Image Processing

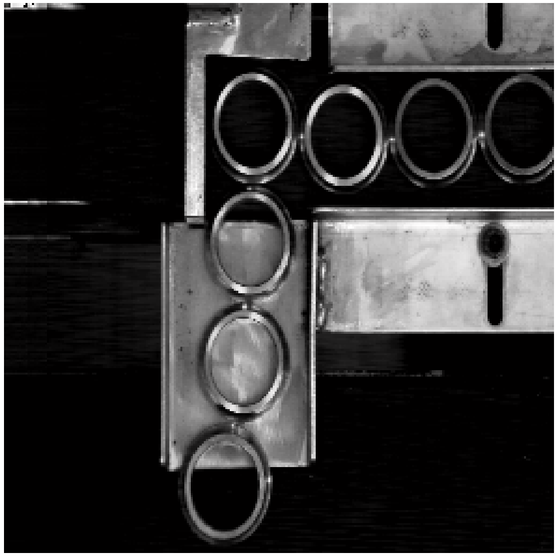
Traitement d'images

Yuliya Tarabalka
yuliya.tarabalka@hyperinet.eu
yuliya.tarabalka@gipsa-lab.grenoble-inp.fr
Tel. 04 76 82 62 68

Examples of image processing

Example 1:

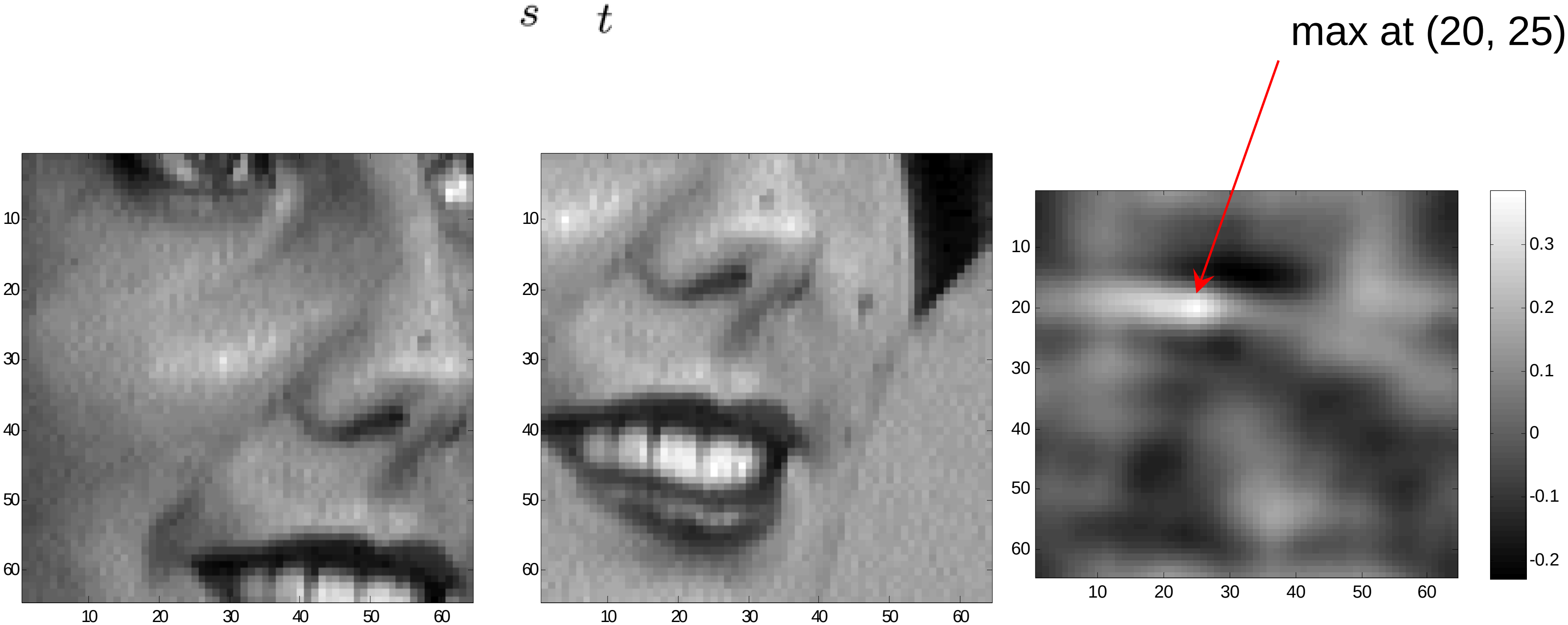
- **We would like to <u>automatically</u> detect and count rings in the image**



Detection by correlation

- **Correlation = degree of similarity**
- **Correlation between *f(x, y)* and *h(x, y)***

$$c(x,y) = \sum_{s} \sum_{t} f(s,t)h(x+s, y+t)$$



Detection by correlation

- **Correlation = degree of similarity**
- **Correlation between *f(x, y)* and *h(x, y)***

$$c(x,y) = \sum_s \sum_t f(s,t)h(x+s,y+t)$$

⇩

sensitive to changes in the amplitude of f and h

- **Correlation coefficient**

$$\gamma(x,y) = \frac{\sum_s \sum_t [f(s,t) - \bar{f}(s,t)][h(x+s,y+t) - \bar{h}]}{\left\{\sum_s \sum_t [f(s,t) - \bar{f}(s,t)]^2 [h(x+s,y+t) - \bar{h}]^2\right\}^{1/2}}$$

⇩

scaled between -1 to 1, scale-independent

Detection by correlation

- **Correlation = degree of similarity**
- **Correlation between *f(x, y)* and *h(x, y)***

$$c(x,y) = \sum_s \sum_t f(s,t)h(x+s,y+t)$$

sensitive to changes in the amplitude of f and h

Significant computational time

- **Correlation coefficient**

$$\gamma(x,y) = \frac{\sum_s \sum_t [f(s,t) - \bar{f}(s,t)][h(x+s,y+t) - \bar{h}]}{\left\{\sum_s \sum_t [f(s,t) - \bar{f}(s,t)]^2 [h(x+s,y+t) - \bar{h}]^2\right\}^{1/2}}$$

scaled between -1 to 1, scale-independent

Detection by correlation

- **Correlation = degree of similarity**
- **Correlation between *f(x, y)* and *h(x, y)***

$$f(x,y) \circ h(x,y) = c(x,y) = \sum_s \sum_t f^*(s,t)h(x+s, y+t)$$

- **Correlation theorem**

$$f(x,y) \circ h(x,y) \Leftrightarrow F^*(u,v)H(u,v)$$

$$f^*(x,y)h(x,y) \Leftrightarrow F(u,v) \circ H(u,v)$$

Phase correlation method

Phase correlation method

Given two input images f and h

- Calculate the DFT of both images

$$F = FT\{f\} \qquad\qquad H = FT\{h\}$$

- Calculate the cross-power spectrum by taking the complex conjugate of the second result, multiplying the FTs together elementwise, and normalizing this product elementwise

$$C = \frac{FH^*}{|FH^*|}$$

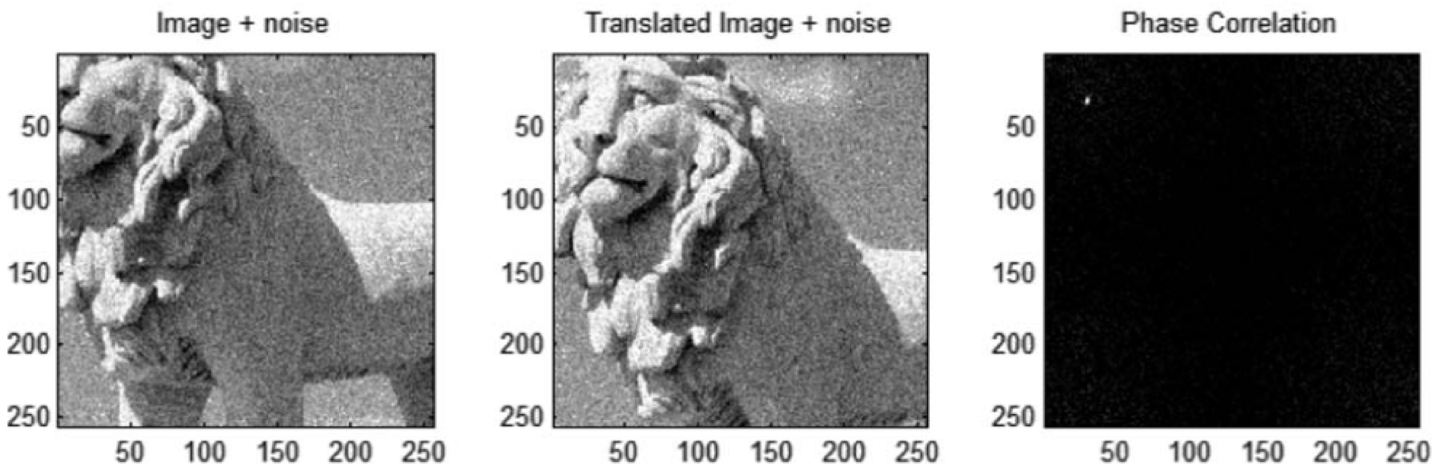
Phase correlation method

- **Obtain the normalized cross-correlation by applying the inverse Fourier transform**

$$c = FT^{-1}\{C\}$$

- **Determine the location of the peak in *c***

$$(x_{max}, y_{max}) = \arg\max_{x,y}\{c\}$$



Phase correlation method

- **Obtain the normalized cross-correlation by applying the inverse Fourier transform**

$$c = FT^{-1}\{C\}$$

- **Determine the location of the peak in c**

$$(x_{max}, y_{max}) = \arg\max_{x,y}\{c\}$$

$$\Downarrow$$

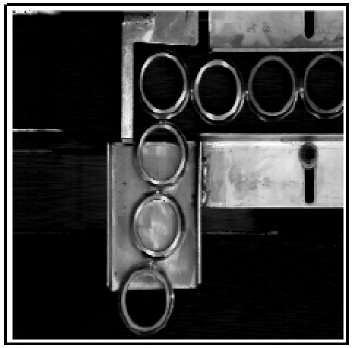
The method is resilient to noise ☺

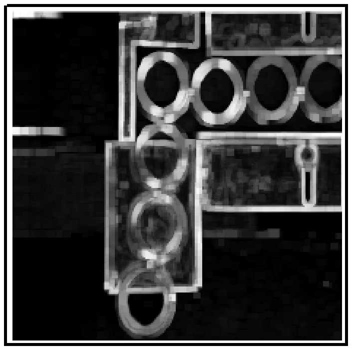
Correlation in the spatial versus frequential domain

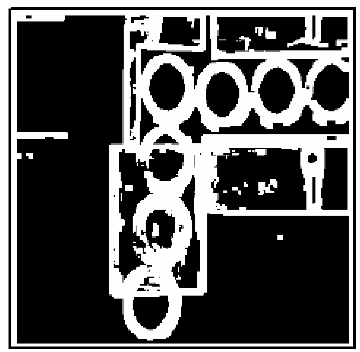
- **Trade-off estimate performed by Campbell [1969]:**

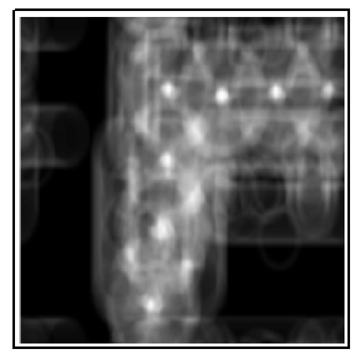
  if the number of nonzero terms in h (smaller image) is less than 132 (approximately 13x13 pixels), correlation in the spatial domain is faster than the FFT approach

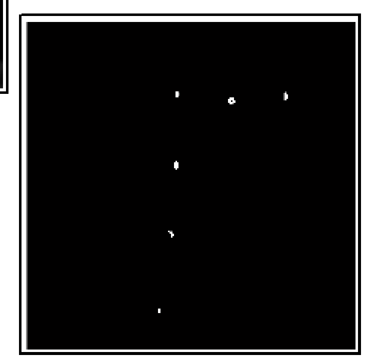
Detection by correlation



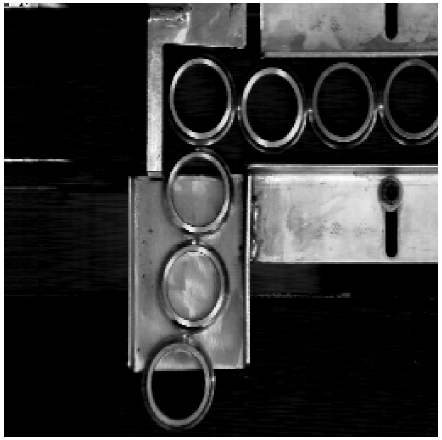


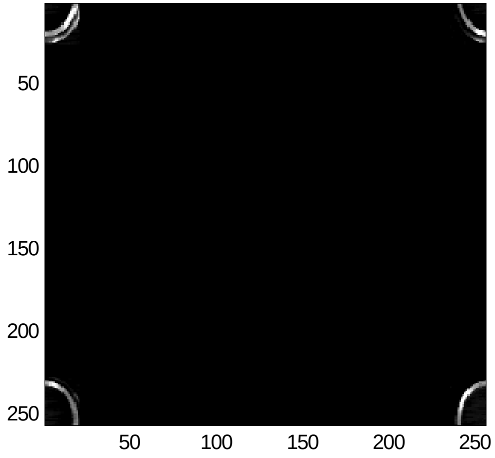




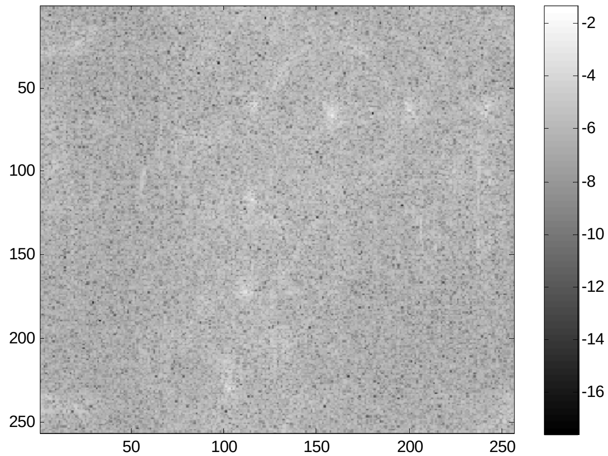


Detection by correlation (2^{nd} trial)

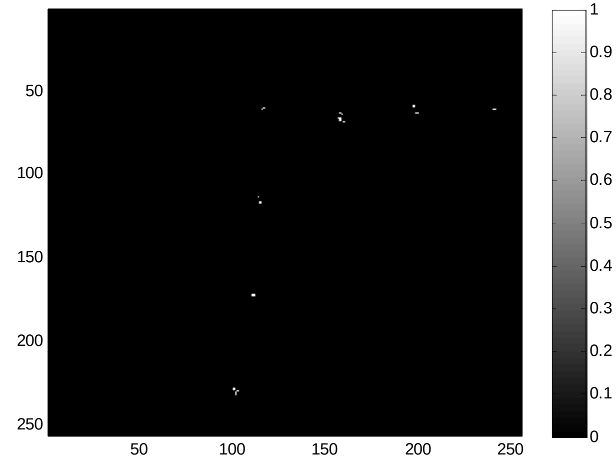


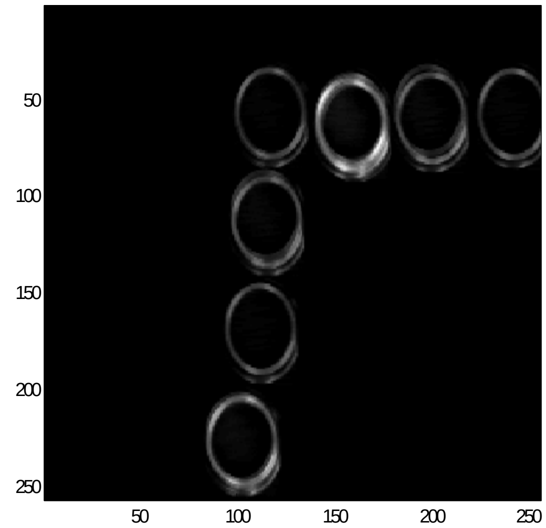


log(phase correlation)



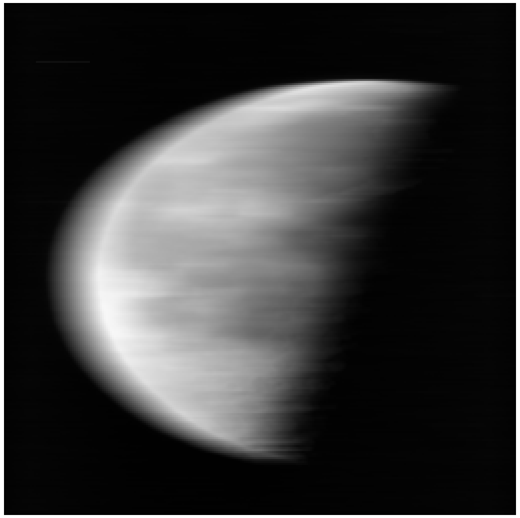
after thresholding





Example 2: Unhook the moon

- **One would like to analyze the surface of the moon**
- **The acquired image is of poor quality ☹**
  - ➢ A movement of the camera during the long-exposure acquisition led to a fuzzy image



Modelling the problem (of image degradation)

- **The resulting grey level of each pixel = average grey level “seen” by the pixel during the movement**
- **Assuming the movement is rectilinear, this is modelled by a convolution of the original image f(x, y) by a linear mask whose size corresponds to the movement amplitude**
  - f(x, y) = original image (unknown)
  - h(x, y) = convolution mask or impulse response (length unknown)
  - g(x, y) = observed fuzzy image

$$\boxed{g = f * h}$$

Modelling the problem (of image degradation)

- $$\boxed{g = f * h}$$

- **There can also be some unknown noise b(x, y). Assuming it is additive:**

$$\boxed{g = f * h + b}$$

$$\Big\downarrow \text{ FT}$$

$$\boxed{G = F \cdot H + B} \Rightarrow \boxed{F = G/H - B/H}$$

- **Assuming the noise can be neglected:** $\boxed{\widehat{F} = G/H}$

Modelling the problem (of image degradation)

- **Assuming the noise can be neglected:** $\boxed{\widehat{F} = G/H}$

- **The original image can then be recovered by computing the inverse FT**

- **This restoration technique is called by** **(pseudo) inverse filtering**
  - f(x, y) is restored thanks to a filter whose transfer function is 1/H

Modelling the problem (of image degradation)

- **This restoration technique is called by** **(pseudo) inverse filtering**

$$\boxed{\widehat{F} = G/H}$$

- **However, directly applied, this method may fail**
  - For the frequencies where H gets null (or takes values very close to zero), B/H cannot be neglected any longer: the restored image is then heavily corrupted by the noise.
  - This problem can be tackled by using a filter whose transfer function is 1/H when |H| is greater than a given threshold, and zero otherwise (pseudo-inverse filtering)

Identifying the model's parameters

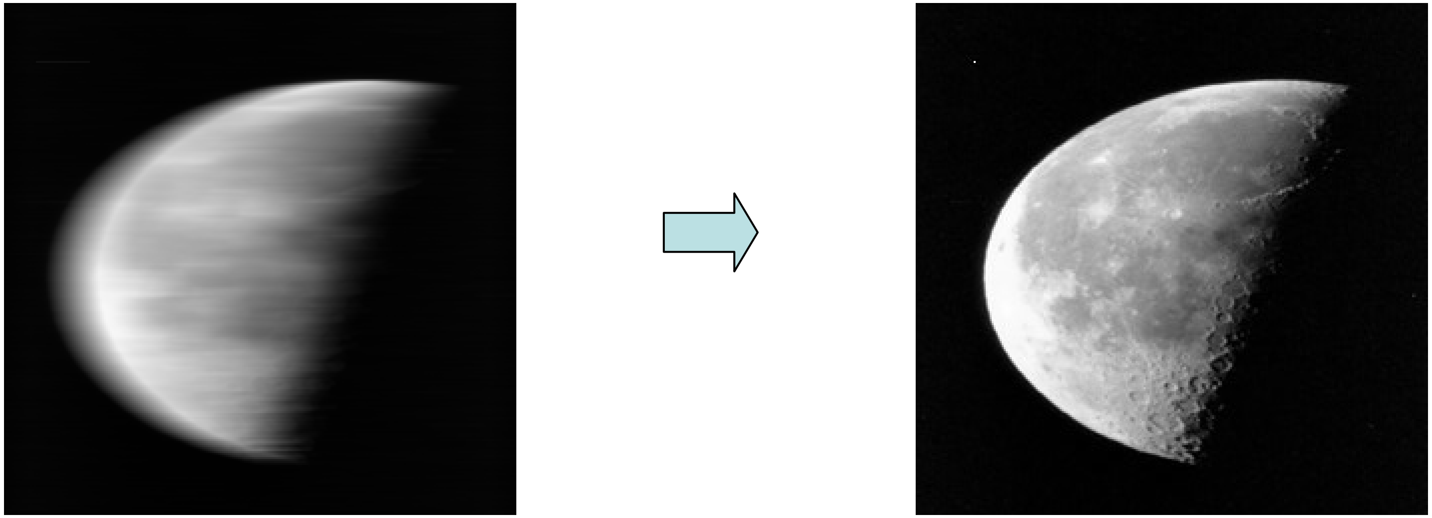
- **With astronomic images, one can be lucky… if a far star is in the black background of *f(x, y)*, it can be consider as a Dirac. In *g(x, y)*, it then directly leads to the impulse response of the filter that degraded the image (i.e. *h*)…**

- **Look for such a star in the dark regions of *g* (use profiles, thresholds…)**

- **If one can't directly estimate *h* from *g*, another solution consists in the trial-error strategy**

Using the identified model

- **The image can now be restored**

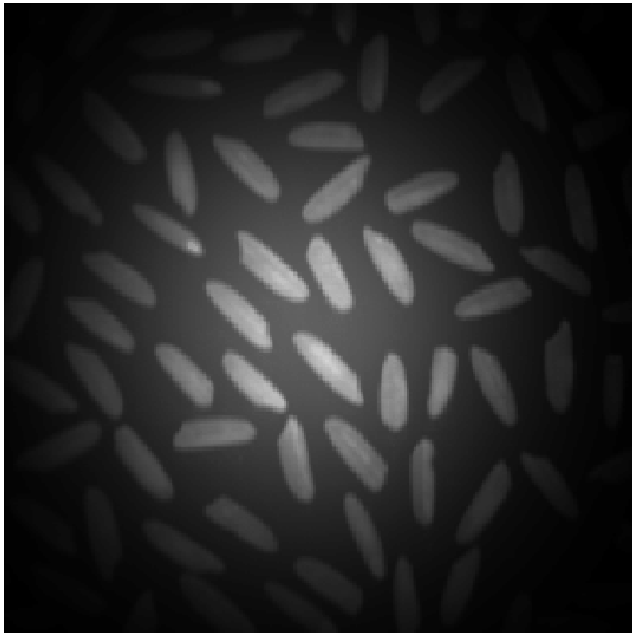


- **It is possible to further improve the result (contrast, edges…)**

Example 3: A portion of rice

- **One would like to analyze a picture featuring some rice seeds**
- **The aim is to obtain a segmented binary image**

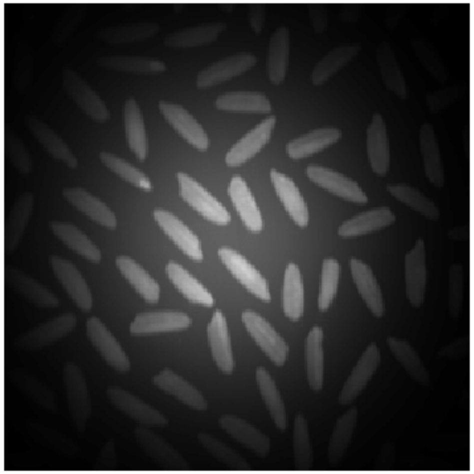
☹ The background is not uniform
☹ Neither are the seeds

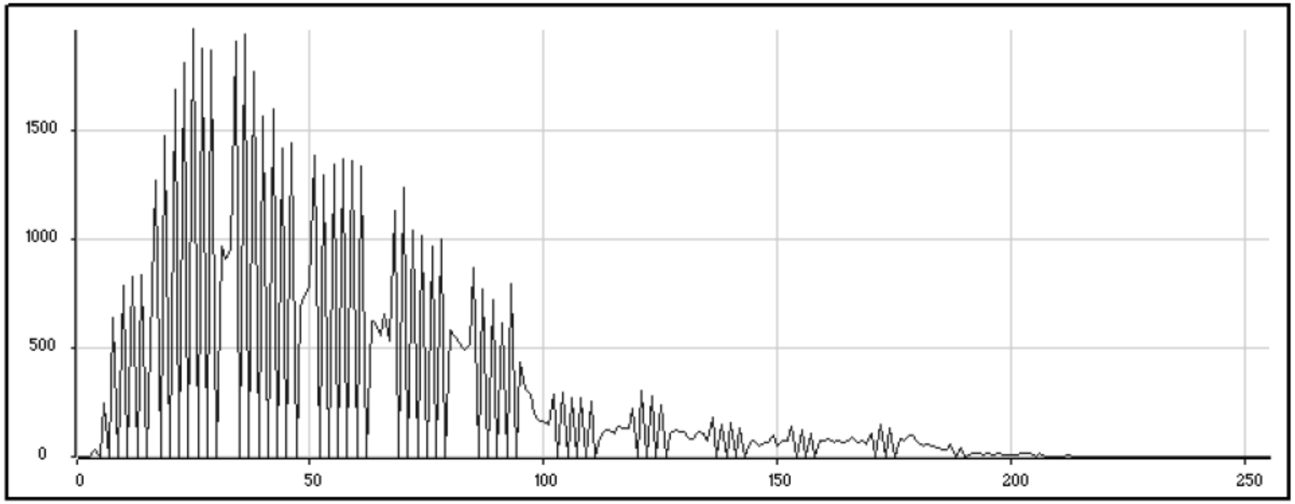


256x256 image
Every pixel is coded by 8 bits

A portion of rice

- **Image histogram**

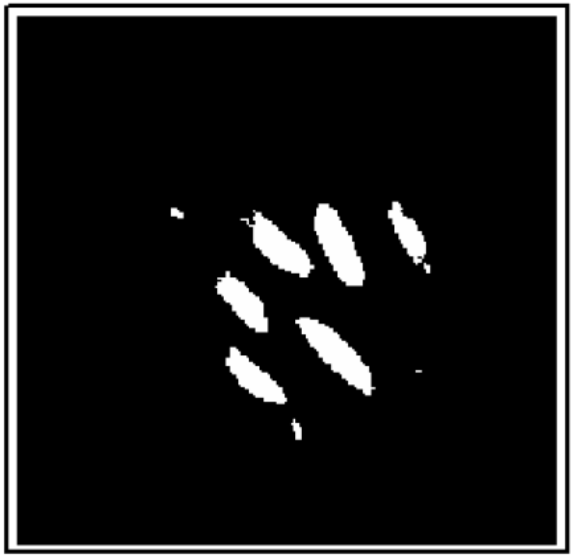




➢ We do not clearly see distinct classes

A portion of rice

- **Thresholding**
  - ➢ Rice seeds are bright

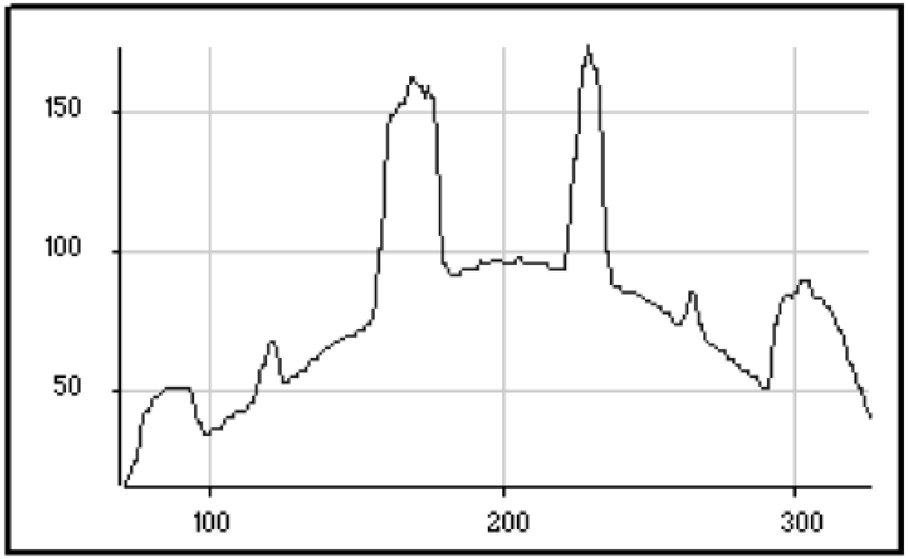


Threshold = 150
Central seeds are detected,
but peripheral seeds are missing

Threshold = 50
Central seeds are aggregated
with the part of the background

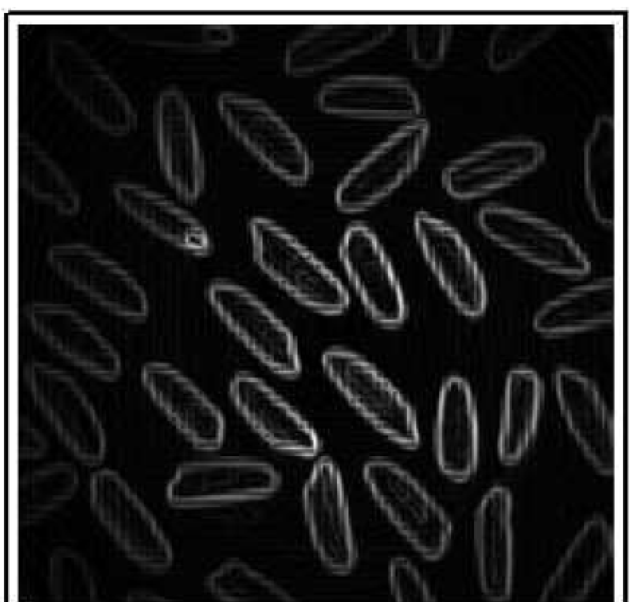
A portion of rice

- **A slice of the original image (horizontal line in the center)**
  - ➢ The background of the image is not uniform



A portion of rice

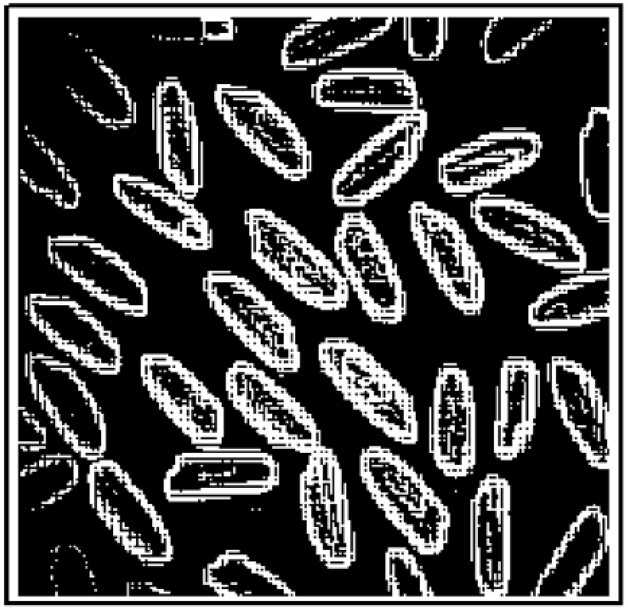
- **Edge detection**
  - Seeds do not have a homogeneous grey level in the image
  - However, they are (almost) always brighter than their neighborhood
  - Sobel filtering for edge detection



The next step is to “close” these edges for further isolating every seed

A portion of rice

- **Edge detection**
  - ➢ Seeds do not have a homogeneous grey level in the image
  - ➢ However, they are (almost) always brighter than their neighborhood
  - ➢ (Morphological dilation – original image) followed by thresholding



The next step is to “close” these edges for further isolating every seed

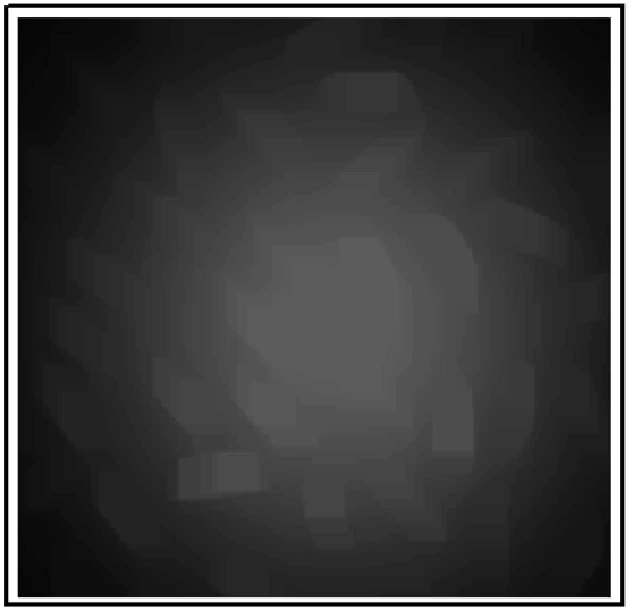
Estimation of the background

- **In practice: by calibration**
  - Empty image (without seeds) is used

- **If the empty image is not available?**

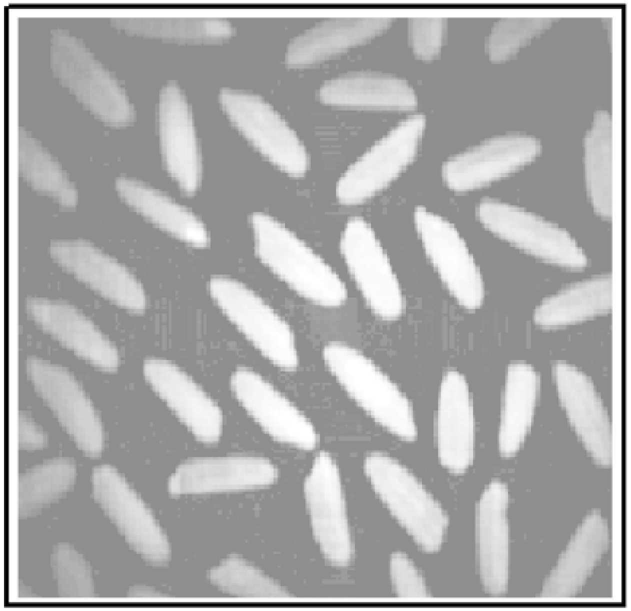
Estimation of the background

- **We will try to remove only seeds from the image**
- **Morphological opening with a structuring element larger (at least in one dimension) than the rice seeds**



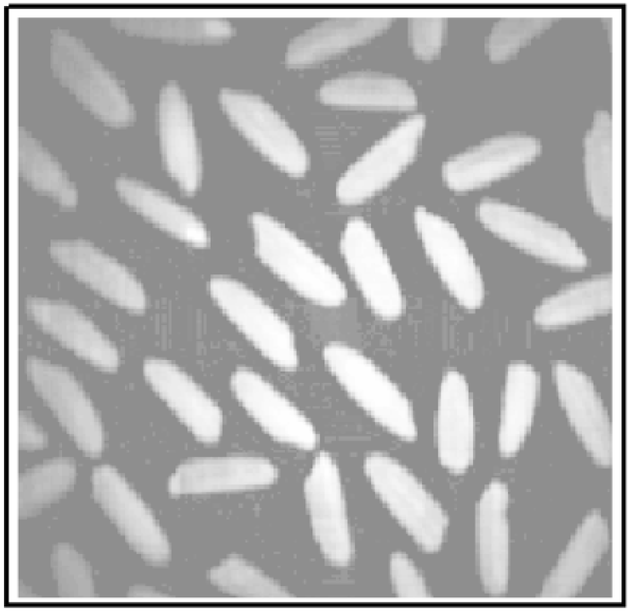
Estimation of the background

- **Morphological opening with a structuring element larger (at least in one dimension) than the rice seeds**
- **Original image – opening:**



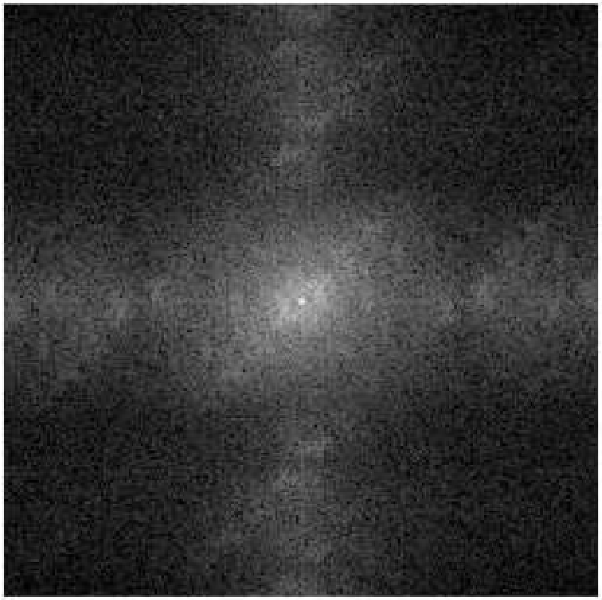
Estimation of the background

- **Original image – opening**
- **Thresholding**



Estimation of the background: frequential approach

- **We can consider that the intensity variations of the background are slower than the intensity variations of the seeds**
- **From here, the information related to the background is situated in the low frequencies**



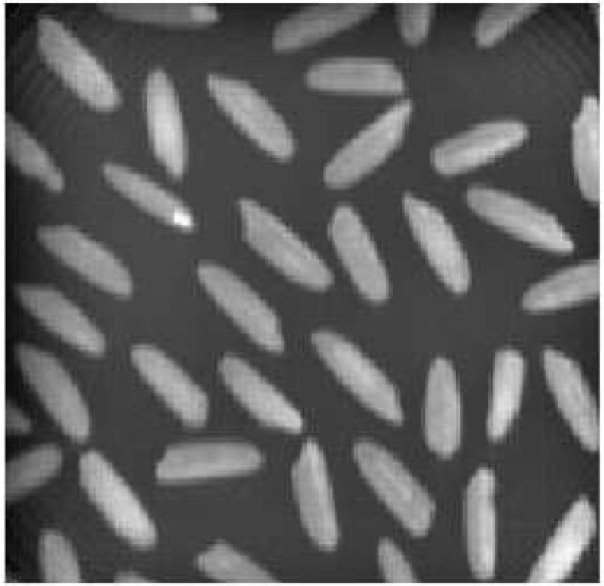
|FT(original image)|

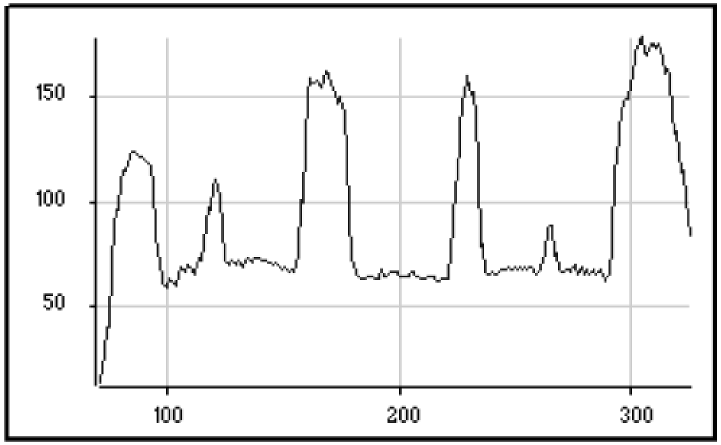
Estimation of the background: frequential approach

- **Lowpass filtering (Butterworth Lowpass) of the original image with a relatively low cutoff frequency**

Estimation of the background: frequential approach

- **To compensate the dome, we divide the original image by the (lowpass) filtered image (or substract two images)**





Now the background is compensated

Estimation of the background: frequential approach

- **Thresholding**