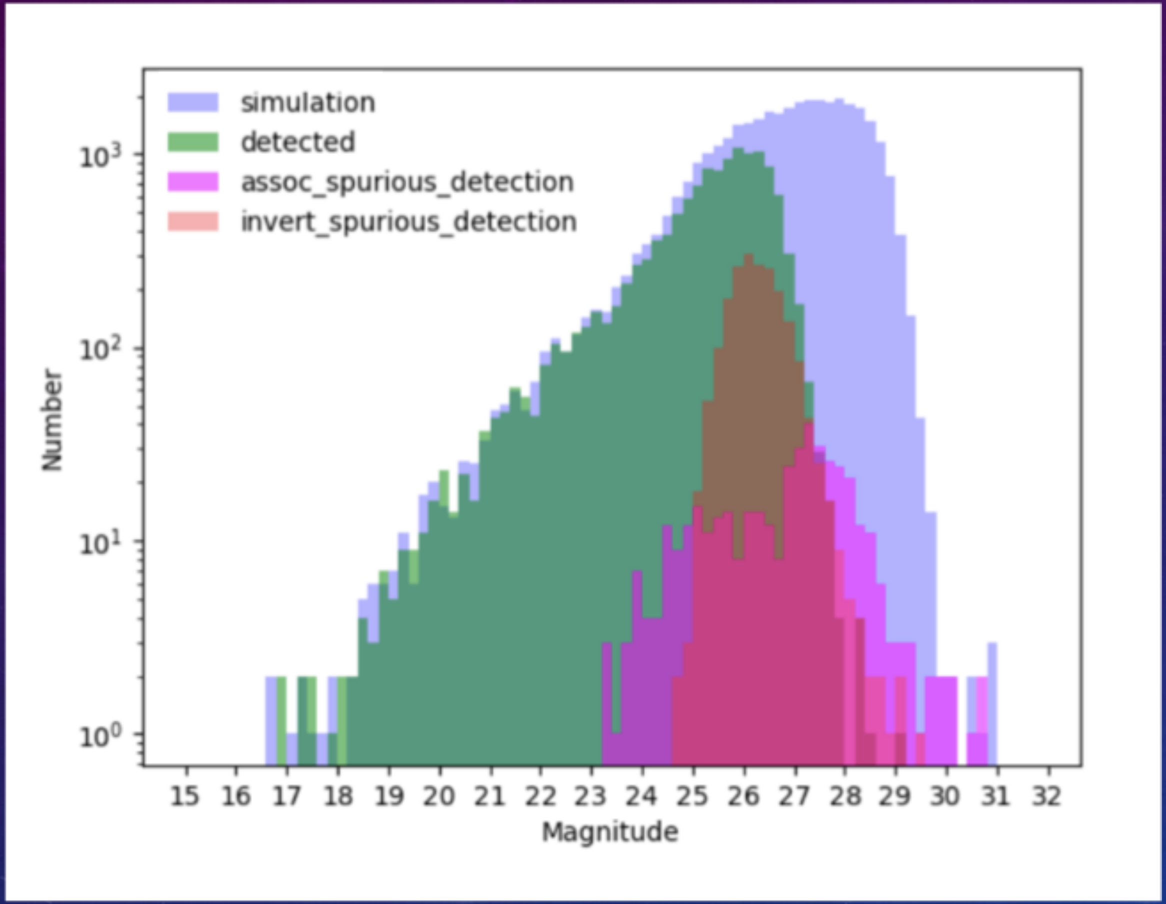
Inverted Image Method: This method is based on the assumption that, noise is equally distributed at negative and positive values (following a Gaussian distribution).

Inverting the image means that real objects become negative, while noise can be negative or positive, running the detection algorithm estimates how much spurious objects we are detecting



COMPARISON METRIC

How do we compare the efficiency of these denoising algorithms?

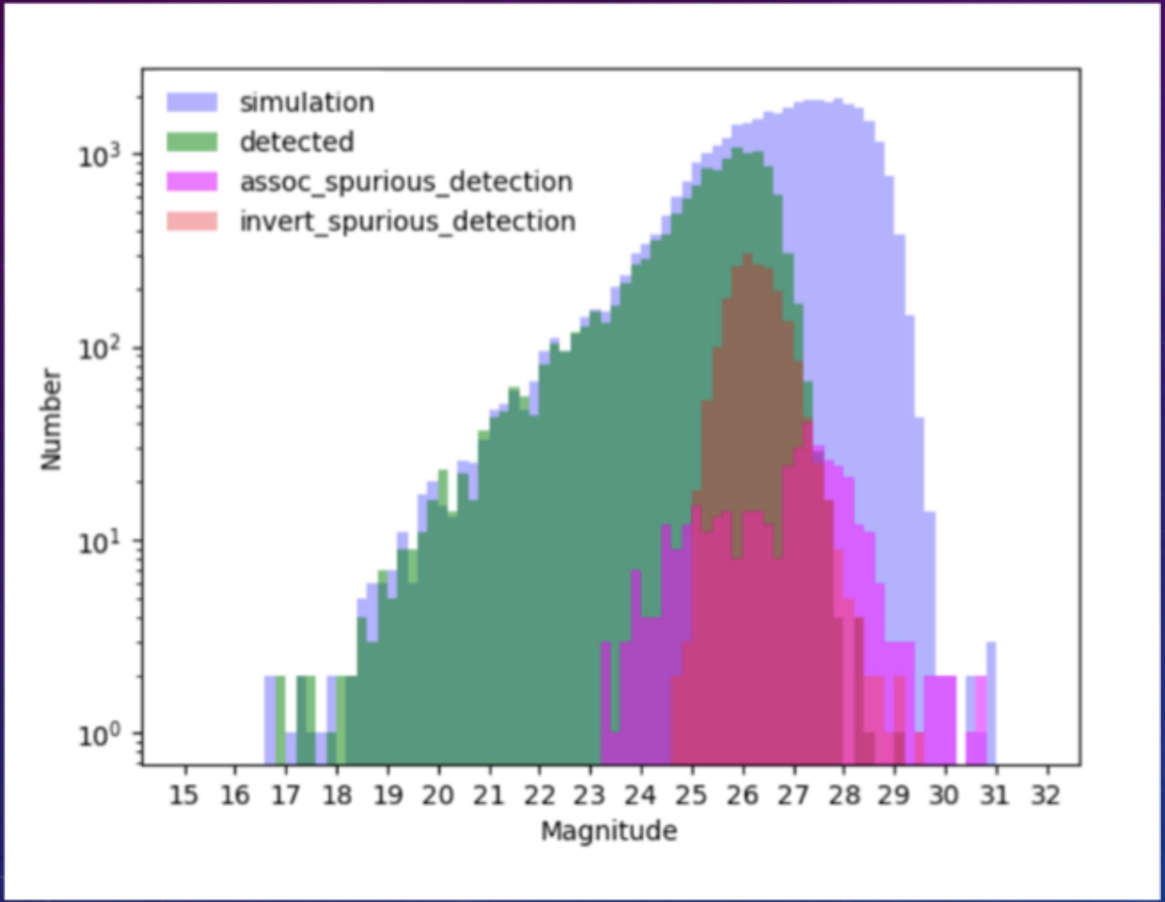


The quantity used in astronomy to measure fluxes is the magnitude.

Flux

COMPARISON METRIC

How do we compare the efficiency of these denoising algorithms?



Flux

The quantity used in astronomy to measure fluxes is the magnitude.

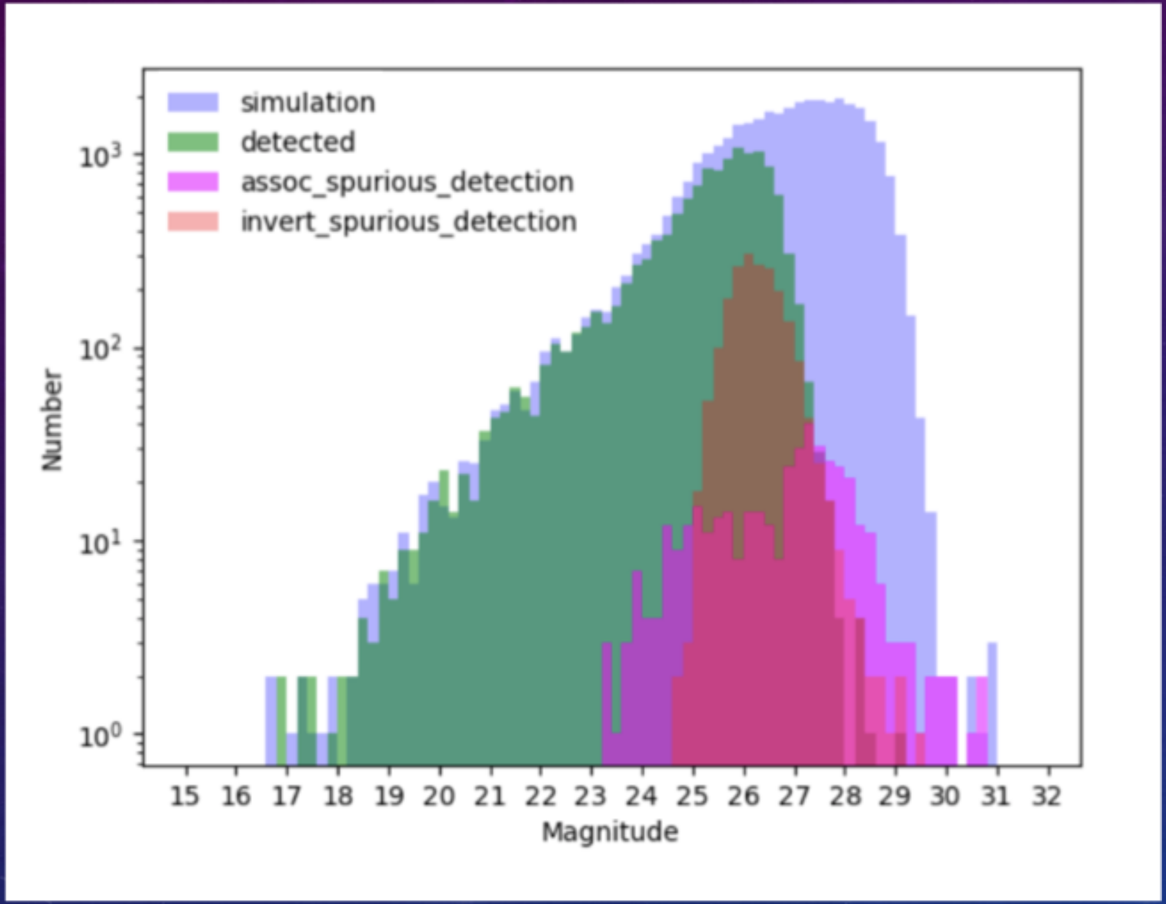
Defined as:

$$m_x = -2.5 \log_{10}\left(\frac{F_x}{F_{x,0}}\right)$$

Pogson 1856

COMPARISON METRIC

How do we compare the efficiency of these denoising algorithms?



Flux

The quantity used in astronomy to measure fluxes is the magnitude.

Defined as:

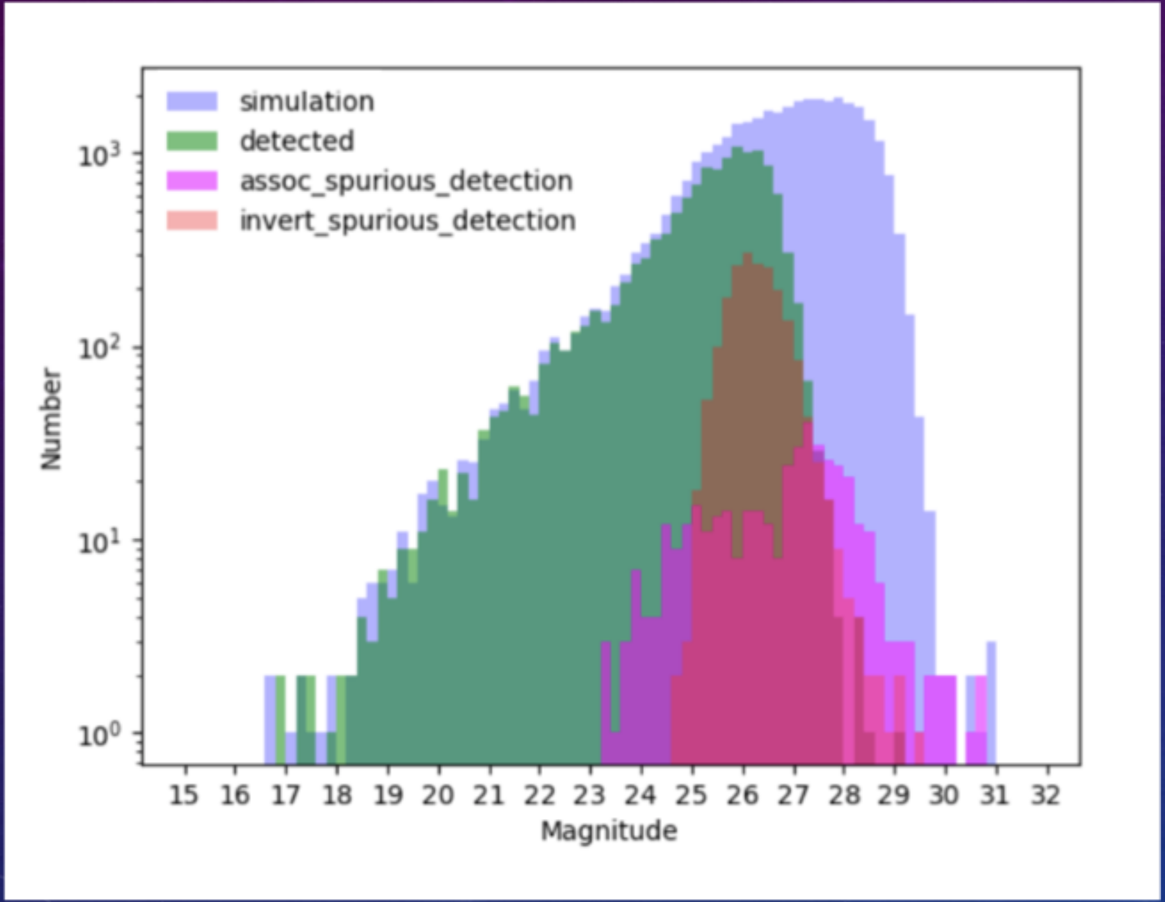
$$m_x = -2.5 \log_{10} \left(\frac{F_x}{F_{x,0}} \right)$$

Pogson 1856

I know..

COMPARISON METRIC

How do we compare the efficiency of these denoising algorithms?



Flux

The quantity used in astronomy to measure fluxes is the magnitude.

Defined as:

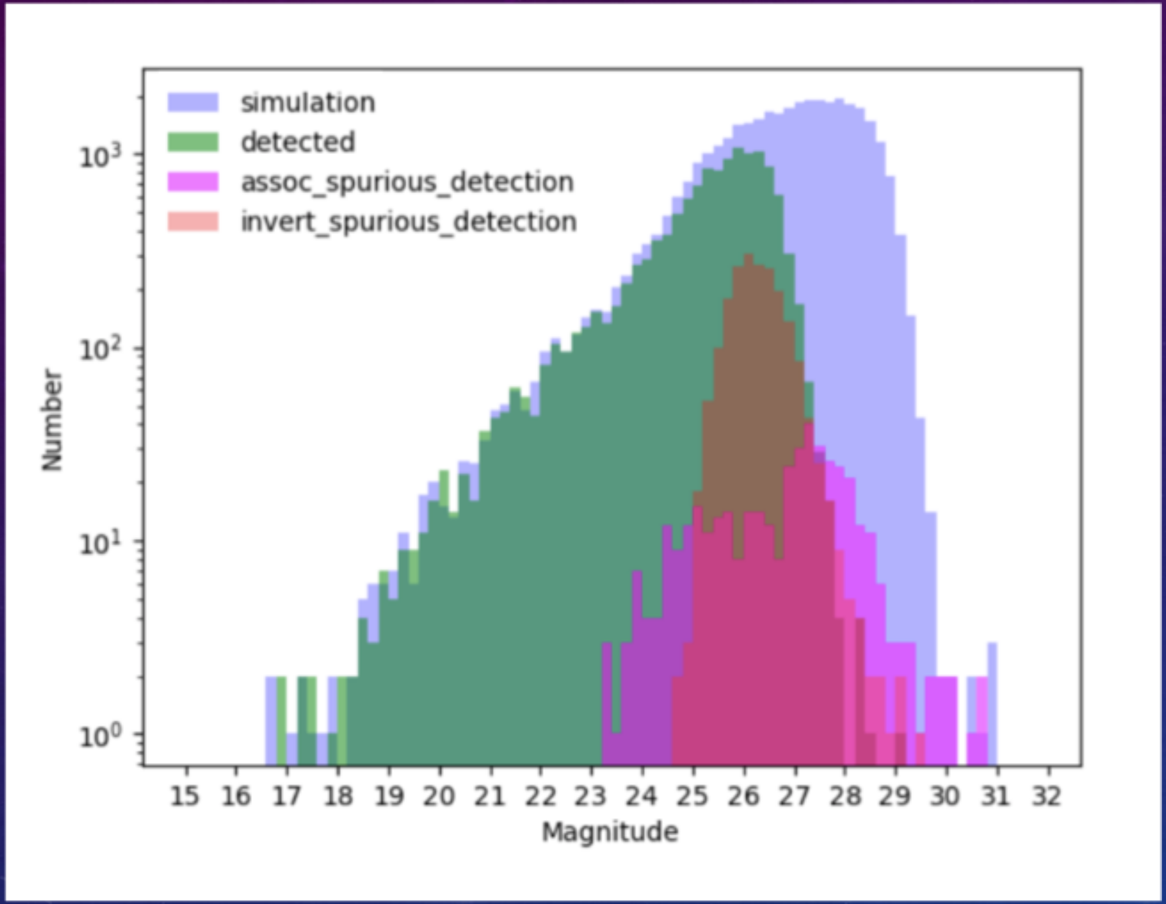
$$m_x = -2.5 \log_{10} \left(\frac{F_x}{F_{x,0}} \right)$$

Pogson 1856

..absolutely not intuitive, but..

COMPARISON METRIC

How do we compare the efficiency of these denoising algorithms?



Flux

The quantity used in astronomy to measure fluxes is the magnitude.

Defined as:

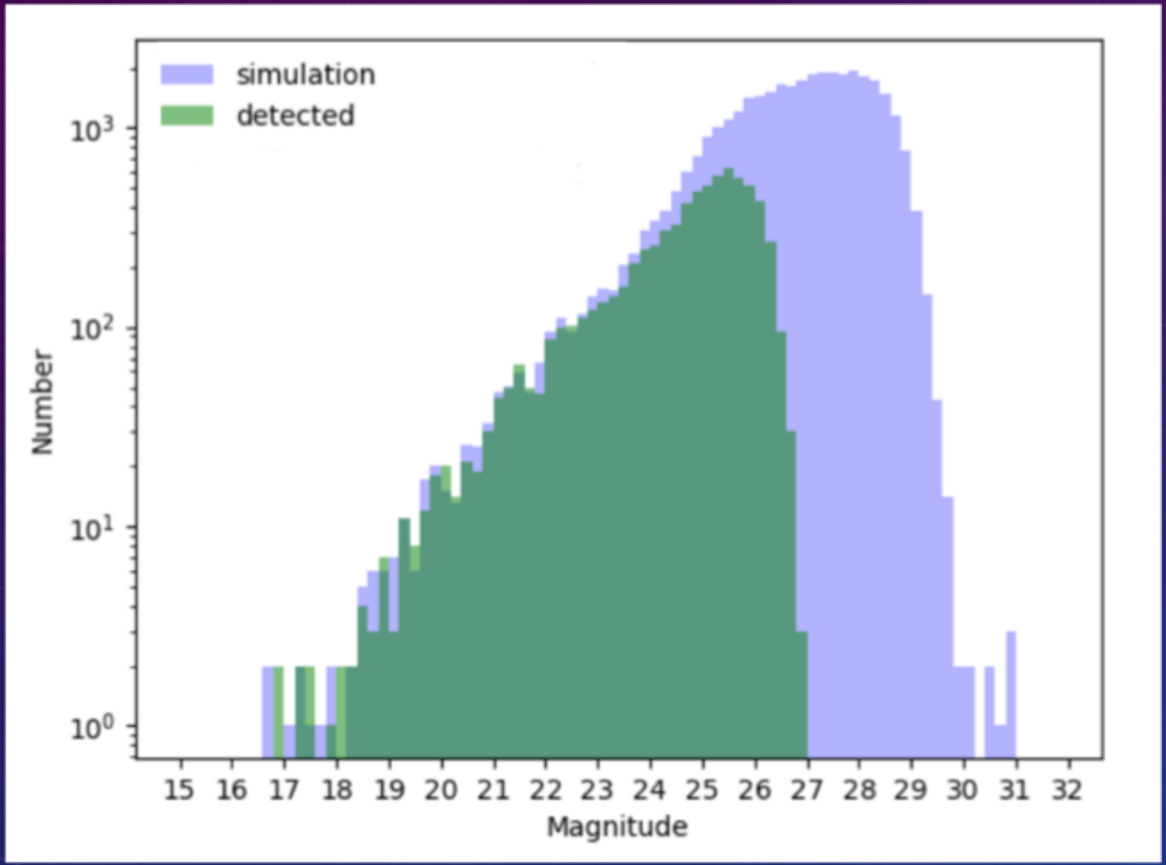
$$m_x = -2.5 \log_{10}\left(\frac{F_x}{F_{x,0}}\right)$$

Pogson 1856

..in Pogson's time a common misconception was that the logarithmic nature of the scale is because the human eye itself has a logarithmic response. (Now seems to be a power law)

COMPARISON METRIC

How do we compare the efficiency of these denoising algorithms?



The quantity used in astronomy to measure fluxes is the magnitude.

Defined as:

$$m_x = -2.5 \log_{10}\left(\frac{F_x}{F_{x,0}}\right)$$

Pogson 1856

..in Pogson's time a common misconception was that the logarithmic nature of the scale is because the human eye itself has a logarithmic response. (Now seems to be a power law)

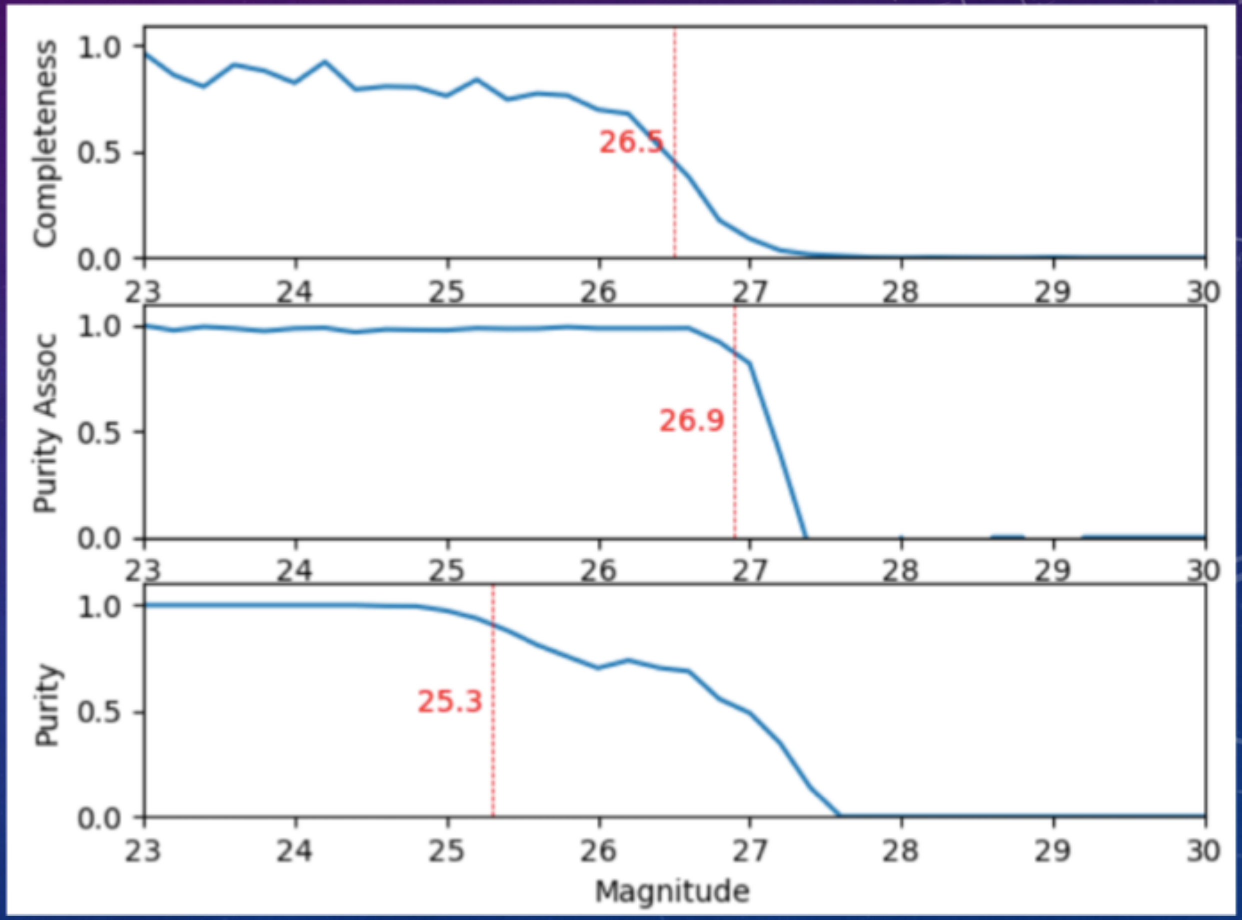
Flux

COMPARISON METRIC

How do we compare the efficiency of these denoising algorithms?
Let's now define two functions:

Completeness = $\frac{N_{detected} - N_{spurious}}{N_{real}}$

Purity = $\frac{N_{spurious}}{N_{detected}}$

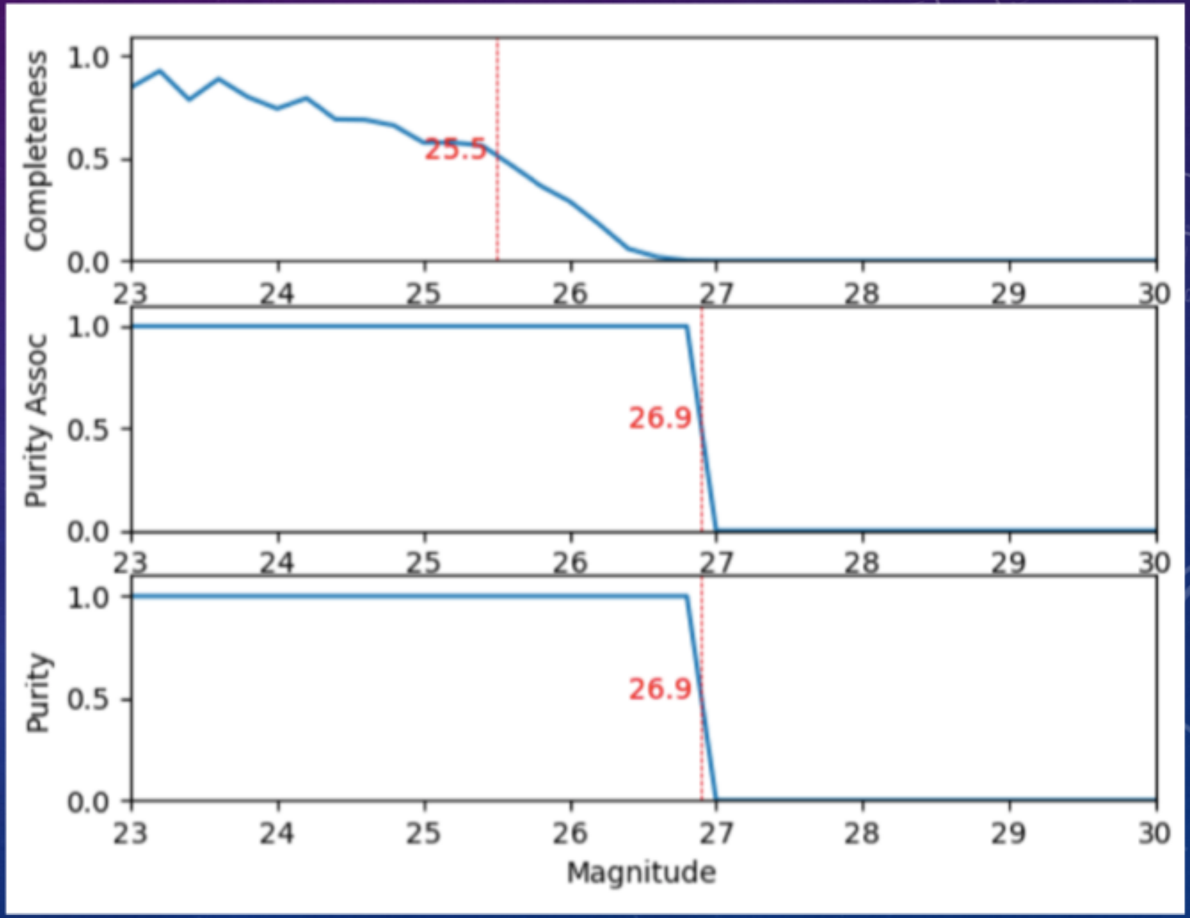


COMPARISON METRIC

How do we compare the efficiency of these denoising algorithms?
Let’s now define two functions:

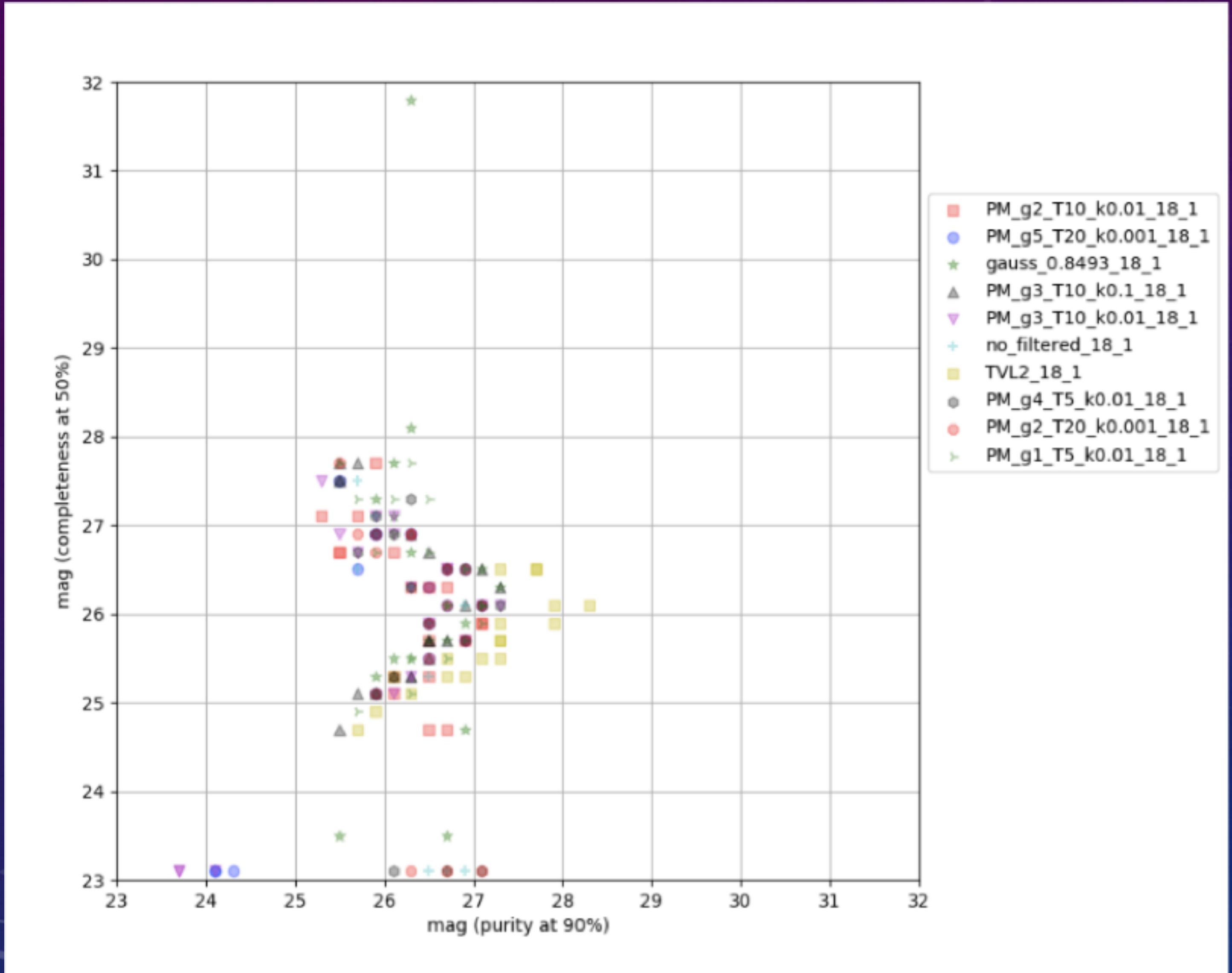
$$\text{Completeness} = \frac{N_{detected} - N_{spurious}}{N_{real}}$$

$$\text{Purity} = \frac{N_{spurious}}{N_{detected}}$$



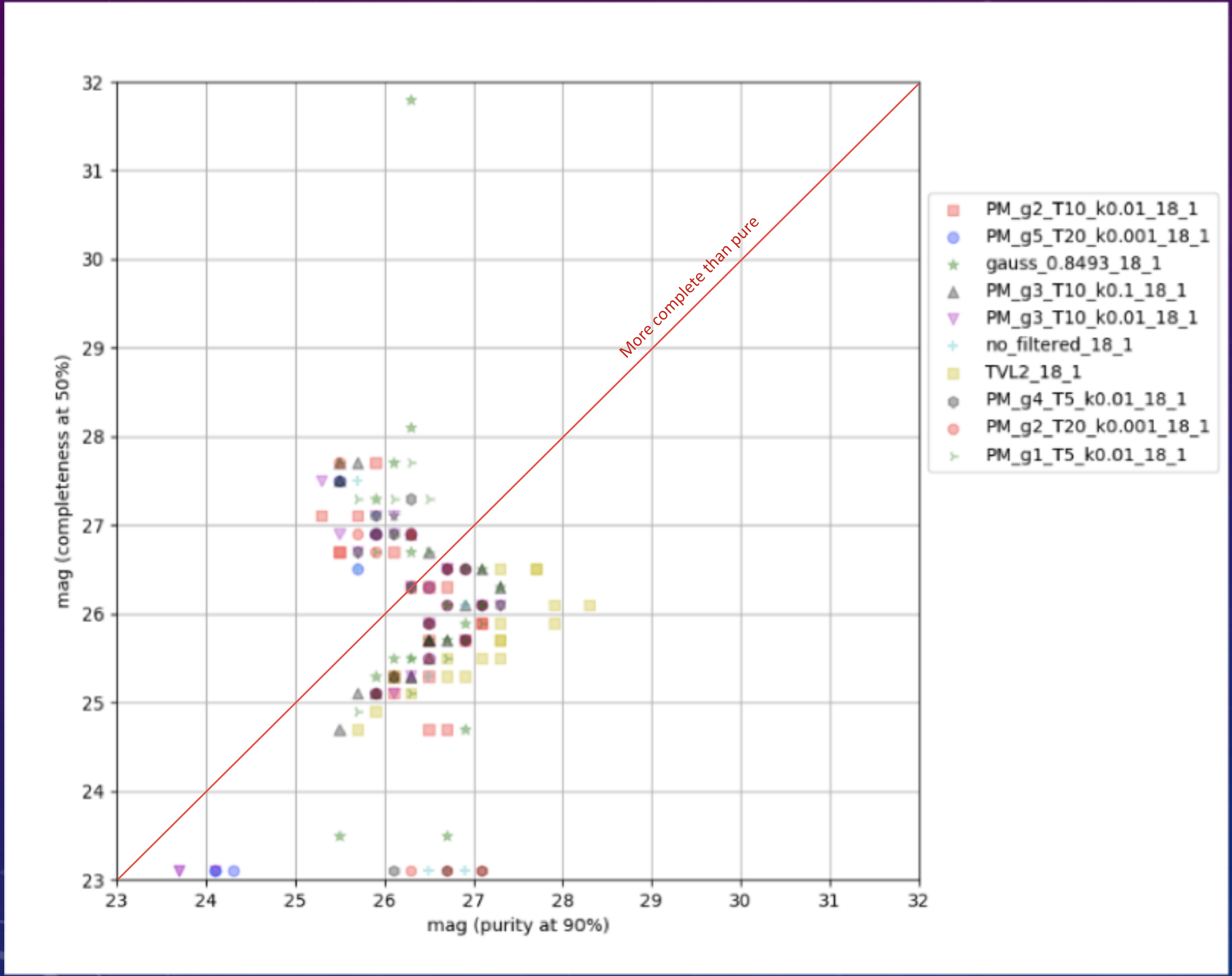
Flux

COMPARISON METRIC



1. Original image (no_filtered)
2. Optimal Gauss filtering (gauss 0.8493)
3. Perona-Malik with different parameters (PMs)
4. ATVD with L2 norm (TVL2)

COMPARISON METRIC

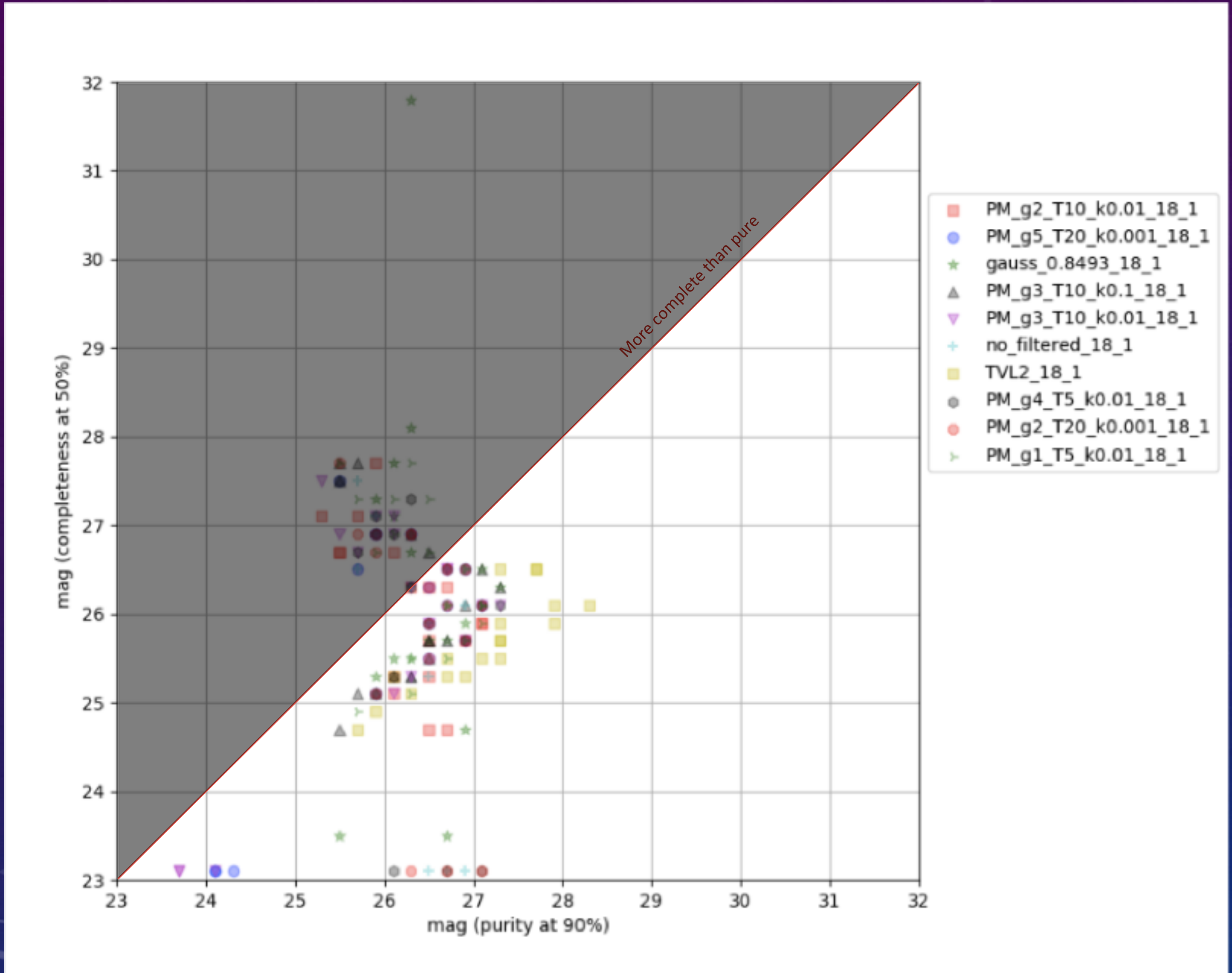


1. Original image (no_filtered)
2. Optimal Gauss filtering (gauss 0.8493)
3. Perona-Malik with different parameters (PMs)
4. ATVD with L2 norm (TVL2)

As they are defined, completeness and purity compel to chose a tradeoff

Catalogues which are more complete than pure are not interesting

COMPARISON METRIC



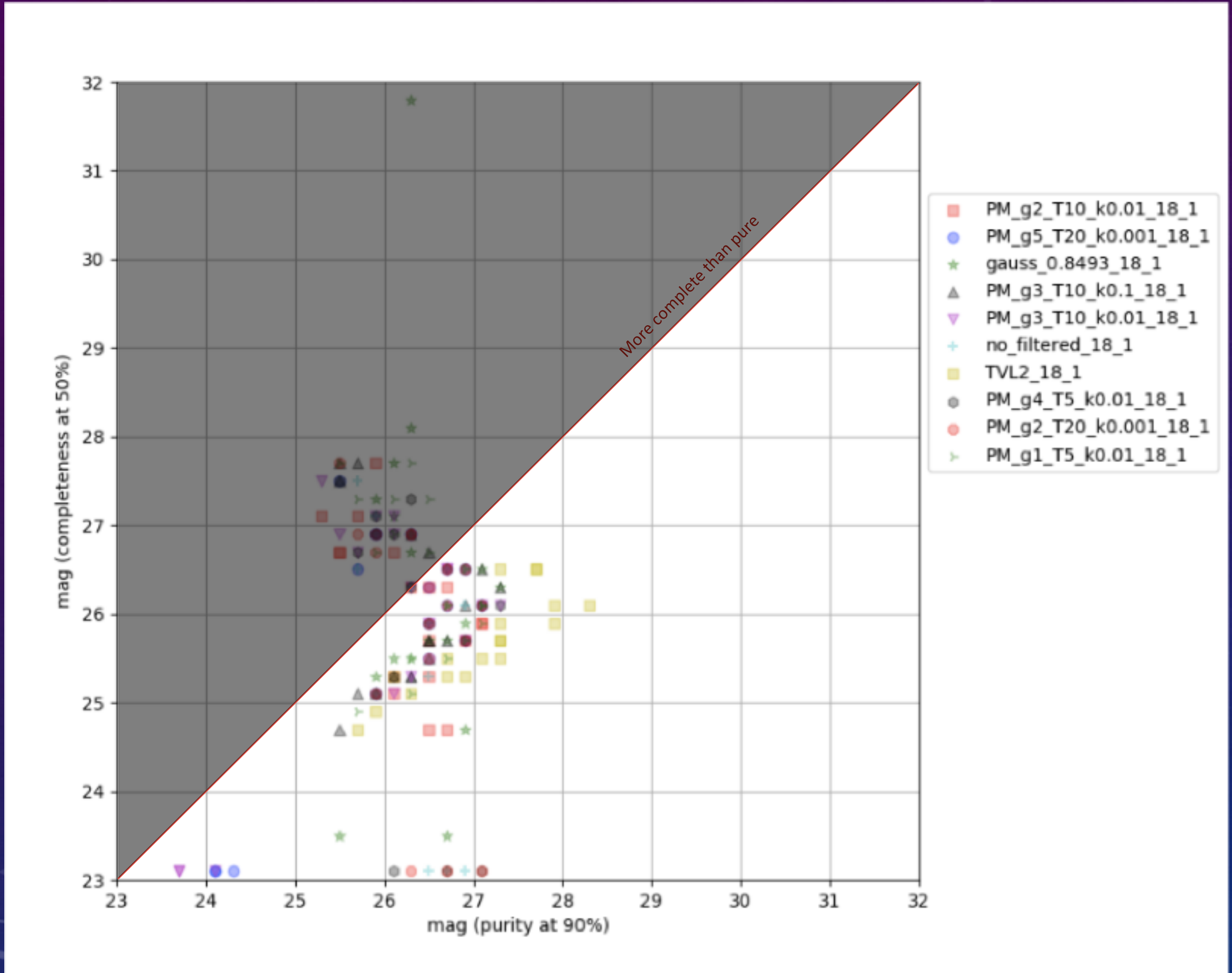
1. Original image (no_filtered)
2. Optimal Gauss filtering (gauss_0.8493)
3. Perona-Malik with different parameters (PMs)
4. ATVD with L2 norm (TVL2)

As they are defined, completeness and purity compel to chose a trade-off

Catalogues which are more complete than pure are not interesting

From the remaining we need to understand which one can better suits our needs

COMPARISON METRIC



1. Original image (no_filtered)
2. Optimal Gauss filtering (gauss_0.8493)
3. Perona-Malik with different parameters (PMs)
4. ATVD with L2 norm (TVL2)

As they are defined, completeness and purity compel to chose a trade-off

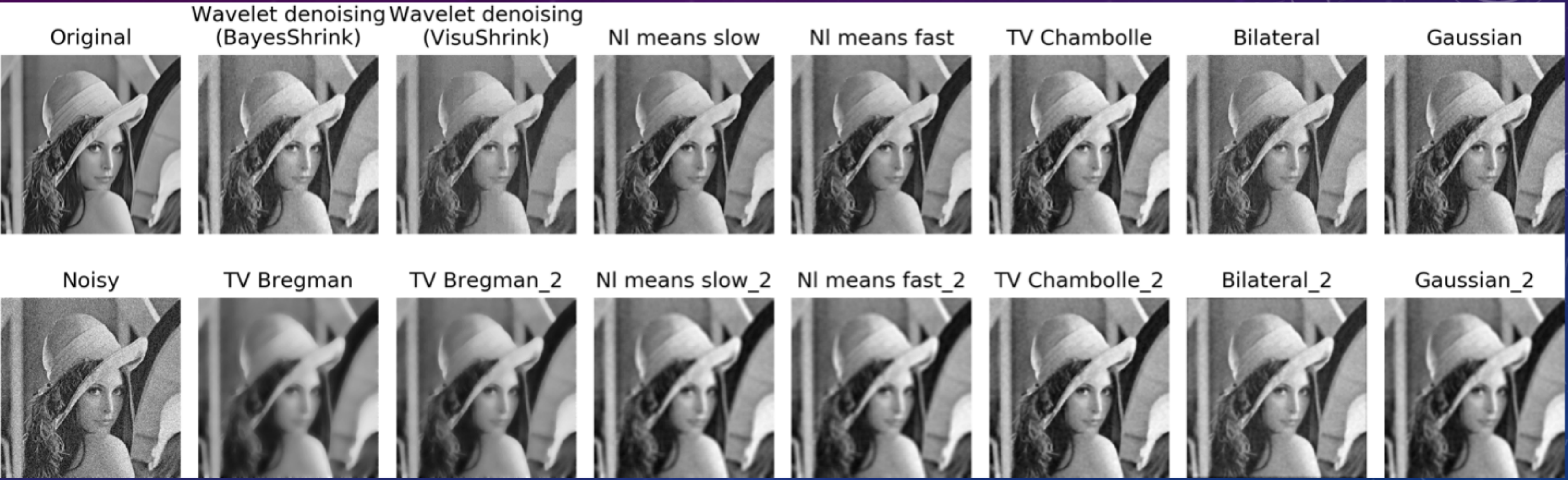
Catalogues which are more complete than pure are not interesting

From the remaining we need to understand which one can better suits our needs

ATVD seems to have the best performances!

FUTURE DEVELOPMENT

1. Testing and comparing other algorithms in order to have a more complete concept of denoising performances



FUTURE DEVELOPMENT

1. Testing and comparing other algorithms in order to have a more complete concept of denoising performances
2. Extend the testing methods, considering other important parameters from the literature:
   Peak-Signal-Noise-Ratio, Mean squared error, Structural similar index, Shannon entropy, etc.

Index	psnr	mse	ssim	entropy
original	inf	0	1	-9.05779
tv chambolle	29.3465	0.00116239	0.899884	-8.89291
gaussian	28.4342	0.00143409	0.857023	-8.96995
nl means fast	28.3812	0.00145171	0.8494	-9.01115
tv chambolle (more)	27.8747	0.00163129	0.885682	-8.77168
wavelet bayes	27.199	0.0019059	0.838671	-8.9369
nl means slow	26.863	0.00205919	0.773758	-9.08077
bilateral (more)	26.184	0.00240771	0.863263	-8.64535
wavelet visu	25.6125	0.00274634	0.743312	-8.85059
gaussian (more)	25.2295	0.00299954	0.837015	-8.62429
tv bregman (more)	25.0241	0.00314478	0.842216	-8.49623
nl means fast using $ \sigma$	24.3939	0.00363585	0.808501	-8.60324
nl means slow using $ \sigma$	24.3549	0.00366865	0.806727	-8.60162
tv bregman	23.4071	0.00456338	0.810777	-8.35067
bilateral	22.9335	0.00508916	0.568917	-9.49442
noisy	18.5439	0.0139833	0.346106	-32.3608

FUTURE DEVELOPMENT

1. Testing and comparing other algorithms in order to have a more complete concept of denoising performances
2. Extend the testing methods, considering other important parameters from the literature:
   Peak-Signal-Noise-Ratio, Mean squared error, Structural similar index, Shannon entropy, etc.
3. Paper Work in progress!

Thanks for the attention!

☺