Module ECUE
“Machine Learning for Image Processing”

Data Enhancement

Xavier Descombes
Morpheme team
INRIA/I3S/iBV

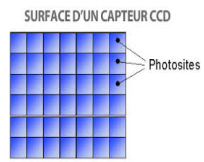
Introduction : why data enhancement ?

- Data (images in our case) can be subject to :
  - noise (additive gaussian, pepper and salt,…)
  - poor contrast (backlight, bad camera tuning,…)
  - light heterogeneity (optical effect, shadows,…)
  - Processing artefact
- Goal : improve image quality
  - For visualization
  - As a pre-processing step
- Machine learning
  - Number of approaches are based on features extracted from data (k-means, SVM, random forests,…)
  - Pre-processing improves the quality of features : reduce noise and variability

Data acquisition : noise sources

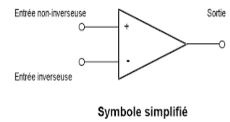
Total signal = useful signal + noise

photons

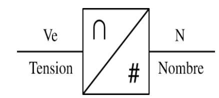


CCD sensor

Electric signal



Amplifier



Analog digital converter

Digital file

The camera detector collects the photons from the light and transforms them in electrical signal. This signal is amplified before being converted into numerical data (integers). Each step in this pipeline can be a source of noise.

Photon shot noise → Poisson distributed random noise

Fixed pattern noise

Lecture noise → Gaussian shaped random noise

Amplifier noise → Multiplicative Gaussian noise

Lecture noise and Amplifier noise: These noises increase with temperature > Thermic noise

Quantification noise → Gaussian shaped random noise

SNR = 10 x log (pT)
SNR : Signal to Noise Ratio
p = photon density
T = exposure time

Examples of noise

Original image

Additive Gaussian

Salt & Pepper

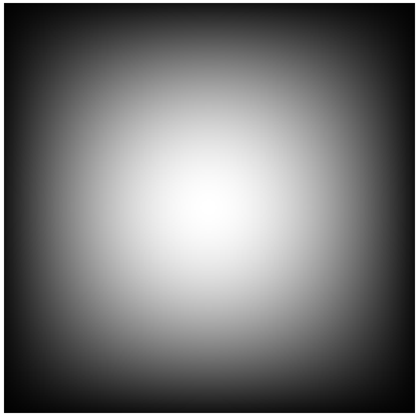
Speckle (multiplicative)

Poisson

Examples of light perturbation

Low contrast

Original image



Transfer function

Windowed image

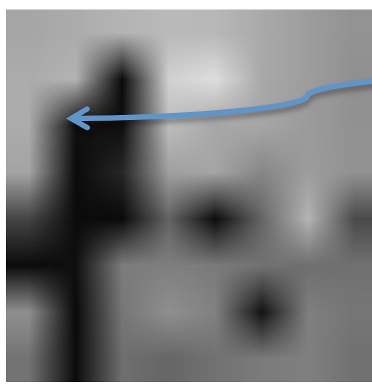
Three main approaches

- Work on the pixels themself (in the spatial domain)
- Work in a transformed domain (Fourier transform for example)
- Work on the histogram

Image: a discrete table





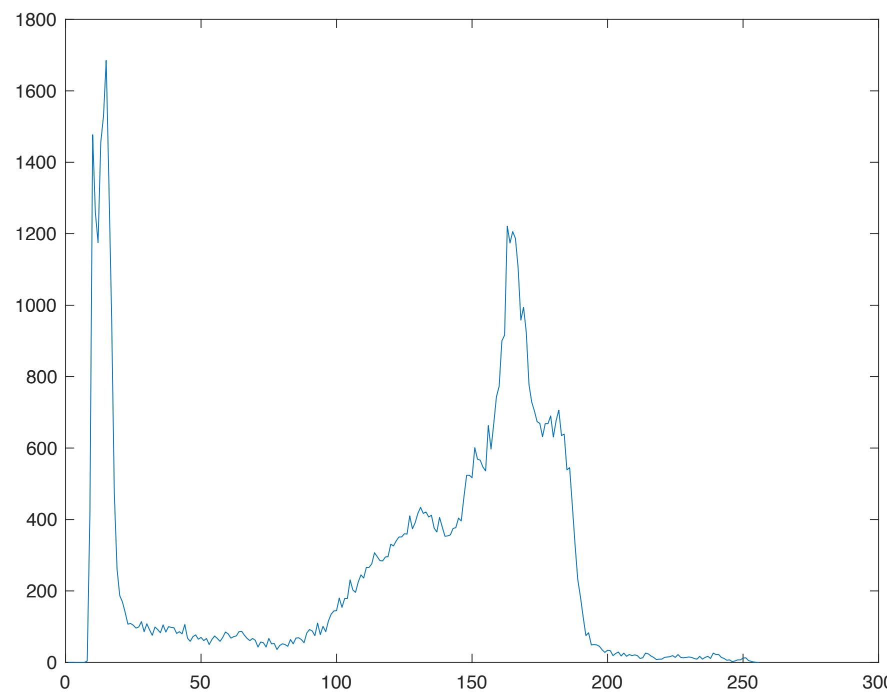


166	172	180	186	184	170	157	147
172	180	11	209	223	169	158	144
172	17	15	191	168	167	155	146
164	11	39	160	166	135	158	146
69	12	10	100	12	102	181	74
11	16	125	128	123	115	111	113
143	10	119	144	130	19	128	118
115	12	111	102	114	123	128	113

Histogram

$$X \in <1, NG>, h(x) = \frac{\#\{s \in I : i_s = x\}}{\#\{s \in I\}}$$

can be interprated as a probability



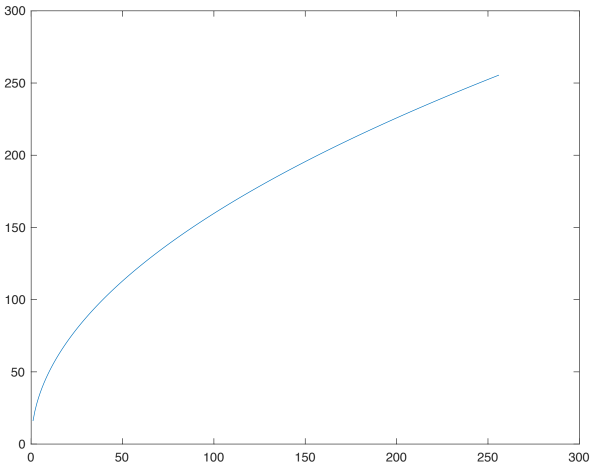
Histogram transform

$$T :< 1, NG > \to < 1, NG >$$
$$x \mapsto T(x)$$

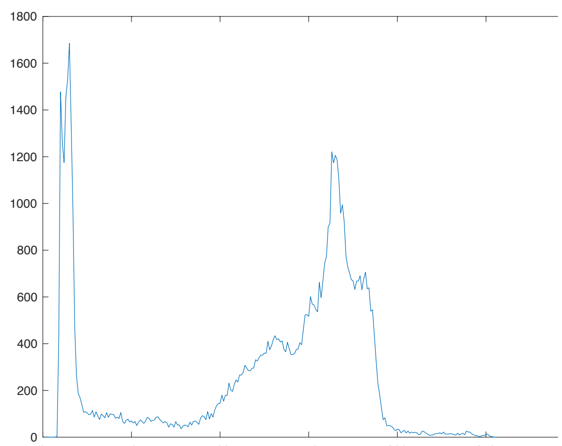
Algorithm :

- Input : image I
- output : image J
- for each pixel s in I :
  read x = I(s)
  set y(s) = T(x)

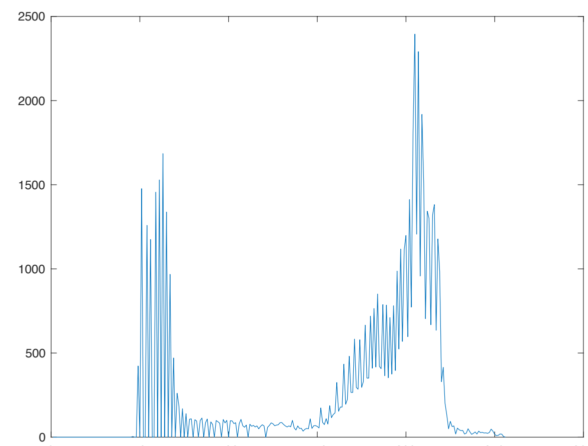
Transform 1



300
250
200
150
100
50
0
0
50
100
150
200
250
300

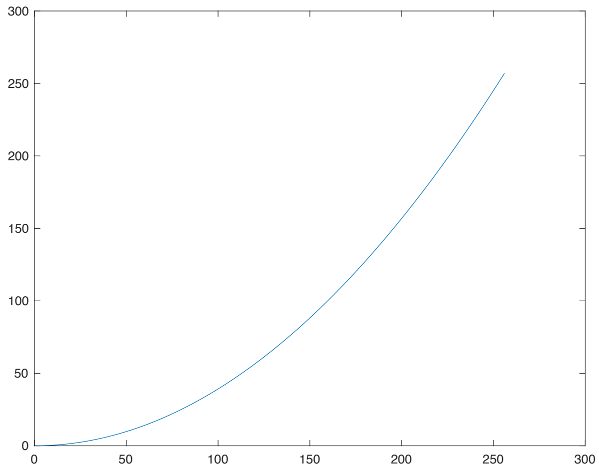


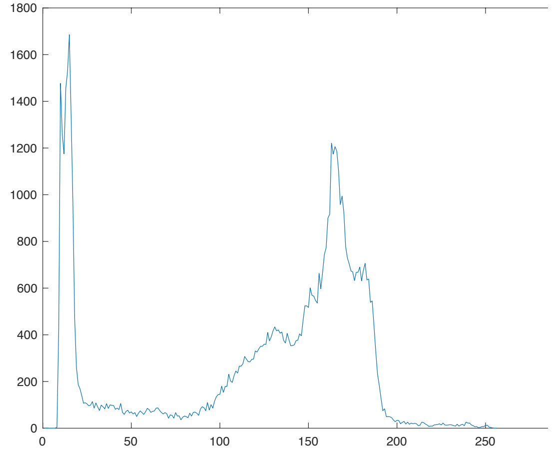
1800
1600
1400
1200
1000
800
600
400
200
0
0
50
100
150
200
250

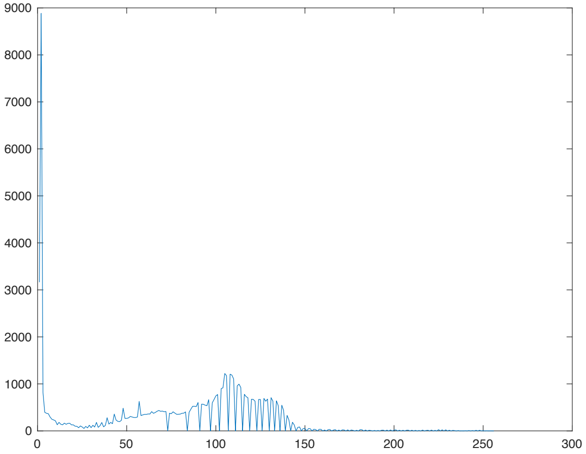


2500
2000
1500
1000
500
0
0
50
100
150
200
250
300

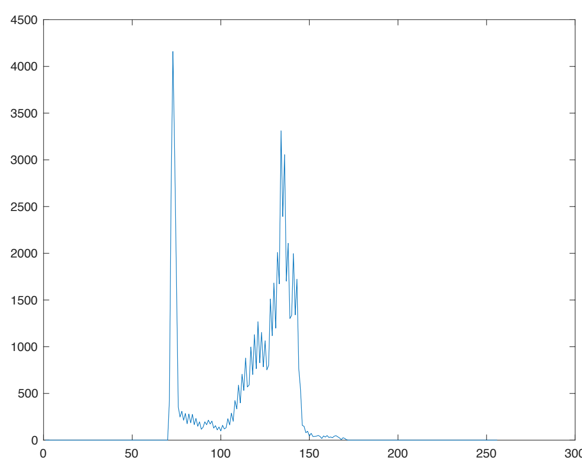
Transform 2

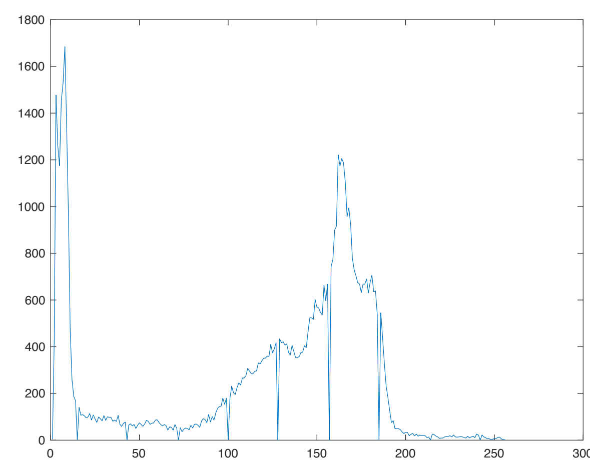

300
250
200
150
100
50
0
0
50
100
150
200
250
300


1800
1600
1400
1200
1000
800
600
400
200
0
0
50
100
150
200
250

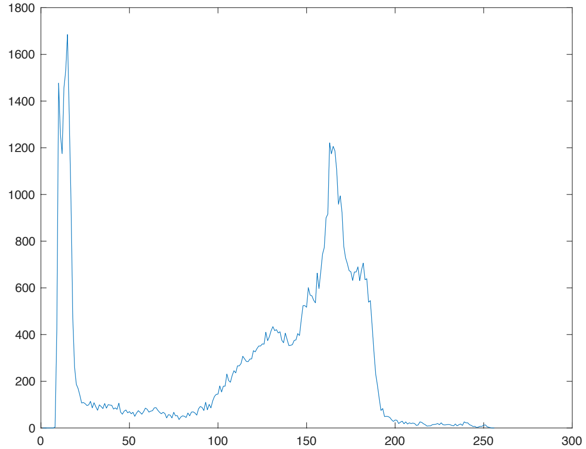

9000
8000
7000
6000
5000
4000
3000
2000
1000
0
0
50
100
150
200
250
300

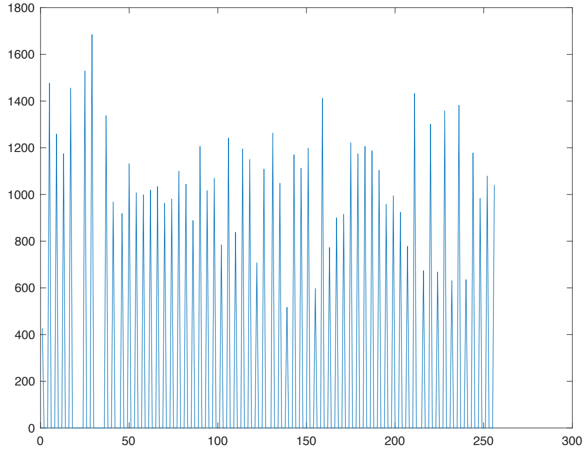
Histogram extension





Histogram equalization

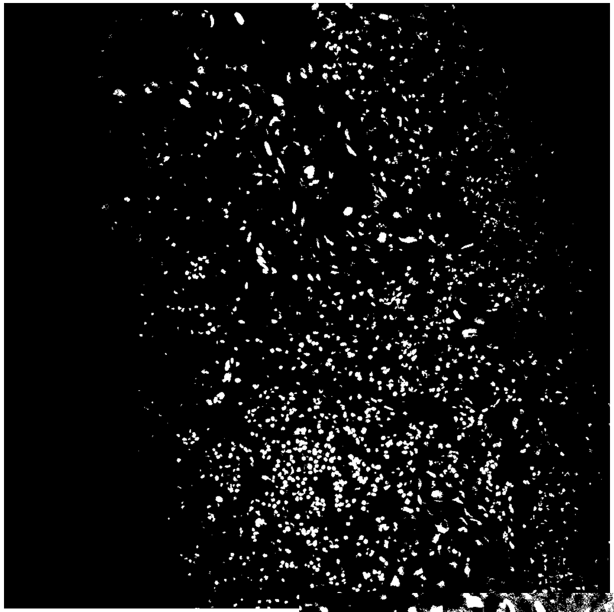


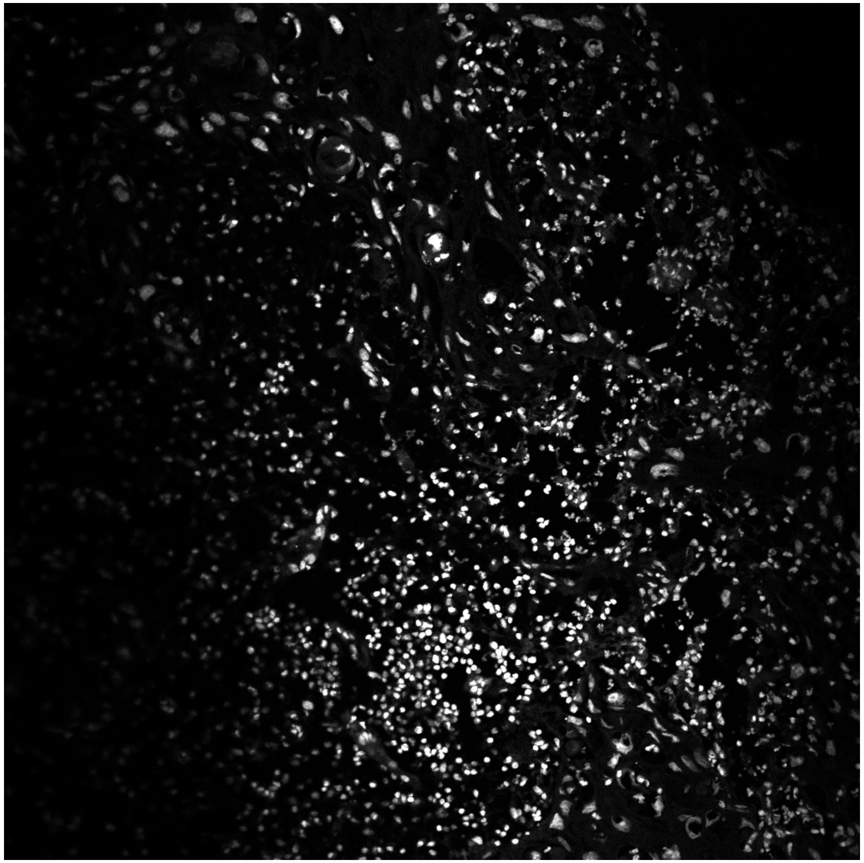


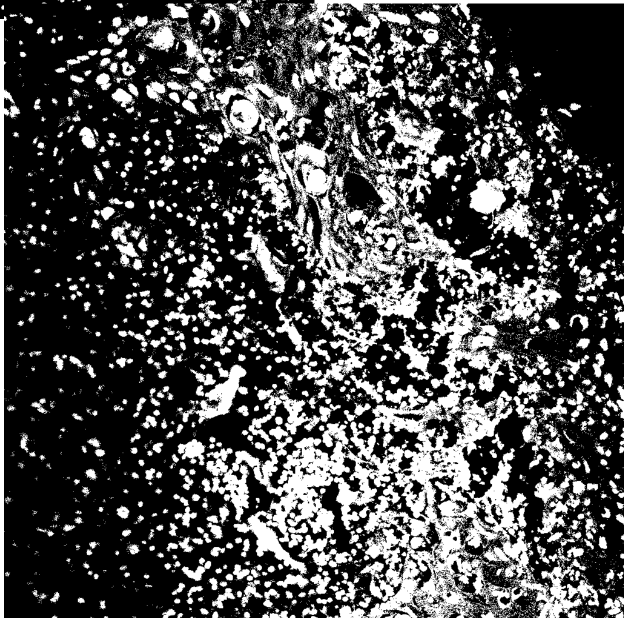
Local normalization

The intensity may be heterogeneous :

how to threshold ?





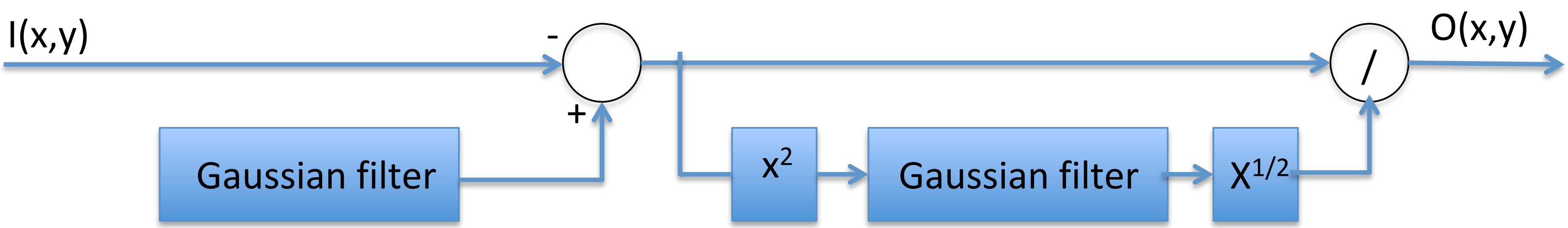


Local Normalization

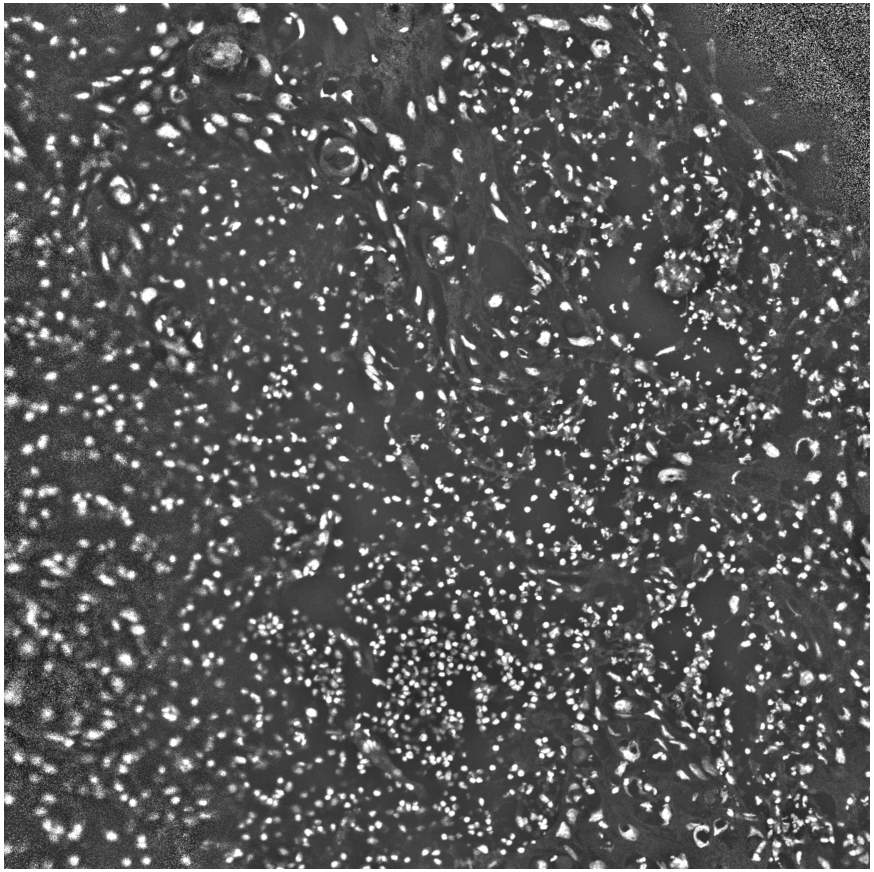
Goal : same mean and same variance everywhere

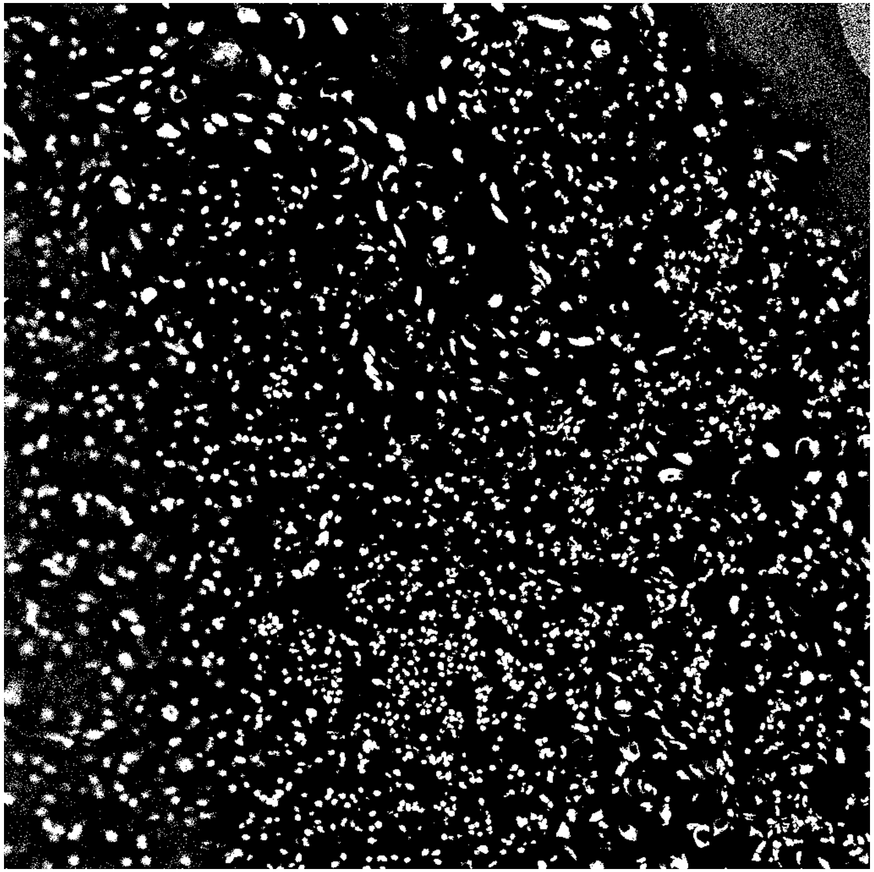
$$O(x,y) = \frac{I(x,y) - \mu_I(x,y)}{\sigma_I(x,y)}$$

In practice :

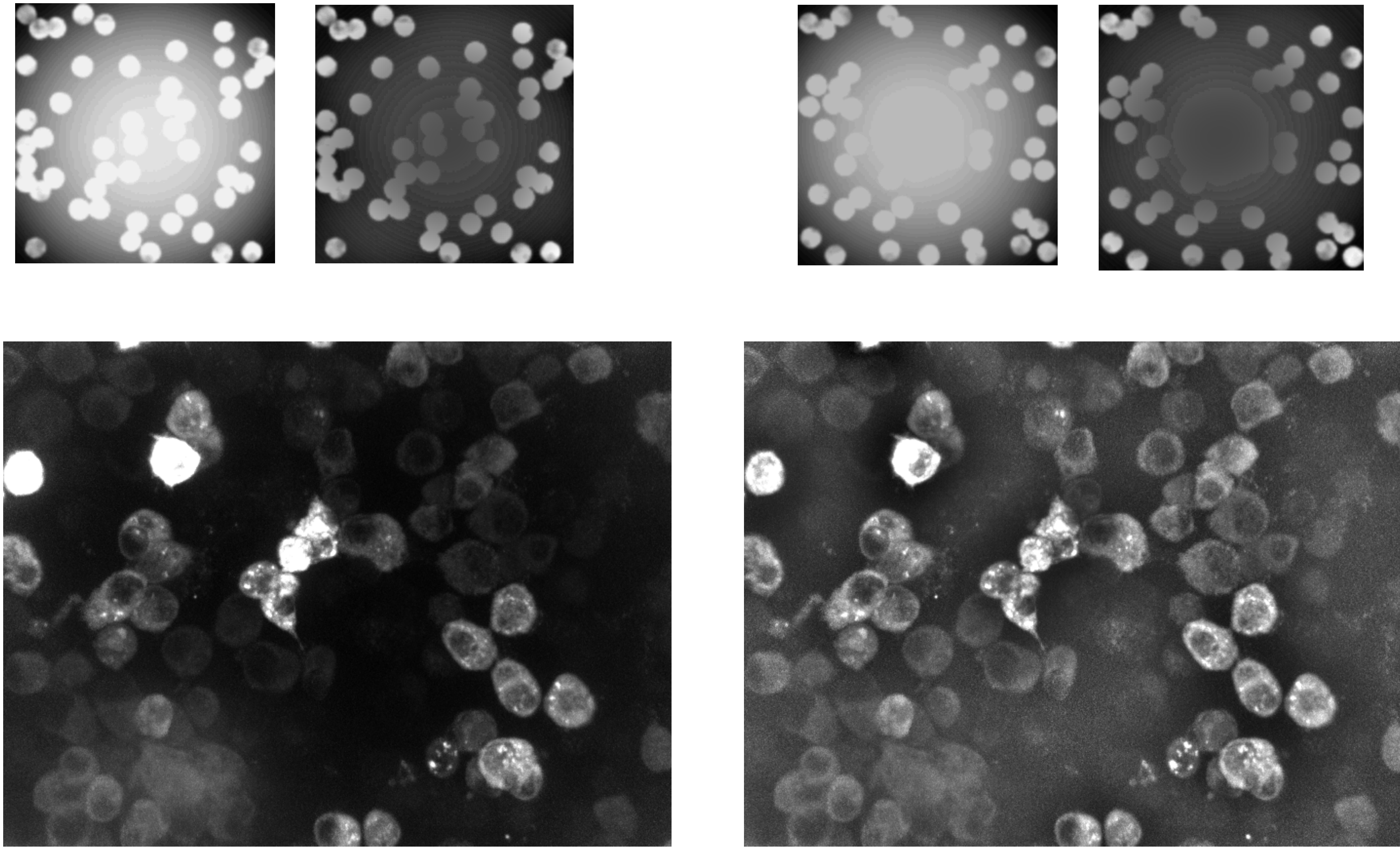


Local normalization





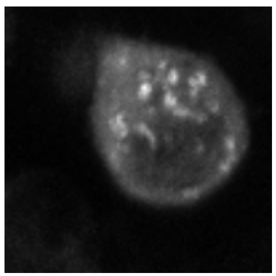
Spatial normalisation

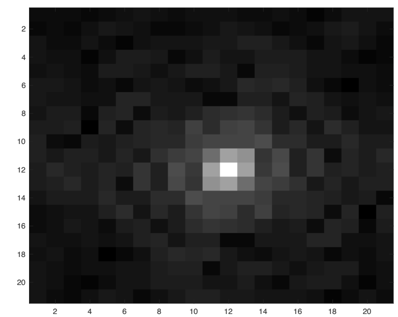


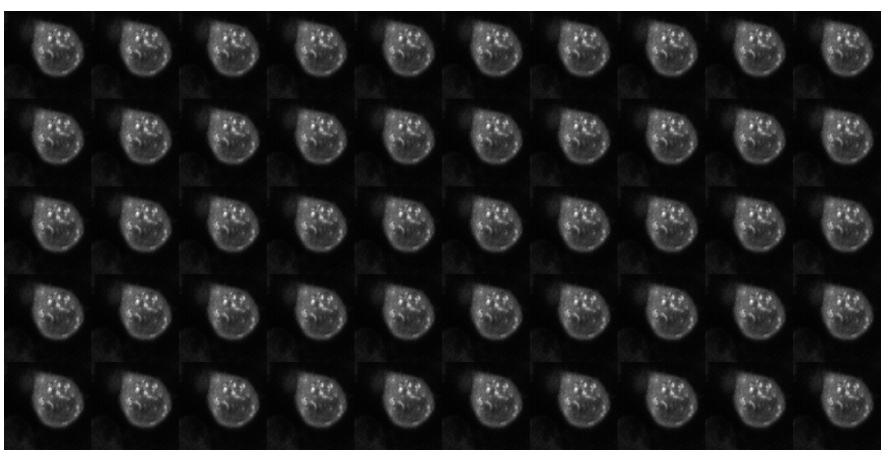
Fourier transform

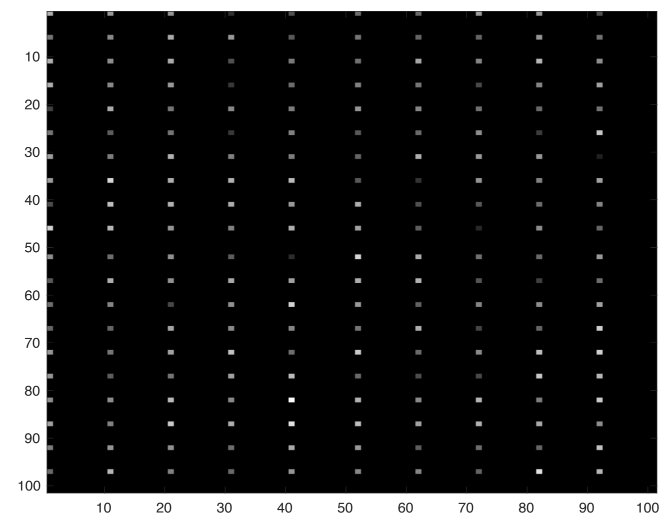
$$fftI(p,q) = \sum_{j=0}^{m-1} \sum_{k=0}^{n-1} I(j,k) e^{-2\pi i \frac{jp}{m}} e^{-2\pi i \frac{kq}{n}}$$

Frequency analysis

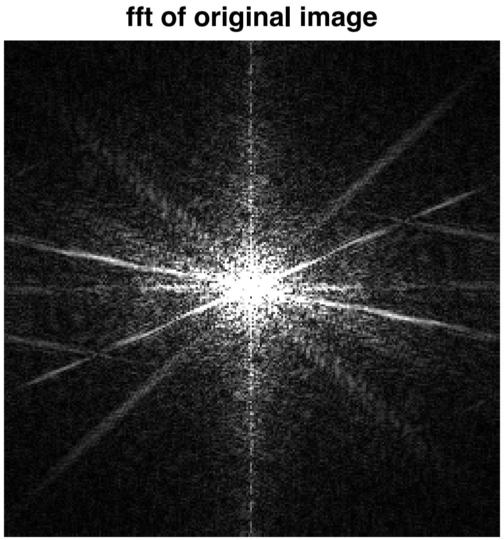


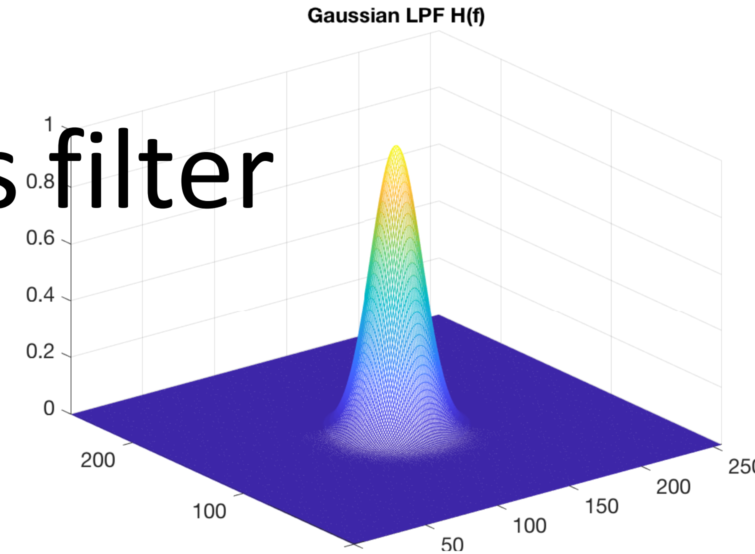




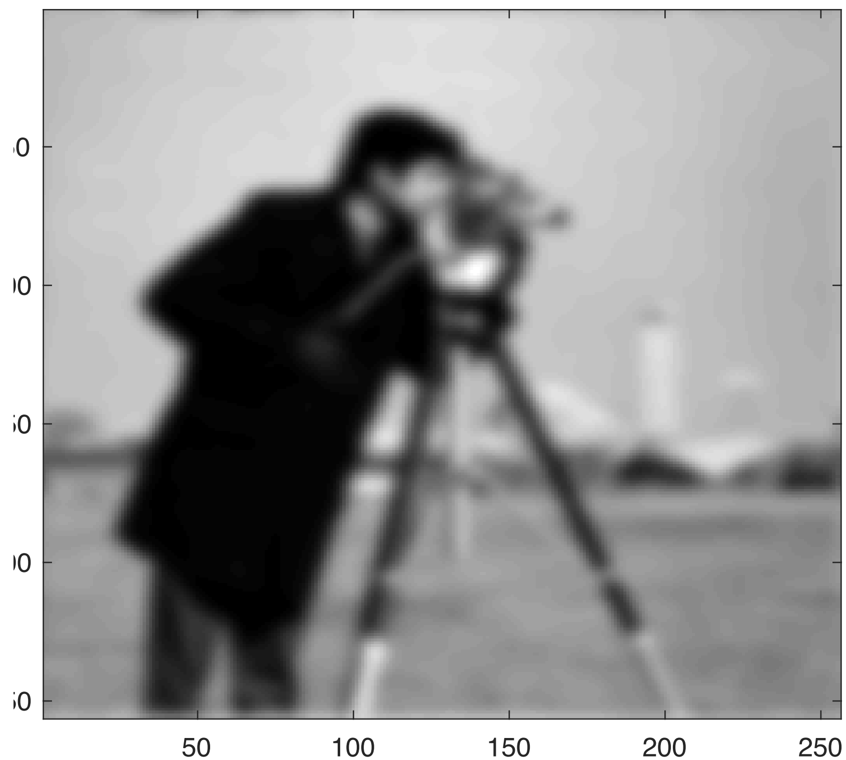


Lowpass filter

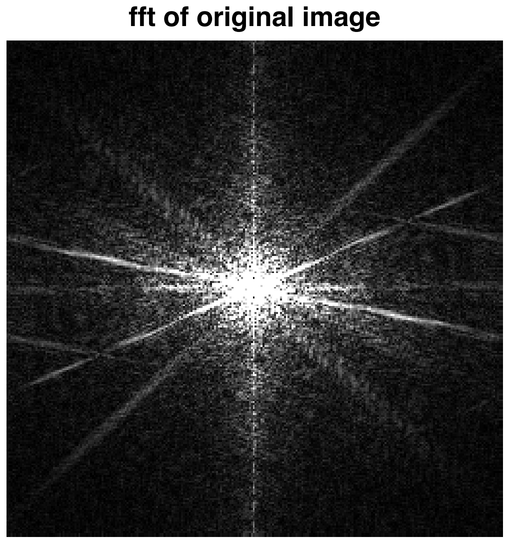


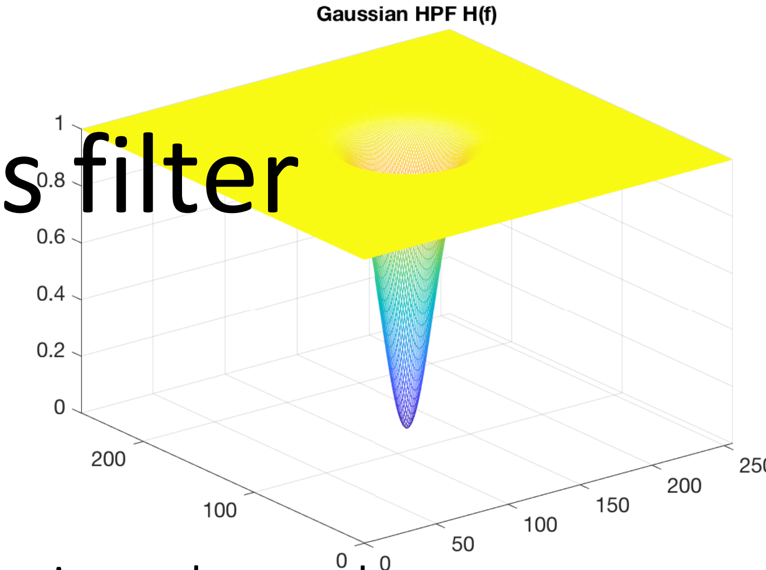


Cut high frequencies : remove noise (but also edges)



Highpass filter





Cut low frequencies : enhance edges

Spatial filtering : linear

Convolution with a filter : remove noise (example : Gaussian filter)

Lowpass filter : can destroy edges

Adapted to additive noise

$$J = G * I$$

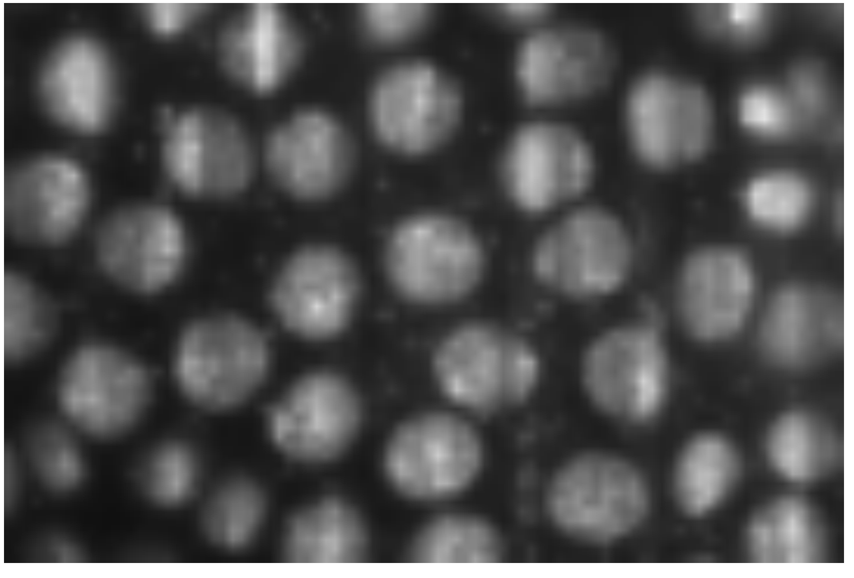
Spatial filtering : non linear

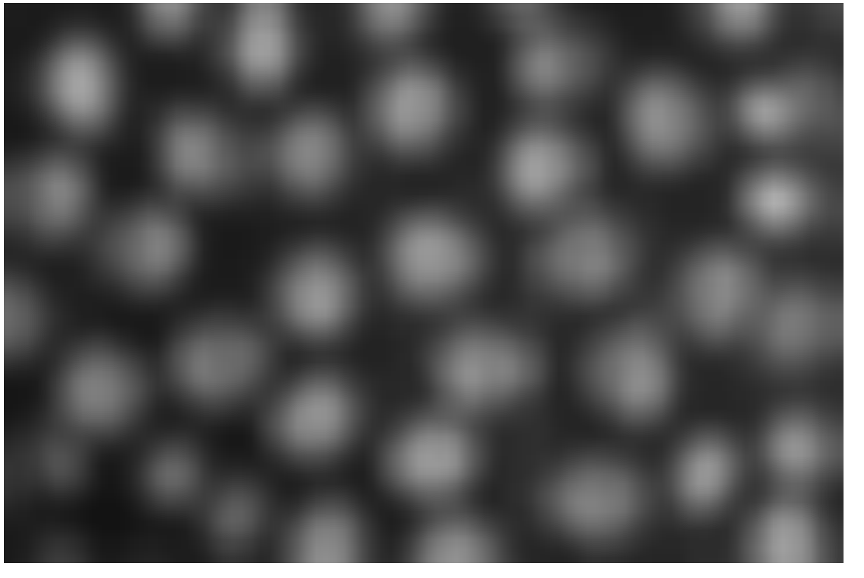
Mostly median filter
Preserve edges
Adapted to salt and pepper noise

Denoising

- Gaussian Filter : $O = G * I$

$$G(x, y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp - \left[\frac{x^2 + y^2}{2\sigma^2} \right]$$



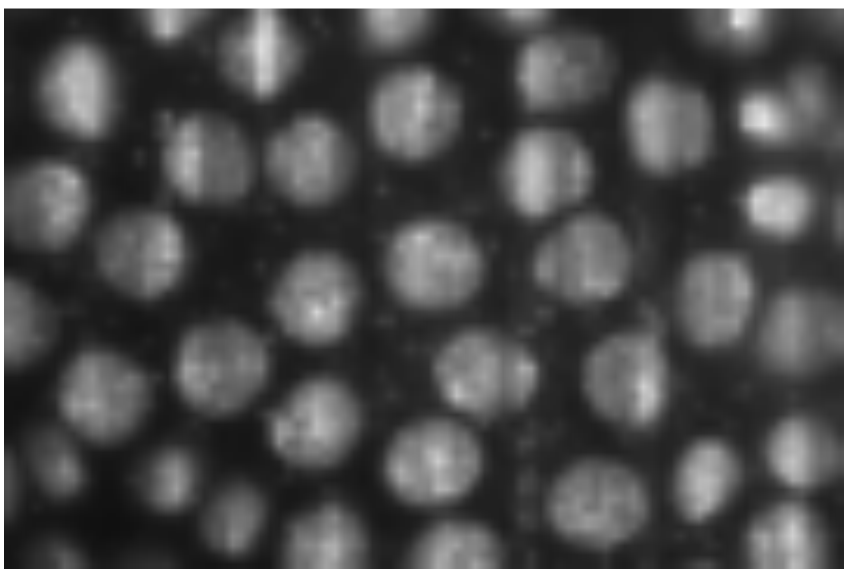


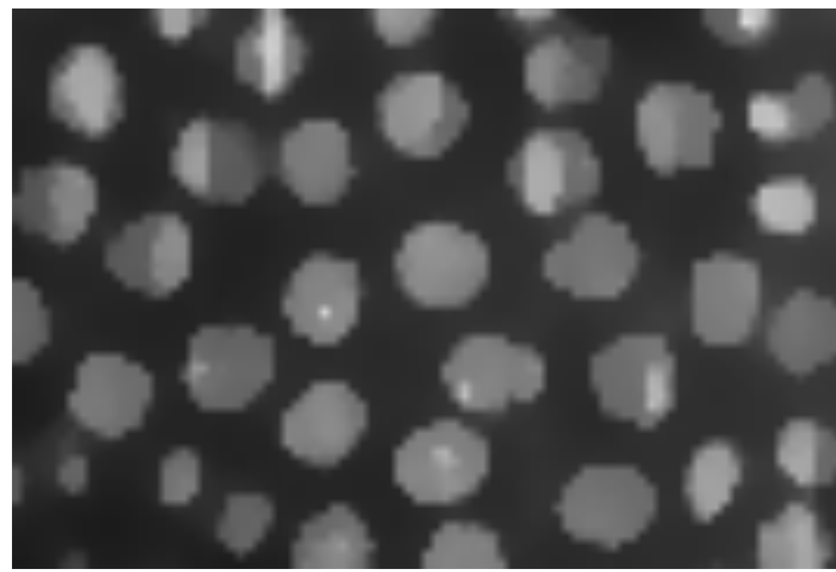
-> Remove noise but blur objects

Denoising

- Restoration : $\operatorname{argmin}\left(||O-I||_{L^n}+\Phi(O)\right)$

Example: $$U=\sum_{s}\frac{(o_s-i_s)^2}{2\sigma^2}-\beta\sum_{s\sim s'}\frac{1}{1+\frac{(o_s-o_{s'})^2}{\delta^2}}$$





-> Higher algorithmic complexity, embed prior information on the solution (cf Bayesian approach).

Linear structures enhancement

- Before detection (thresholding)
- Numerous applications : road, vessels, axons,filaments,….
- Define locally a linear structure : homogeneous intensity in the object direction, heterogeneity in the orthogonal direction

Gabor Filter

Sinusoidal shape weigthed by a Gaussian

Parameters :

- Frequency : λ
- Orientation : θ
- Phase : ϕ
- Gaussian variance : σ^2

$$G_{(x,y)}(u,v|\lambda,\theta,\phi,\sigma) = \exp\left[-\frac{(u-x)^2+(v-y)^2}{2\sigma^2}\right]\sin(\lambda(u\cos\theta - v\sin\theta)+\phi)$$

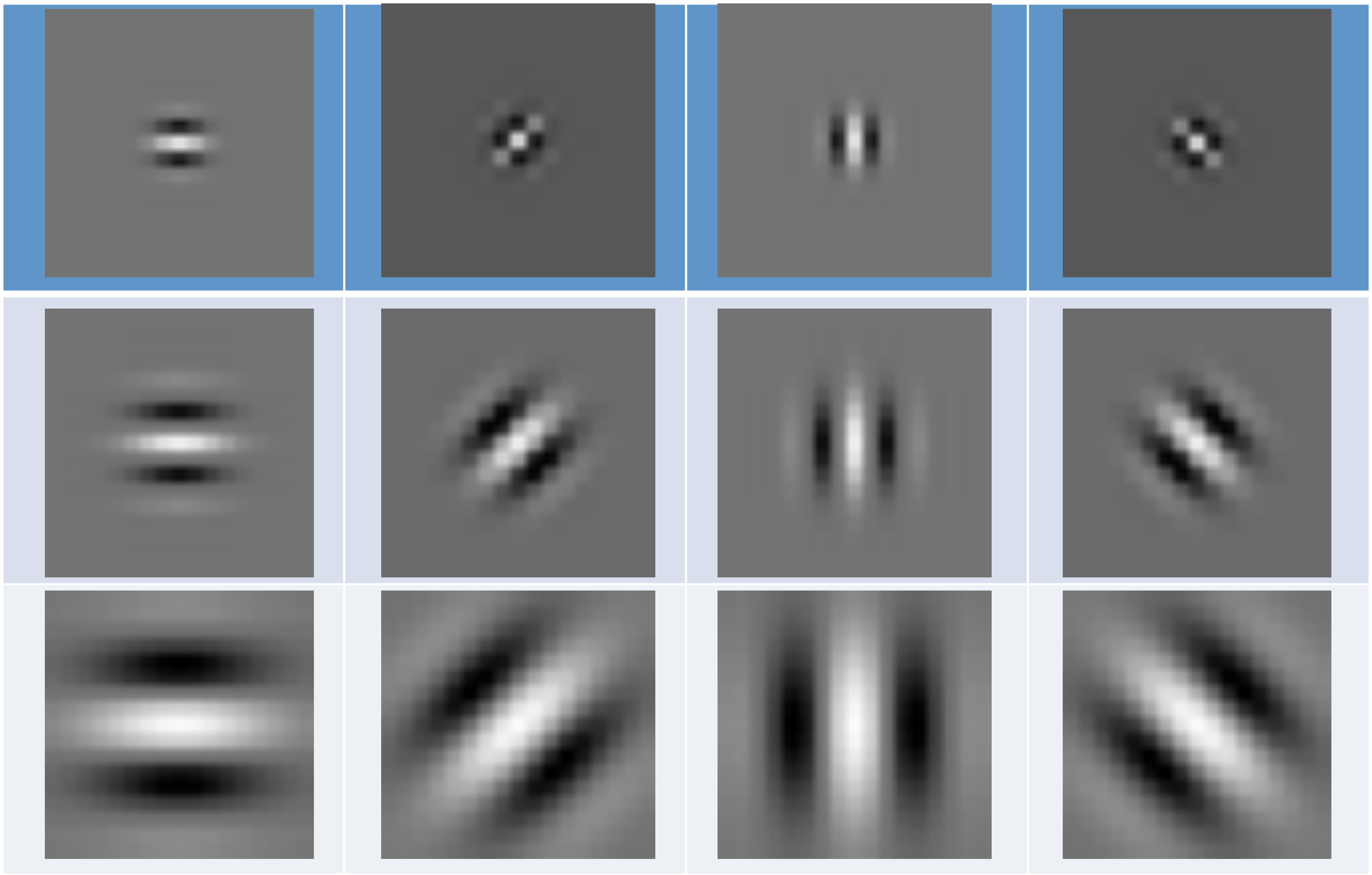
Energy :

$$E(x,y|\lambda,\theta,\sigma) = \left(\sum_{u,v} G_{(x,y)}(u,v|\lambda,\theta,0,\sigma)I(u,v)\right)^2 + \left(\sum_{u,v} G_{(x,y)}(u,v|\lambda,\theta,\frac{\pi}{2},\sigma)I(u,v)\right)^2$$

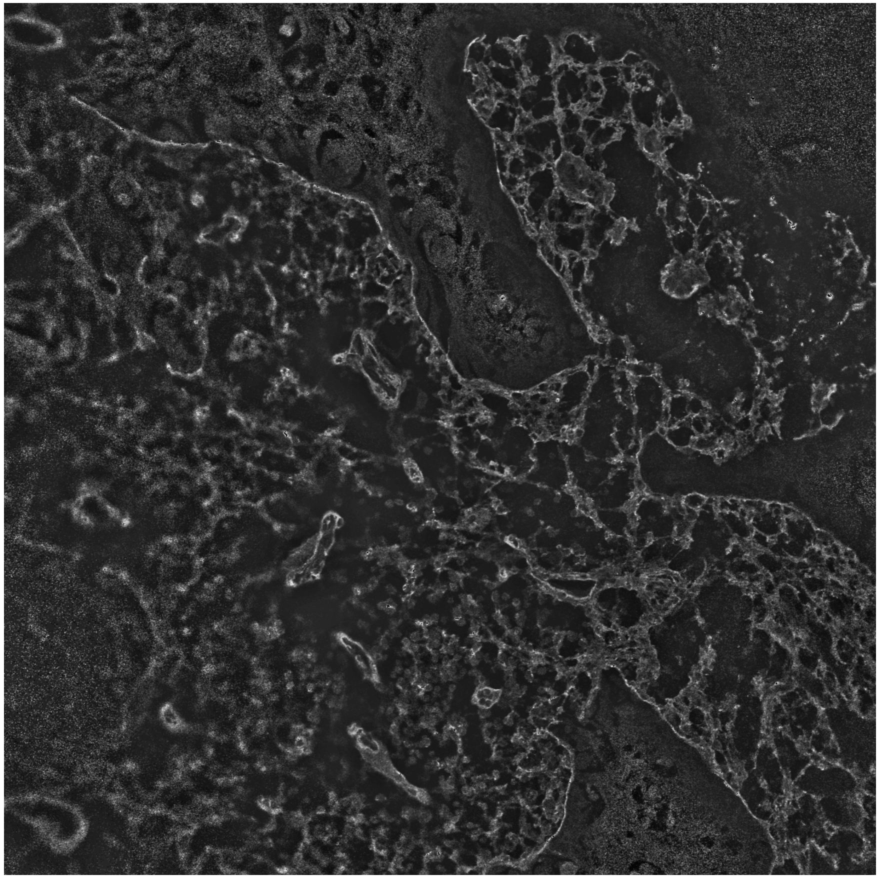
In practice :

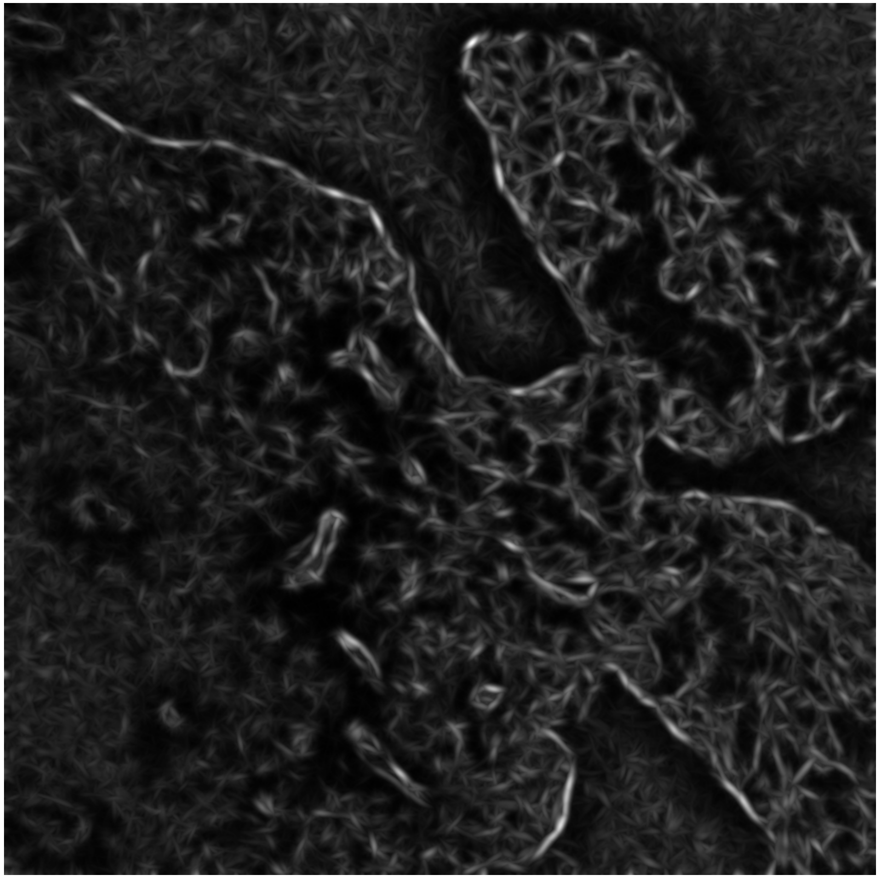
$$F(x,y|\lambda,\theta,\sigma) = \max_{\phi}\left(\sum_{u,v} G_{(x,y)}(u,v|\lambda,\theta,\phi,\sigma)I(u,v)\right)^2$$

Gabor Filters



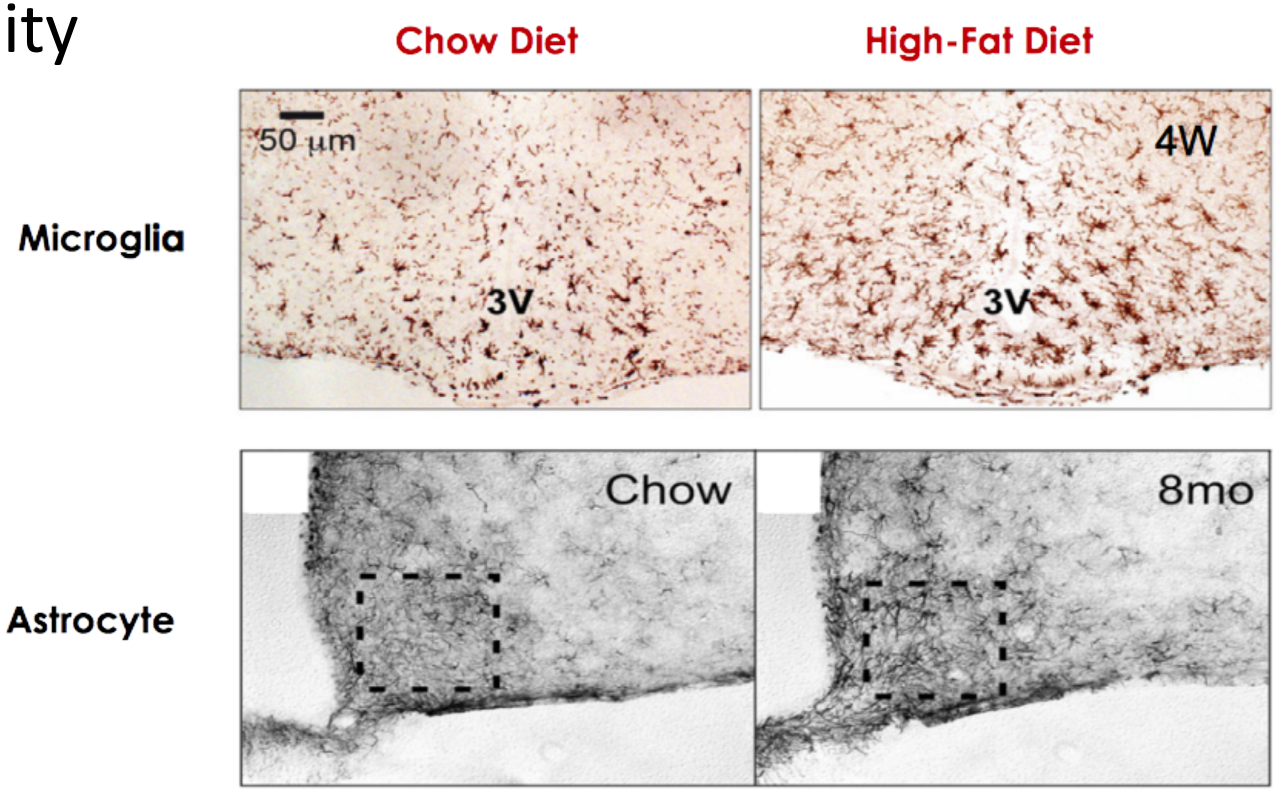
Gabor filter : example





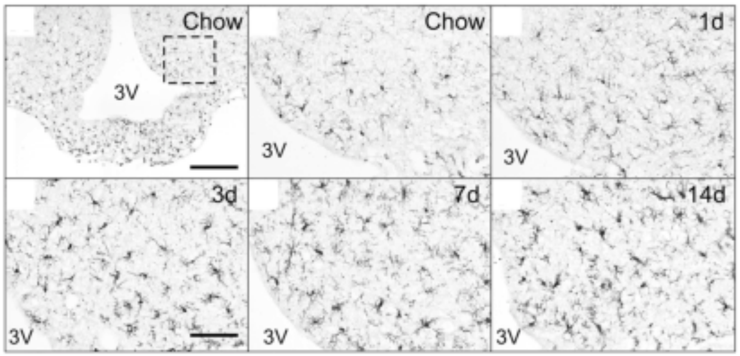
Case study : Detection and classification of neuronal extensions on fluorescence microscopy images: application to the study of metabolic diseases such as obesity

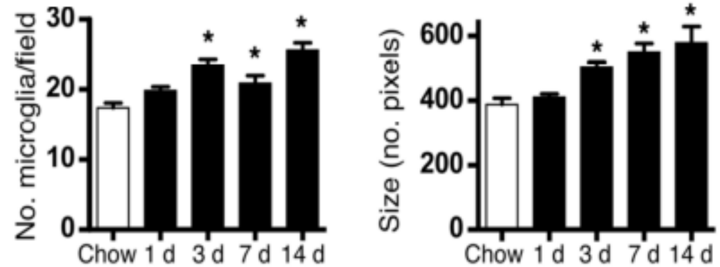
- Biological context :
  - Glial cells reactivity associated with diet induced obesity



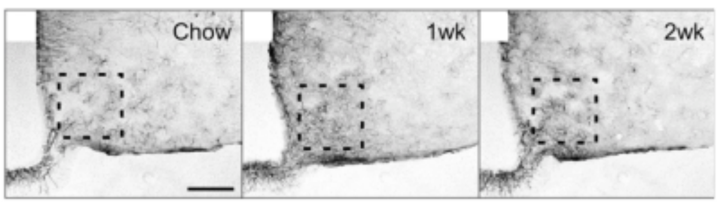
- **High fat diet (HFD) induces glial cells reactivity before the onset of obesity**

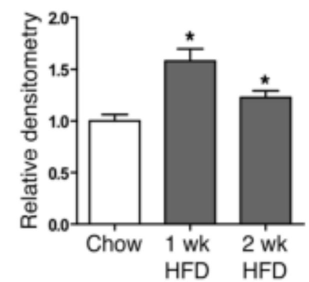
Microglia





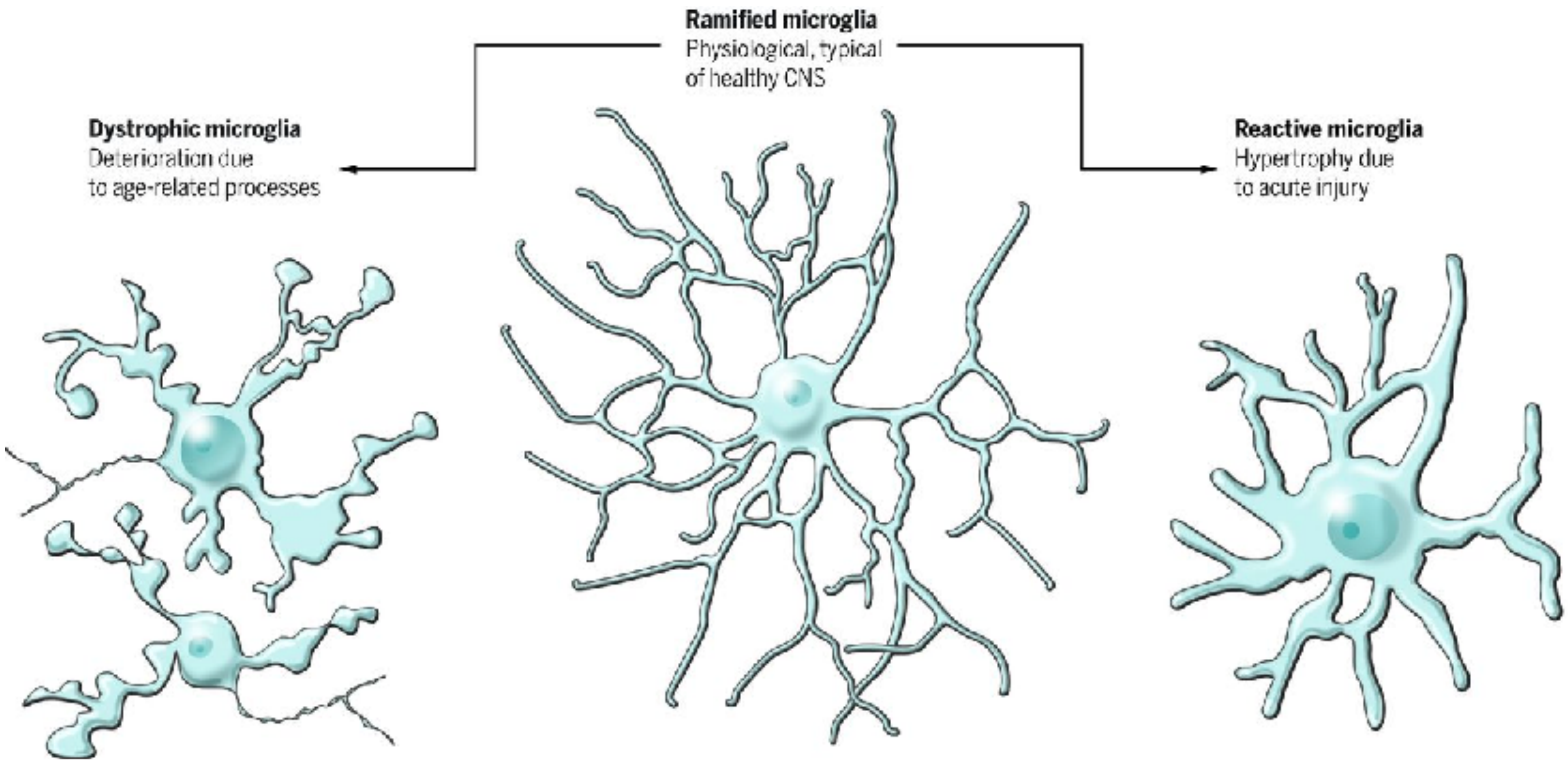
Astrocyte





Does astrocytes and microglia hypothalamic reactivity appear at meal scale in response to HFD?

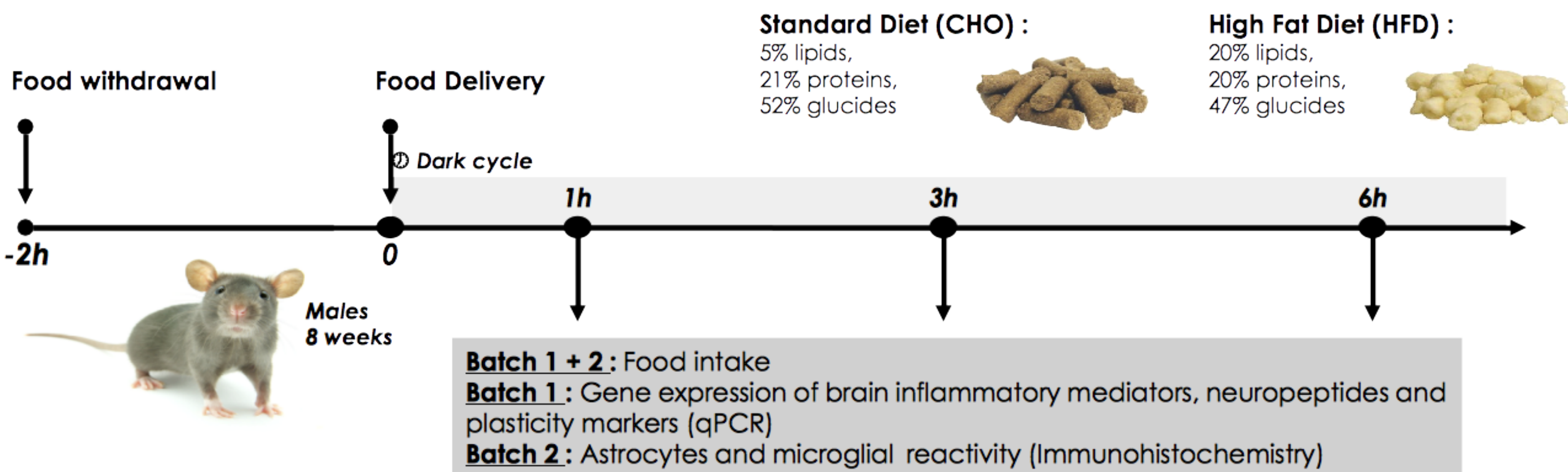
Thaler JP, Yi CX et al. JCI 2012



Morphology of ramified (healthy CNS), reactive, and dystrophic microglia.

Does astrocytes and microglia hypothalamic reactivity appear at meal scale in response to High fat diet (HFD)?

Protocol


Food withdrawal
Food Delivery
Standard Diet (CHO) :
5% lipids,
21% proteins,
52% glucides
High Fat Diet (HFD) :
20% lipids,
20% proteins,
47% glucides
Dark cycle
1h
3h
6h
-2h
0
Males
8 weeks
Batch 1 + 2 : Food intake
Batch 1 : Gene expression of brain inflammatory mediators, neuropeptides and plasticity markers (qPCR)
Batch 2 : Astrocytes and microglial reactivity (Immunohistochemistry)

Goal

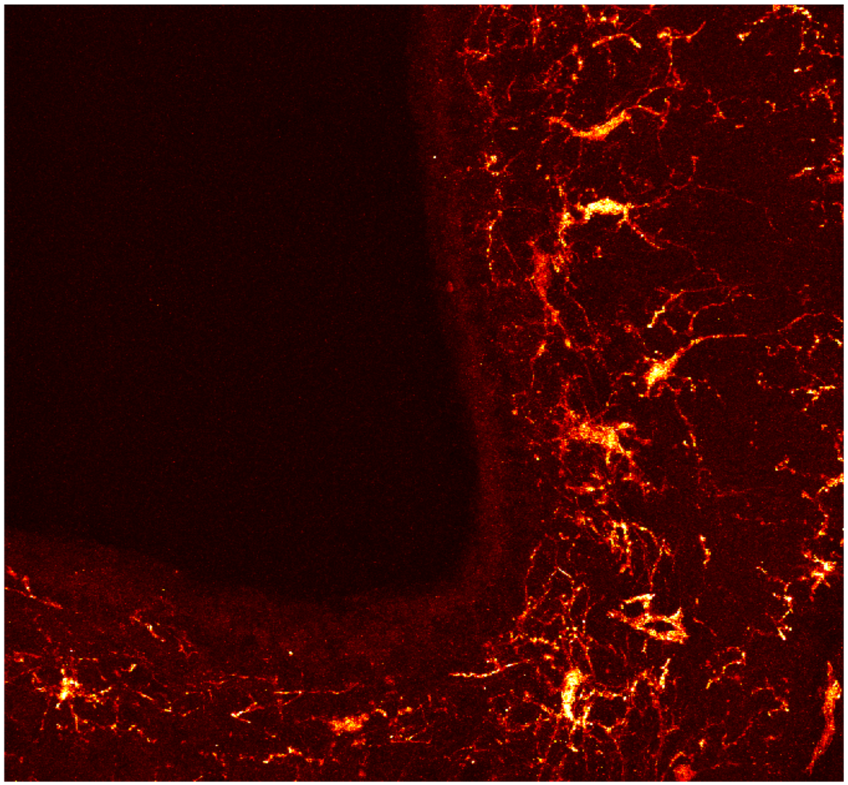
Classify images of neuronal cells into mice fed normally and mice fed with a high-fat diet.

- Two type of cells → **Microglia** / **Astrocyte**
- Two different areas of interest of the hypothalamus → **Median Eminence (EM)** / **Arcuate Nucleus (ARC)**
- 3D images at different times (1, 3 et 6 hours)

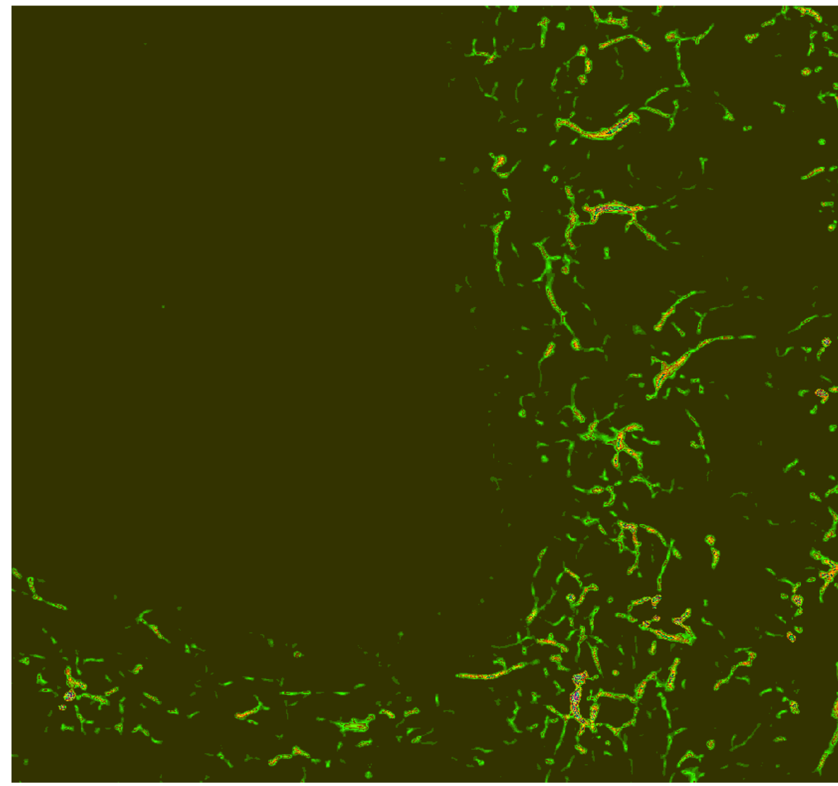
The four steps

- Extensions detection
- Somas detection
- Connection between filaments/soma and filaments/filaments
- Feature extraction

Extensions detection : filament enhancement

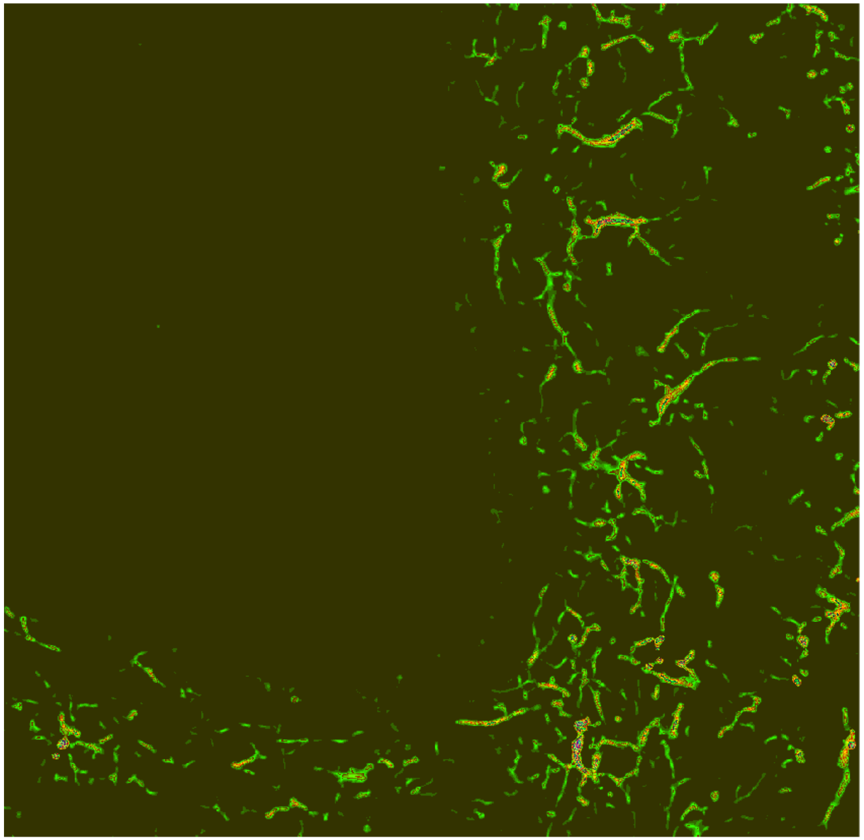


Original MIP (Maximum intensity projection)

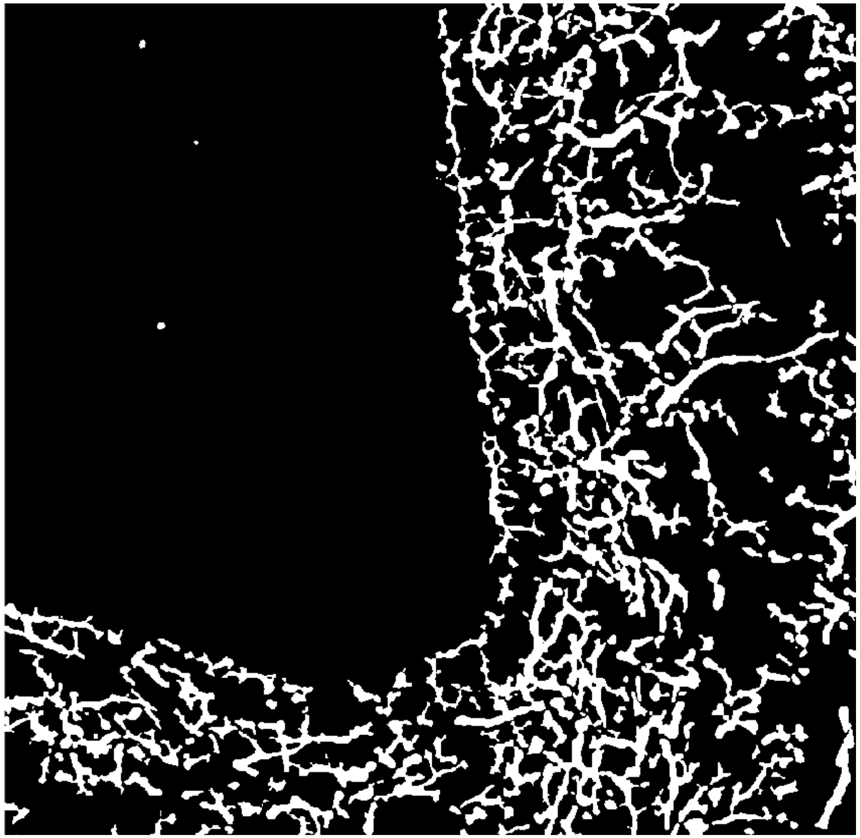


MIP after Frangi filter

Extensions detection : threshold by hysteresis

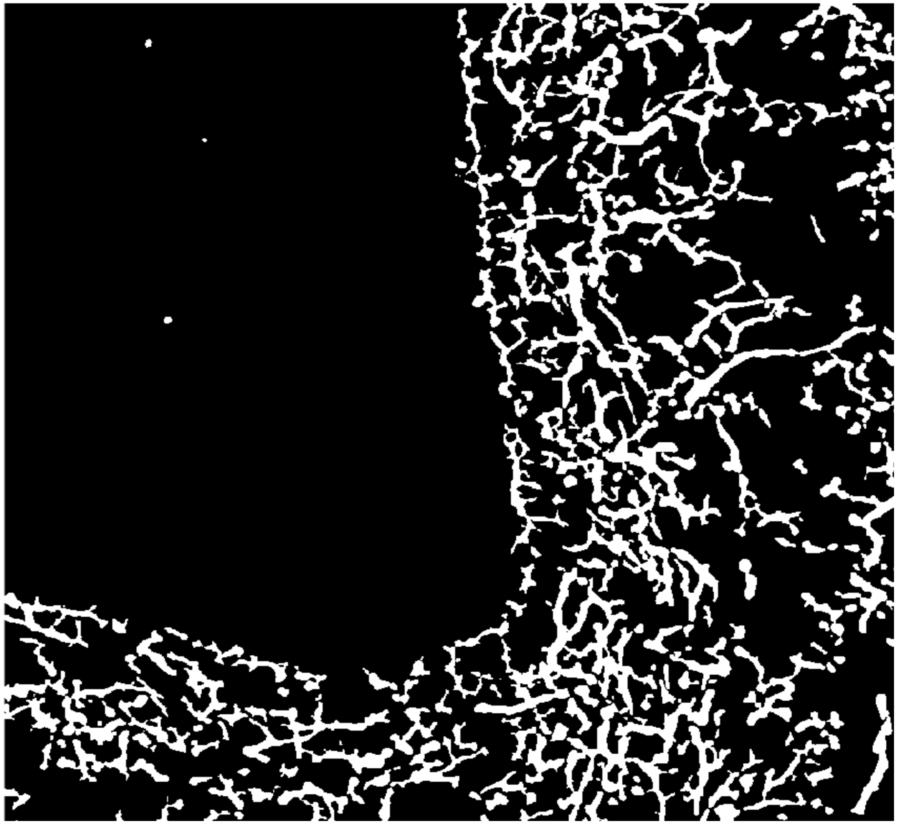


MIP Frangi

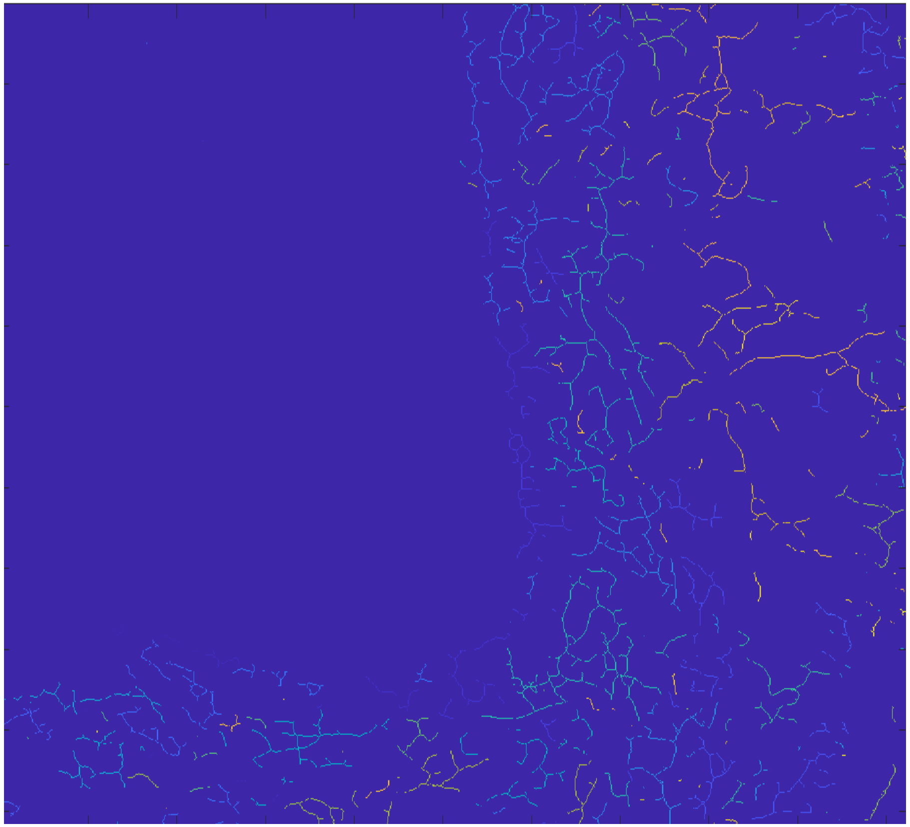


MIP Frangi binarized

Extensiosn detections : skeletonization and labelling

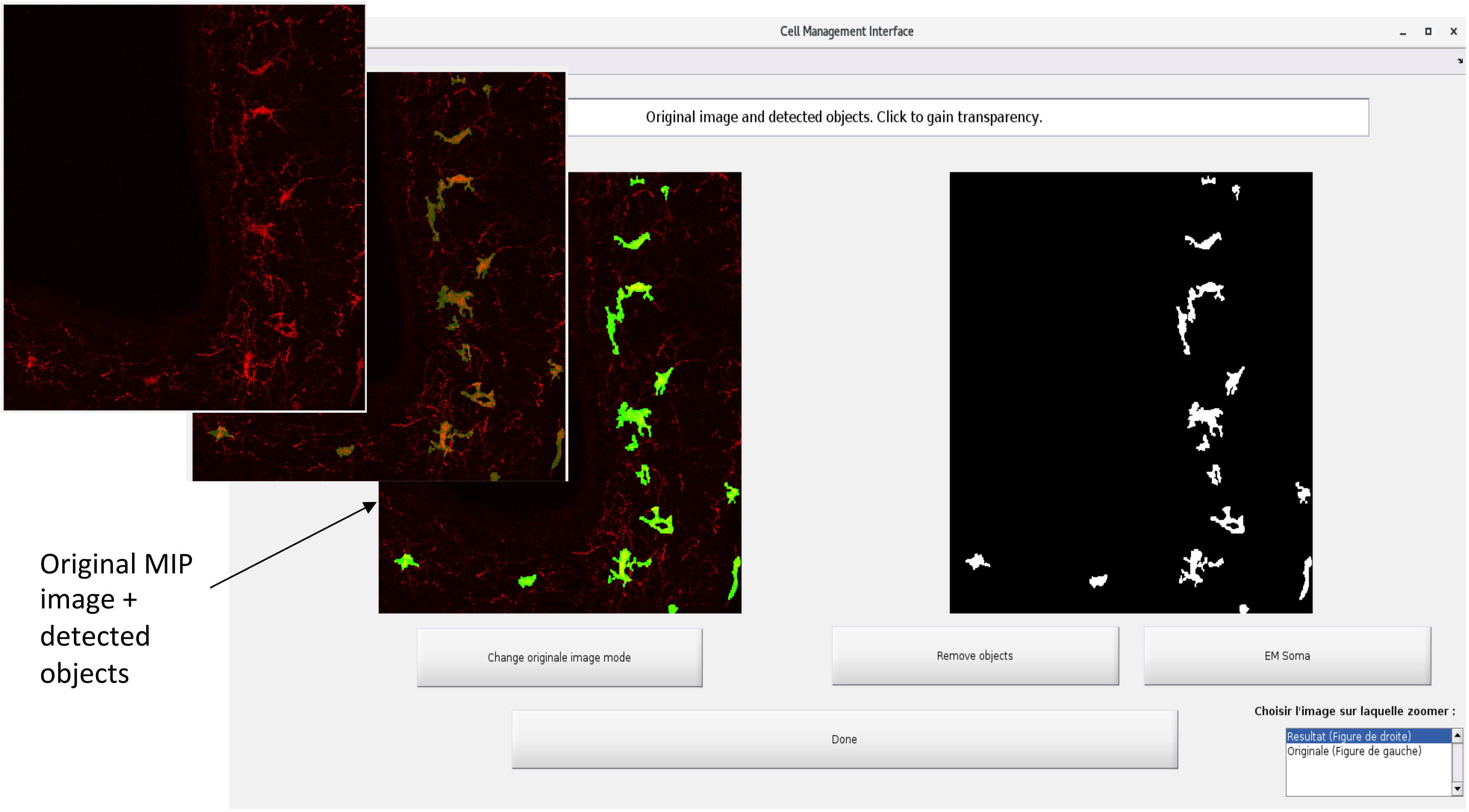


MIP Frangi binarized



MIP of labeled skeleton

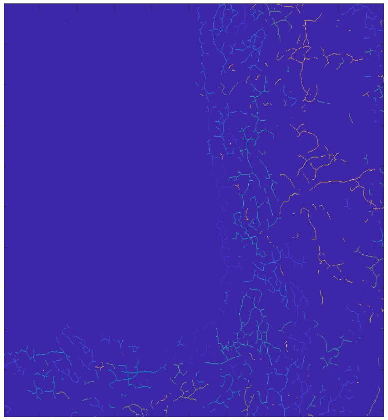
Somas detection



Original MIP image + detected objects

Connection between filaments/soma and filaments/filaments

- Closest filaments

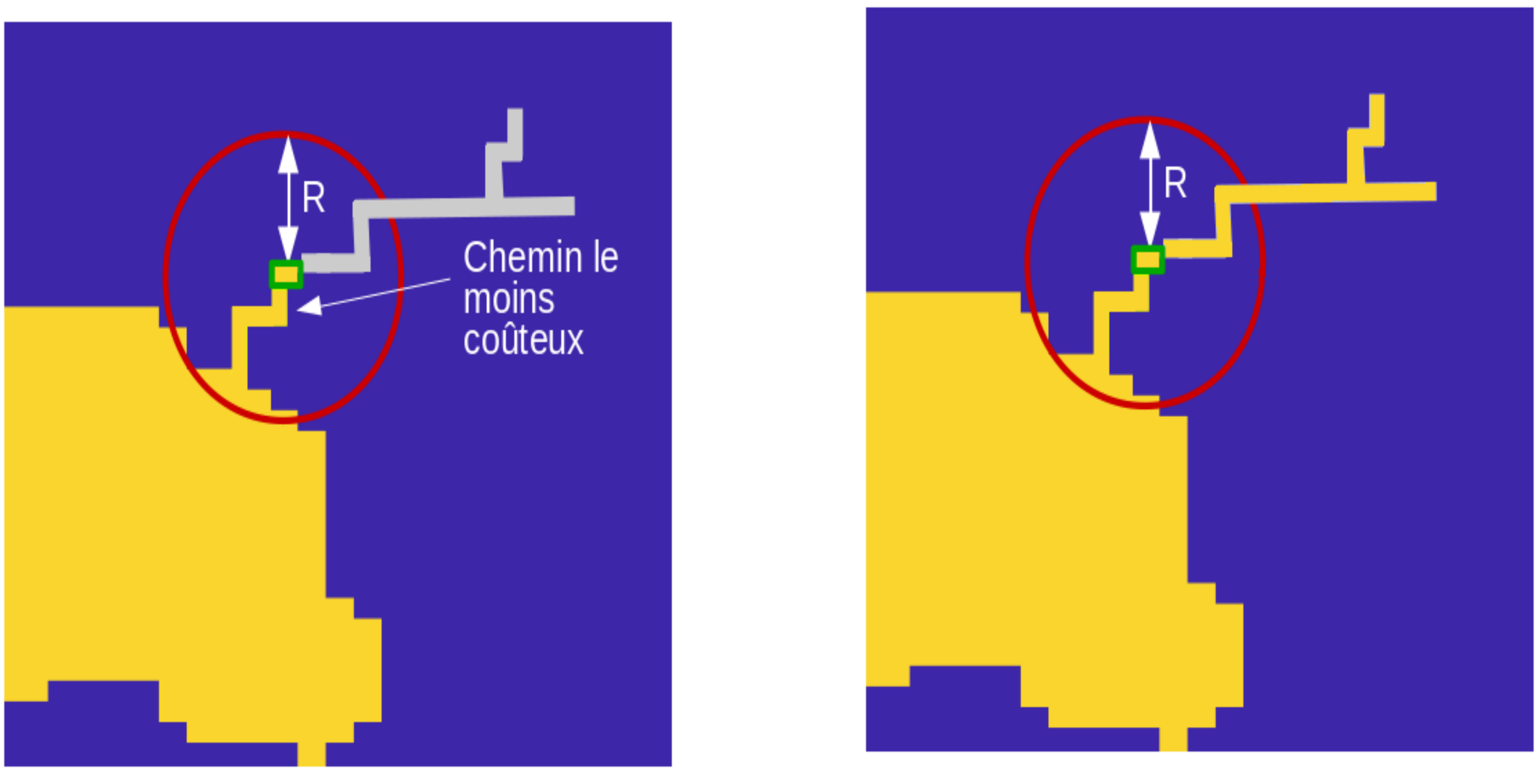


MIP extensions

MIP somas

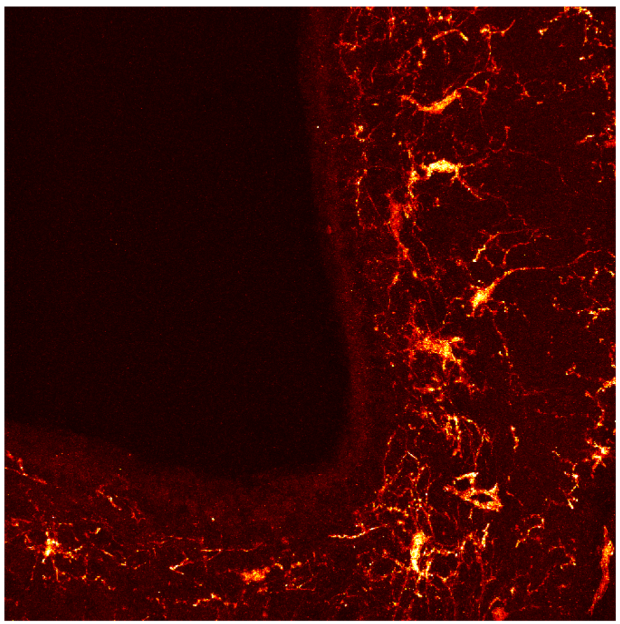
Connection between filaments/soma and filaments/filaments

- Minimal path : dynamic programming

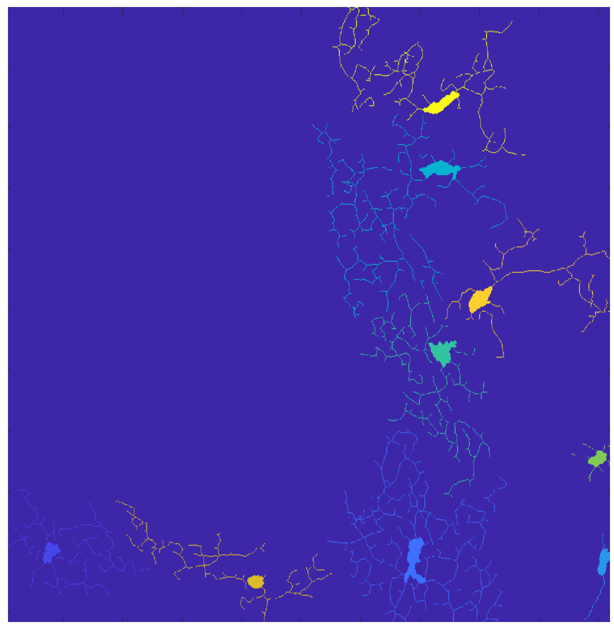


Connection between filaments/soma and filaments/filaments

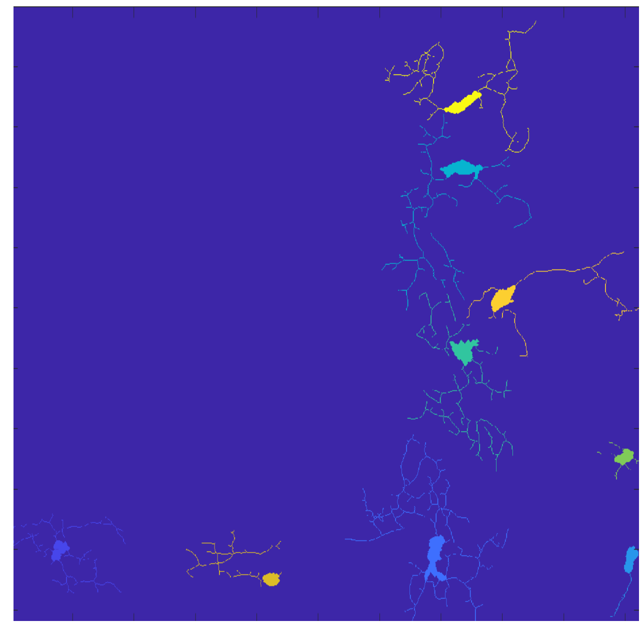
- Threshold on path energy



Original MIP image



Threshold on energy = **Inf**



Threshold on energy = **0.2**

Features extraction

- **Microglia:**

- **Soma**
-
- - Number of main branches
- - Area, diameter and eccentricity of the soma
- - Area in which it is located (EM/ARC)

- **Extensions**

- - Total length
- - Width (average / median / variance)

- **Main branches**

- - Median lengths
- - Average number of branches

- **Branches**

- - Average length / median / variance