Fundamentals of Computer Vision

Unit 3: Fundamentals of Image Processing

Jorge Bernal

Index

1

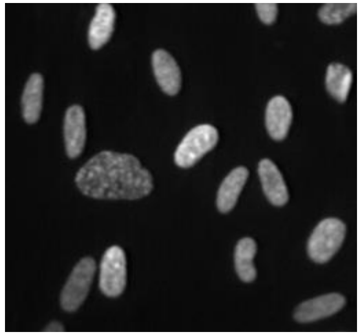
Digital images

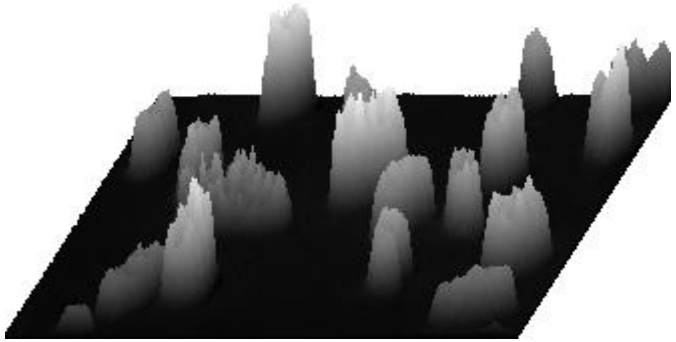
Digital image representation

- An image is a bidimensional function *I(x,y)* in which we represent the light intensity for a given point

- In monochromatic images, light intensity *I(x,y)* is known as grey level

- An image is a matrix data structure composed by pixels which contains discrete information. Sensor does this discrete transformation via two operations: sampling and quantization.





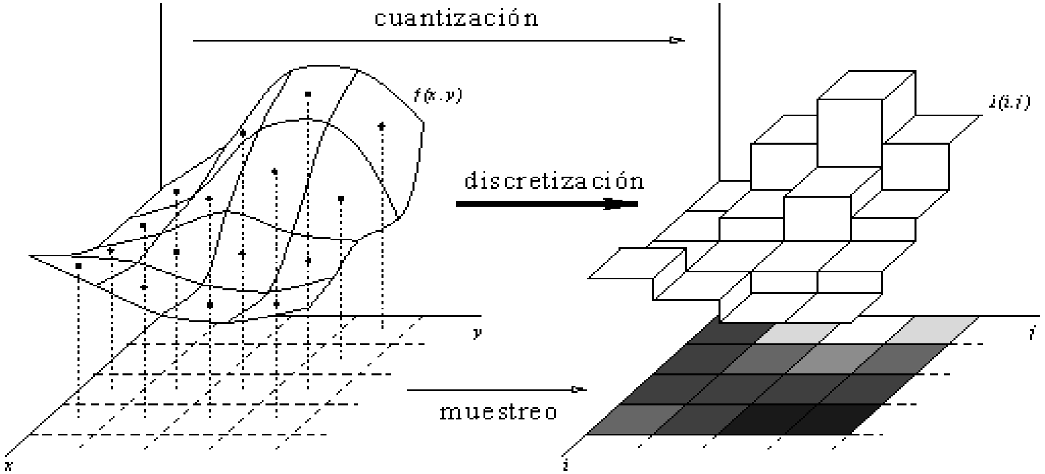
2

Sampling and quantization

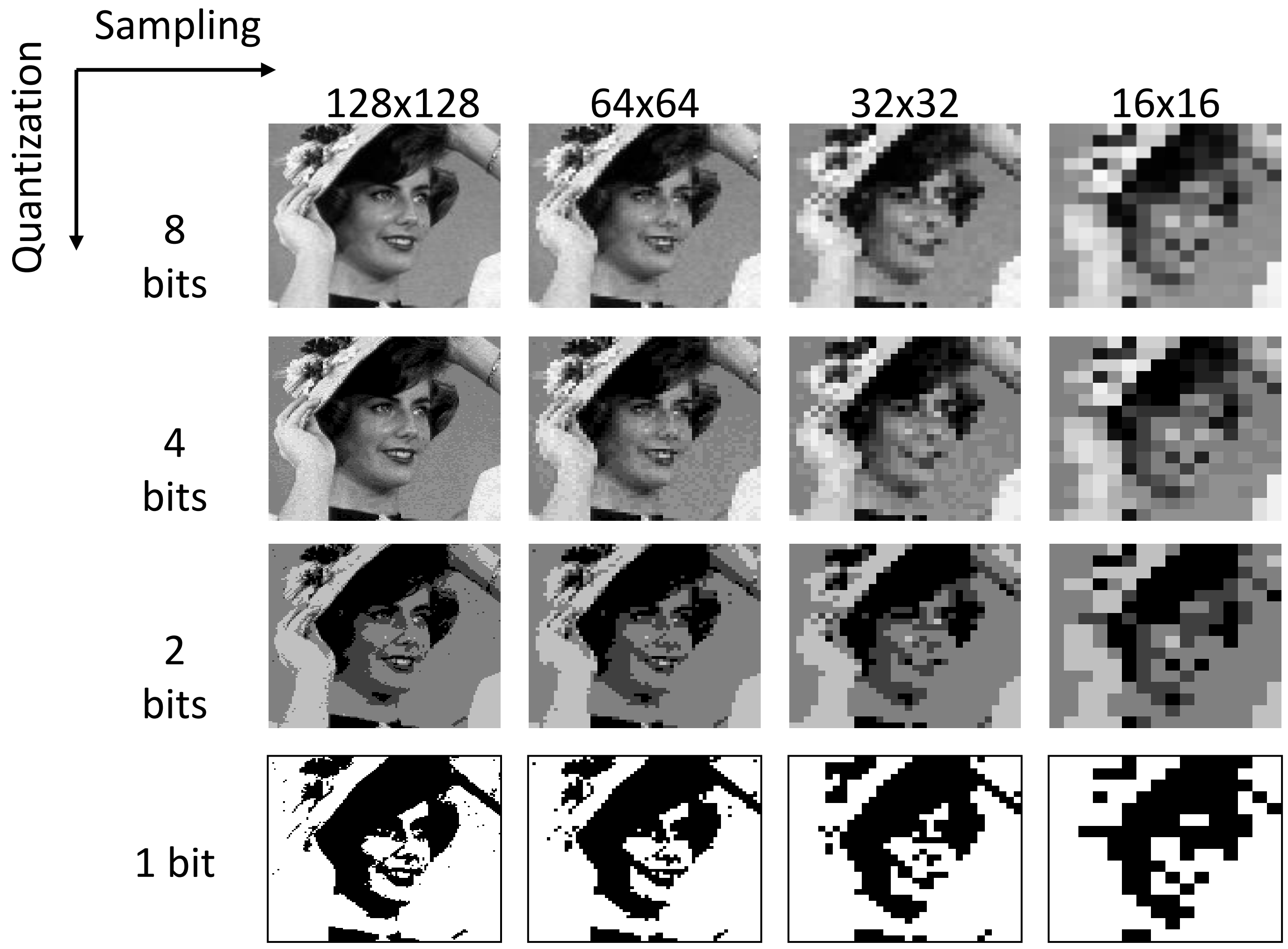
Sampling and quantization

- **Sampling:** discretization in space. This happens as a result of acquiring a continuous signal from real world with a discrete sensor.

- **Quantization:** discretization in amplitude / weight. It is generated as we store the information in a digital format (8 bits, 16 bits).



Sampling and quantization



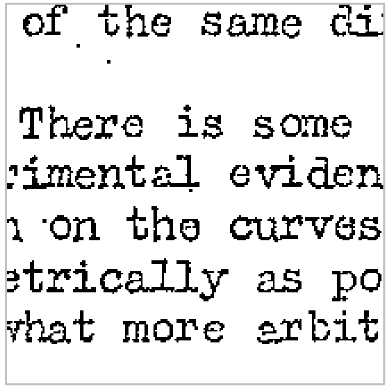
3

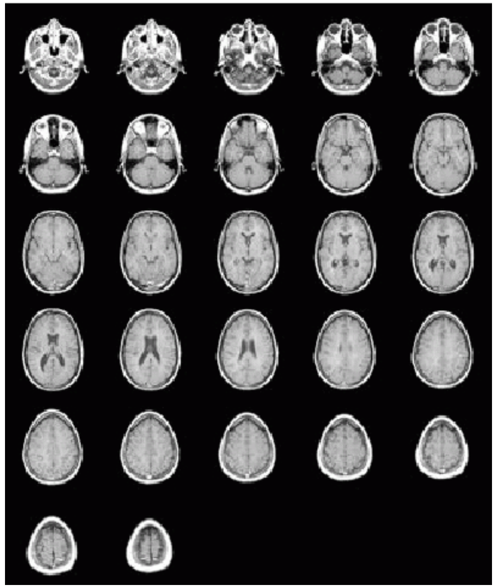
Image types

Image types

Image type	**Definition**
Binary	Only two values: black and white
Monochromatic	One single component, a single color
Color image	Three chromatic components: red, green, blue
Image sequence	Each image represents the scene in a given time instant $I(x,y,t)$
3D image	Images stacked along Z axis (sections of an object)
Image range	2D image where pixels indicate depth
Stereo image	Two images show the same image at the same time from two different points of view

Image types


of the same di
There is some
rimental eviden
n on the curves
etrically as po
what more erbit



4

Basic image enhancement operations

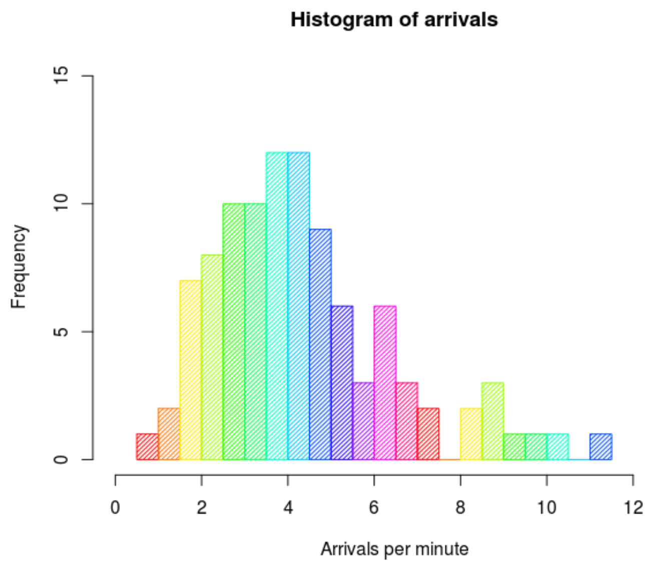
Basic image enhancement operations

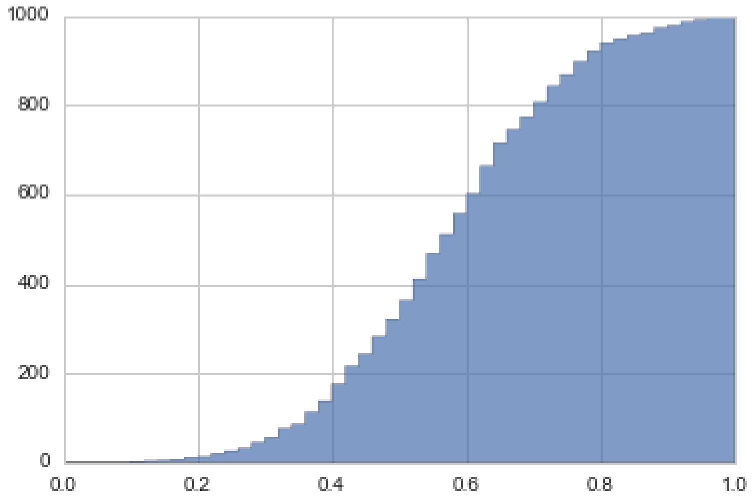
- **Image enhancement:** correction of the errors produced by image acquisition or transmission.

- Set of techniques which objective is to process an image to obtain a better one to achieve a certain purpose (better visualization, noise elimination). Two types:
  - Photometric corrections
  - Geometric corrections

Histogram

- Vector which has the same length as the different intensity levels in an image.
- **Histogram:** I-th position corresponds to the number of pixels in the image that have this intensity level.
- **Cumulative histogram:** I-th position represents the number of pixels with intensity level equal or less than I.

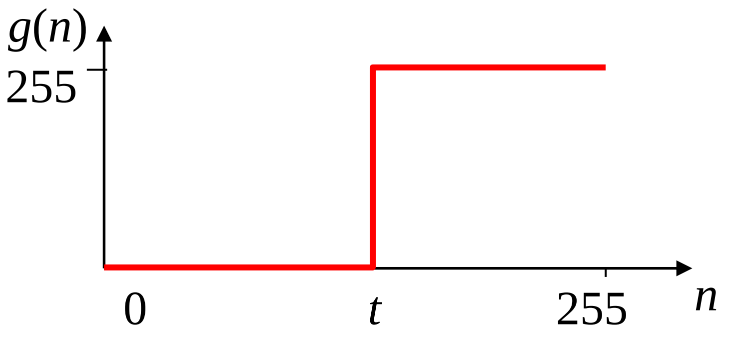




Thresholding

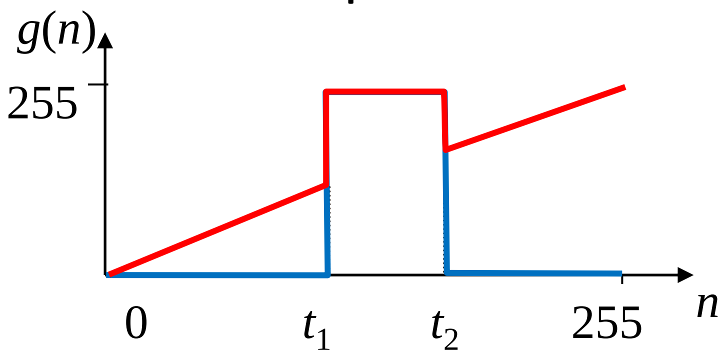
- **Single thresholding:** points with intensity higher or equal to a given threshold receive the maximum output: the rest the minimum value.

$$g(n) = \begin{cases} 255 & \text{if } I(x,y) \geq t \\ 0 & \text{otherwise} \end{cases}$$

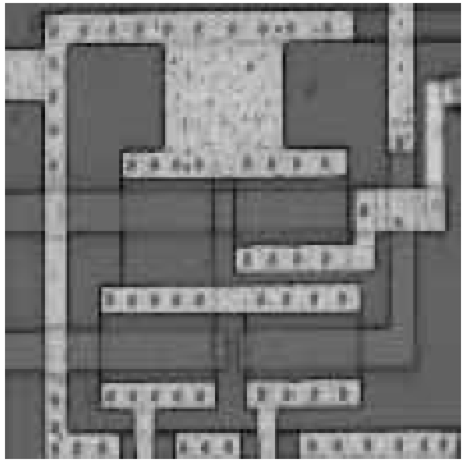


- **Double thresholding:** we select an interval $[t_1,t_2]$ where the output is maxima.

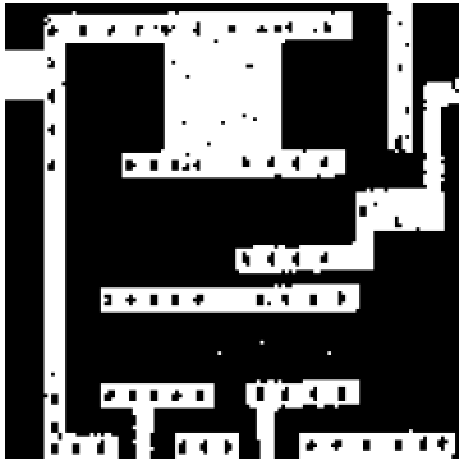
$$g(n) = \begin{cases} 255 & \text{if } t_1 \leq I(x,y) \leq t_2 \\ I(x,y) & \text{otherwise} \end{cases}$$



Thresholding

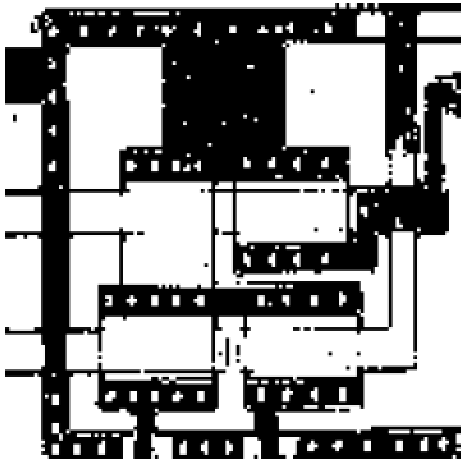


Original image



Single thresholding

(t =180)



Double thresholding

(t_1=100, t_2=150)

Linear transformations

- Transformation of an image I with a range of values (Gmin , Gmax) into a new image I' = g(I) with a range of values (Rmin , Rmax), being g a linear function.

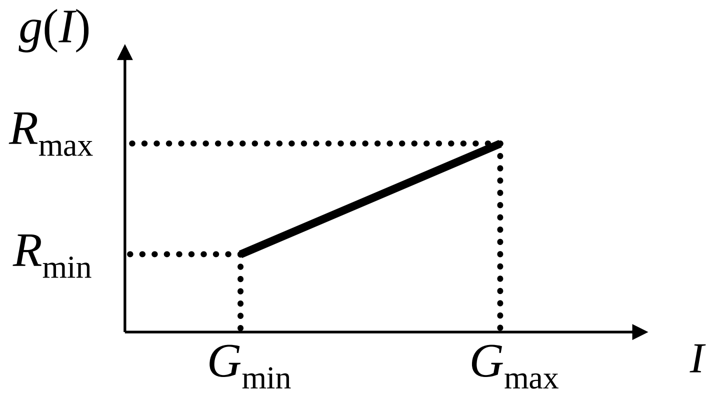
$$g(I(x,y)) = \frac{R_{max} - R_{min}}{G_{max} - G_{min}} (I(x,y) - G_{min}) + R_{min} =$$

$$= \frac{R_{max} - R_{min}}{G_{max} - G_{min}} I(x,y) - \left[\frac{R_{max} - R_{min}}{G_{max} - G_{min}} G_{min} - R_{min} \right]$$

Slope (m):

$m < 1$: contrast decrease

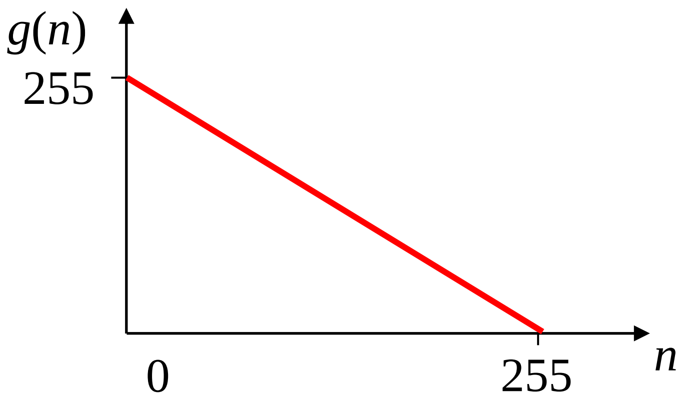
$m > 1$: contrast increase



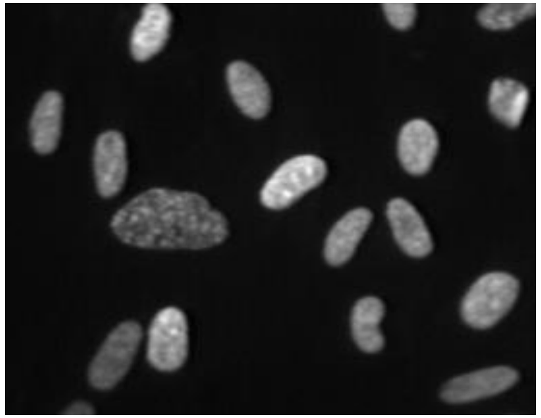
Negative image

- Inverse contrast

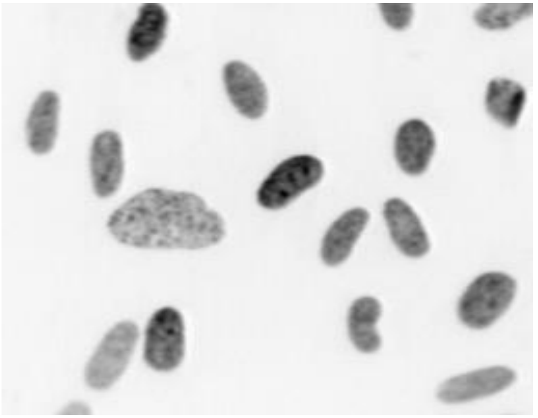
$$g(n) = 255 - I(x, y)$$



Original image



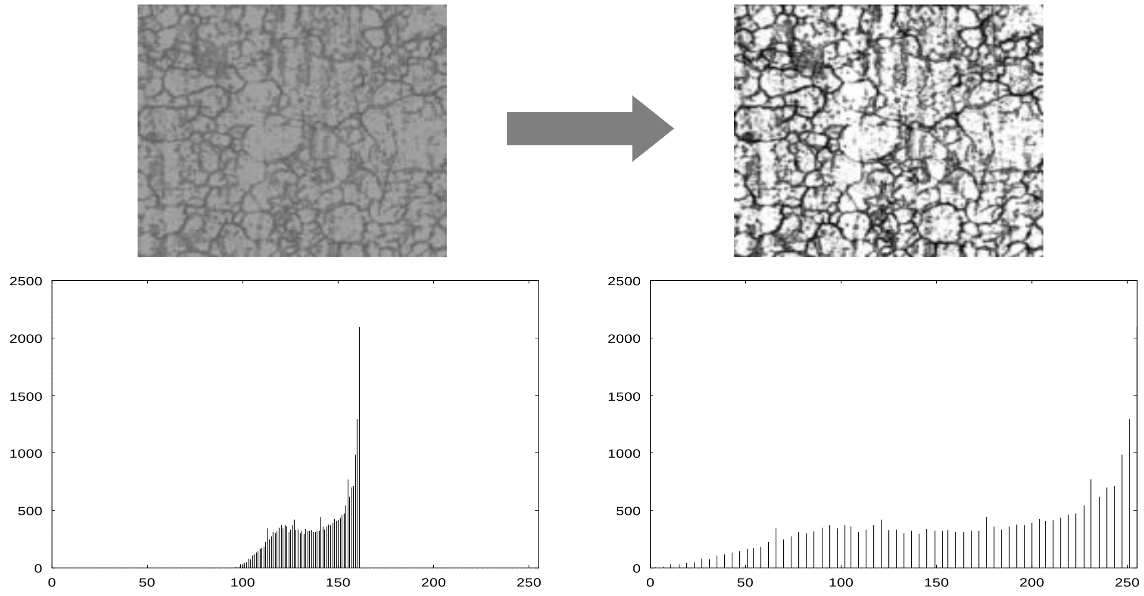
Negative image

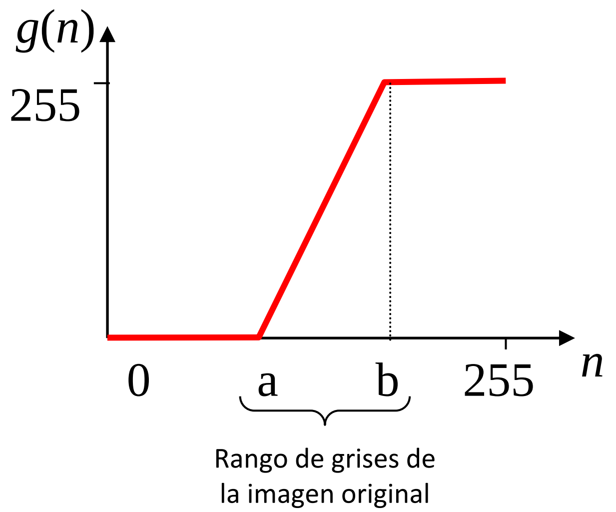


Contrast maximization

- We apply a linear transformation so the values cover all the possible range. Histogram is enlarged.

$$g(n) = 255\left[\frac{I(x,y)-\min(I(x,y))}{\max(I(x,y))-\min(I(x,y))}\right]$$





Non-linear transformations

- These transformations do not modify the values (variable contrast) within the original range of I.

- Two types of transformations:
  - **Logarithmic:** enhancement of dark pixels without saturating bright pixels

$$g(I(x,y)) = \log(I(x,y)+1)$$

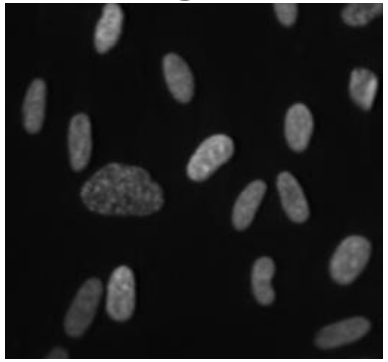
  - **Exponential:** enhancement of bright pixel without saturating dark pixels

$$g(I(x,y)) = e^{I(x,y)}$$

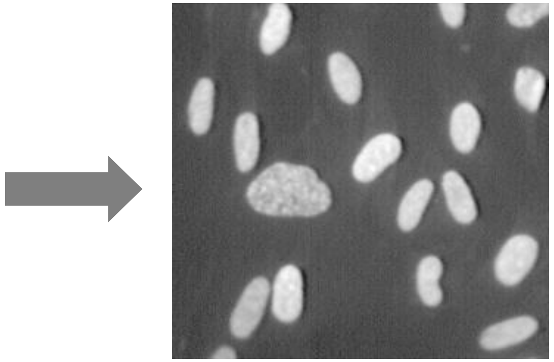
Logarithmic

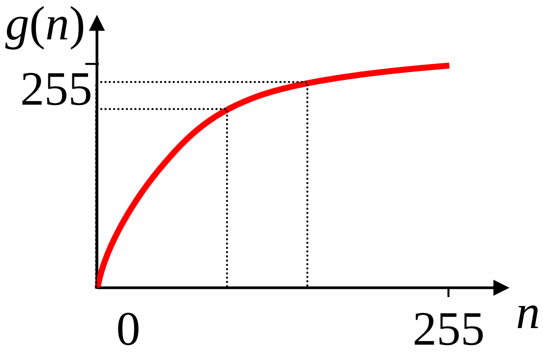
$$g(I(x,y)) = \log(I(x,y)+1)$$

Original

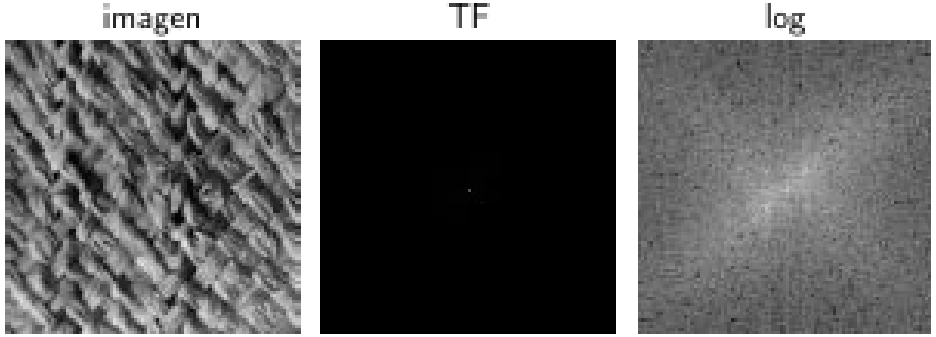


Logarithm



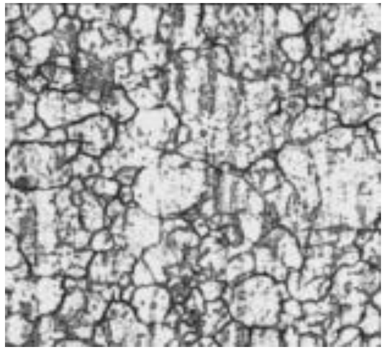


Useful to visualize images with high dynamic range, such as Fourier Transforms

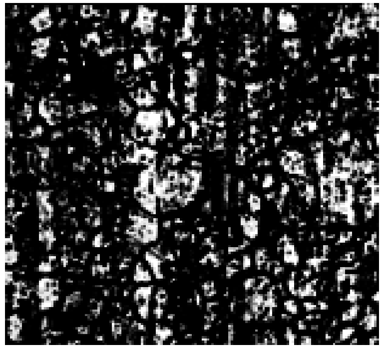


Exponential

Original image



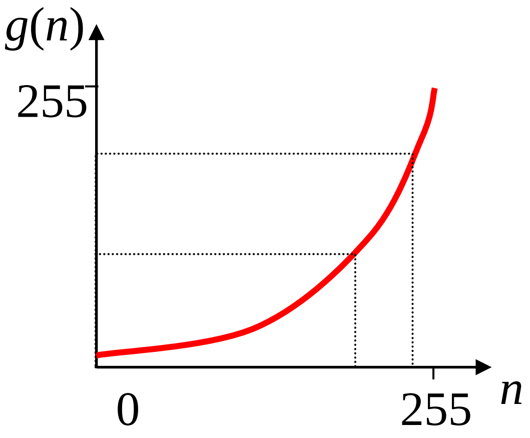
Exponential



$$g(n) = e^{I(x,y)}$$

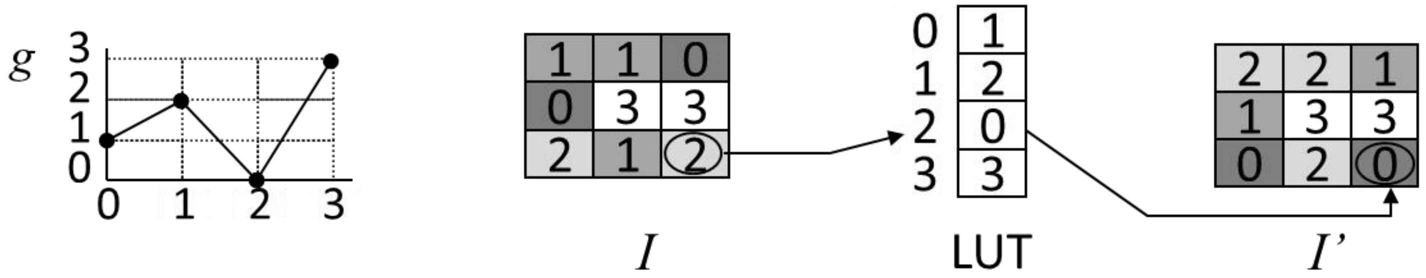
Original image

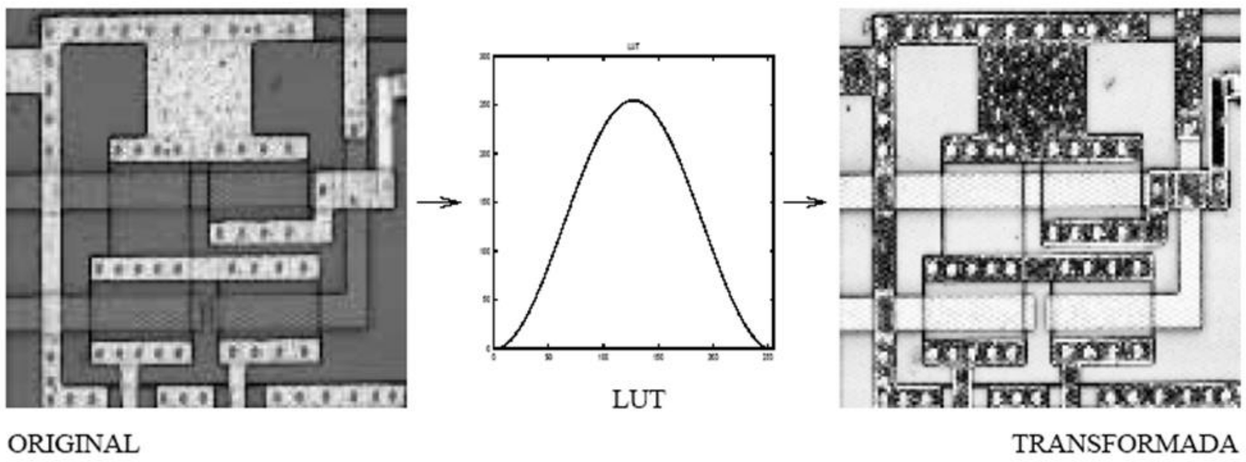
Exponential



L.U.T. (look-up table)

- When we work with discrete and finite signal, g transformations can be applied as an indexation vector (LUT).





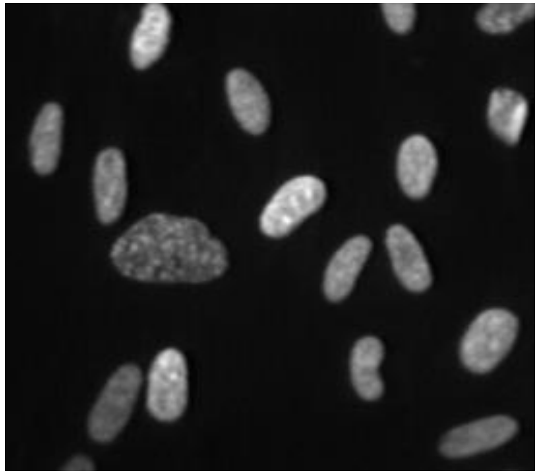
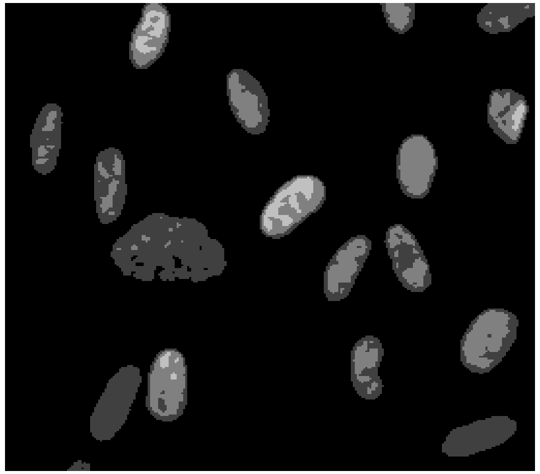
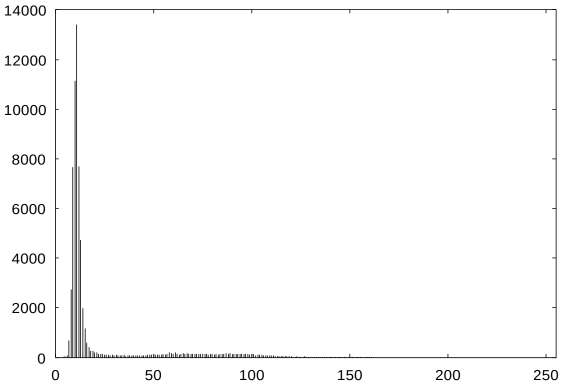
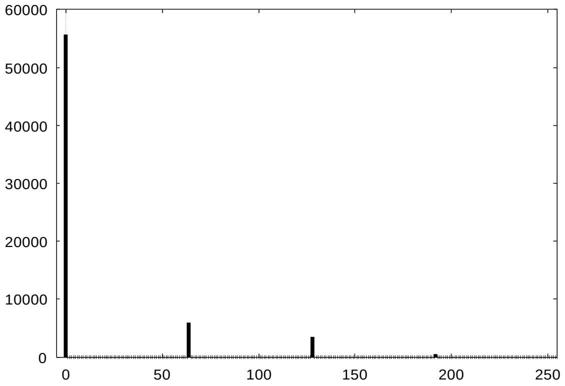
L.U.T. (look-up table)

- LUT defines a vector whose values indicate the expected output for a given input value.
- Pixel values of the original input image index the new values.

Logic operations

- AND, OR, NOT y XOR
- Bit displacement (left and right)

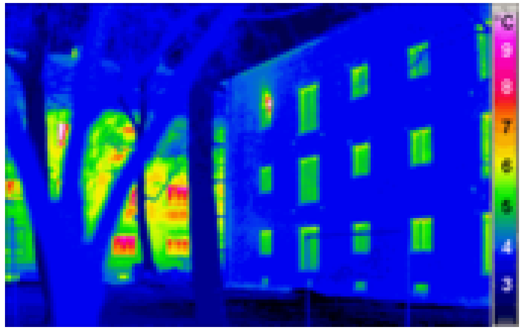
Logic operations

Original image	**AND** with 11000000
	
 	

Requantization with 4 levels of intensity

Pseudocolor and fake color

- Pseudocolor: color visualization of a greyscale image. Each grey level is assigned a color
- Fake color: modification of color channels so the resulting color enhances a particular aspect of the image. Typical object colors are altered, attracting observer's attention



Equalization

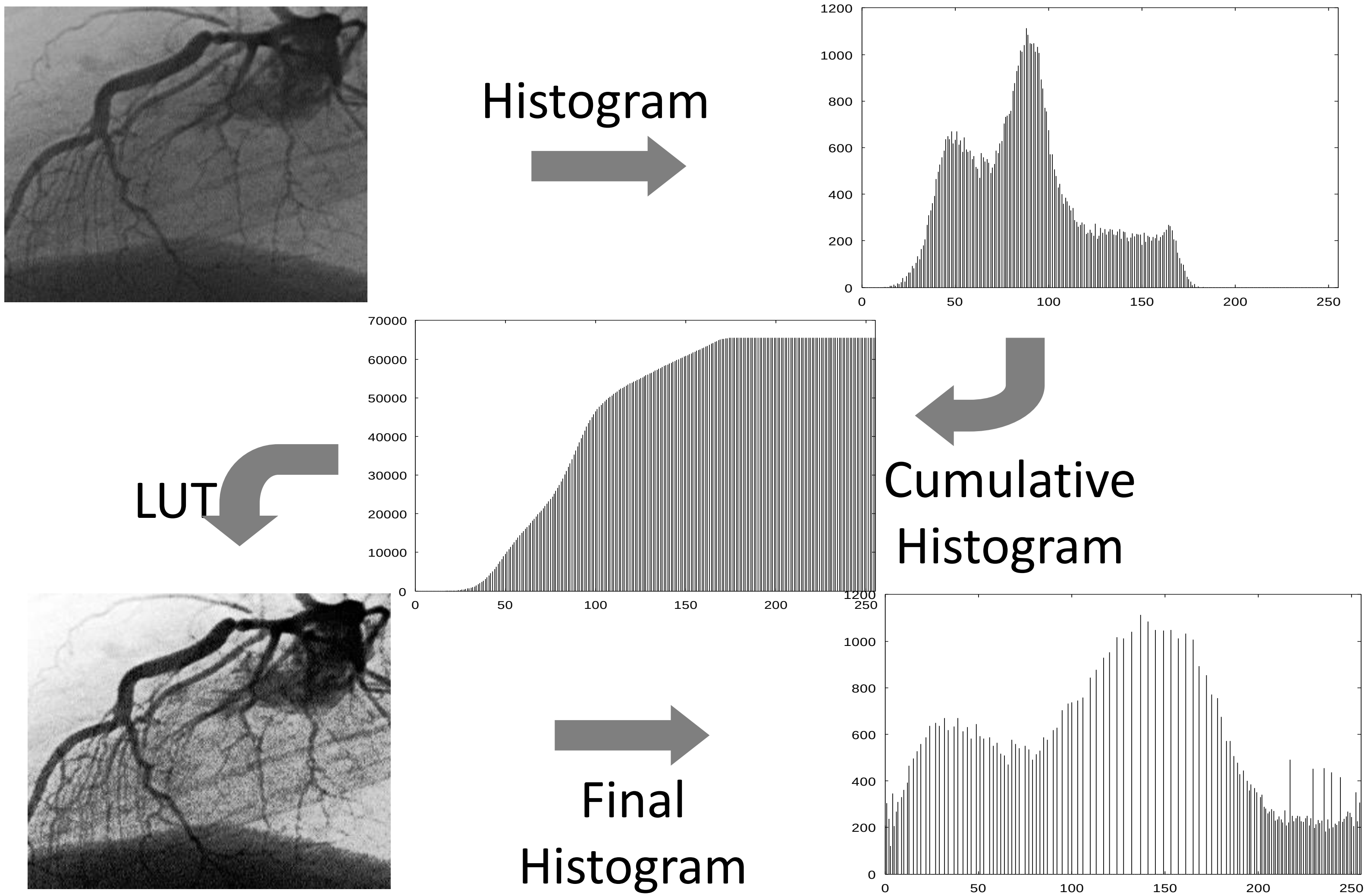
- Process in which every grey level has the same apparition frequency (plain histogram)

- We can achieve this by using a LUT which adjusts the cumulative histogram from the input image to the actual level range that we have

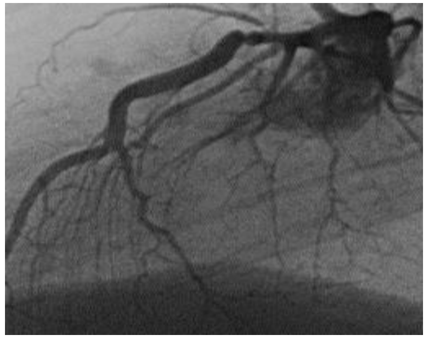
$$g(n) = \frac{\#levels}{\#\,pix} \sum_{i=1}^{n} H[i]$$

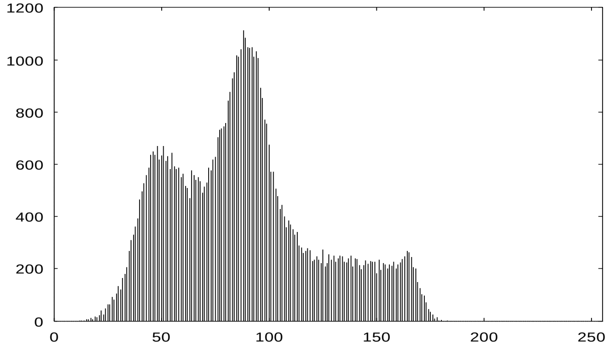
- Variations:
  - Local equalization: the same operation is applied in a window centered in each pixel
  - Logarithmic equalization: the cumulative histogram is powered by a certain constant, changing the shape of the final histogram

Equalization

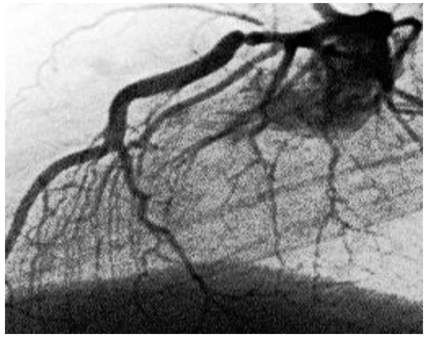


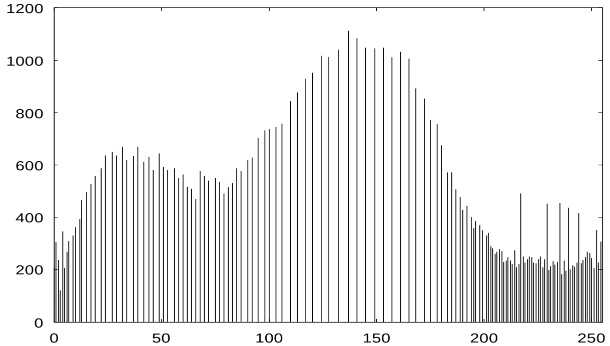
Equalization



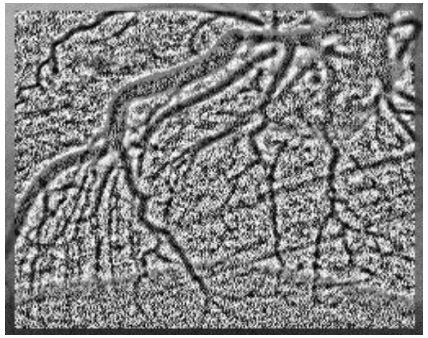


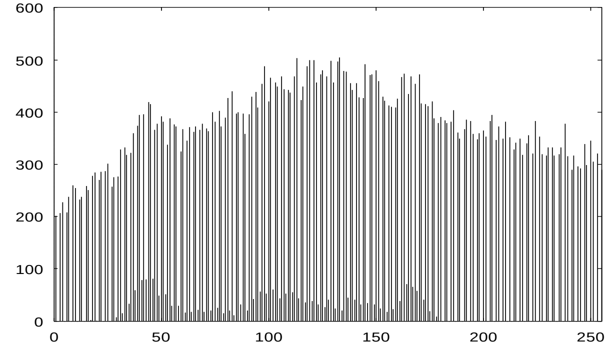
Global equalization





Local equalization using a 13x13 window





Arithmetic transformations

- Result of applying, point by point, the same operator to two images

$$I'(x,y) = I_1(x,y) \circ I_2(x,y)$$

- Example of potential operators
  - Sum
  - Subtraction
  - Product
  - Division
  - AND, OR, XOR

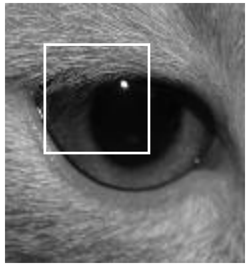
Sum

- Used to reduce acquisition noise. If the scene is stationary, we can make a mean of N images from the same object

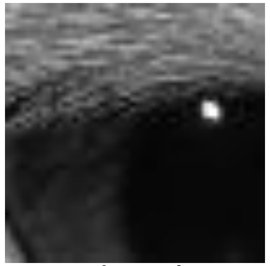
- If noise is:
  - Additive → result = image + noise
  - Not correlated → no pattern is generated
  - With zero mean → values distributed symmetrically around 0

- By making a mean of N images we can increase SNR in a $N^{1/2}$. The greater the SNR, the better the image

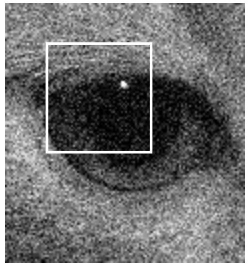
Sum



original

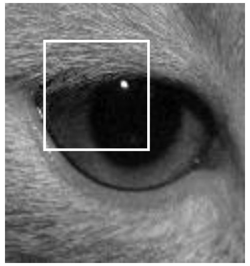


detail

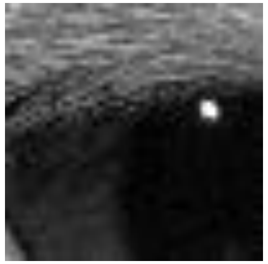


. . . (64 ims) . . .

detail



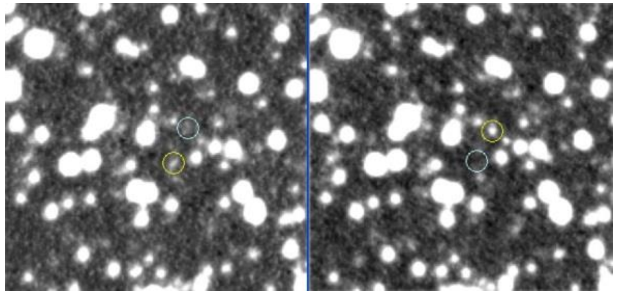
mean

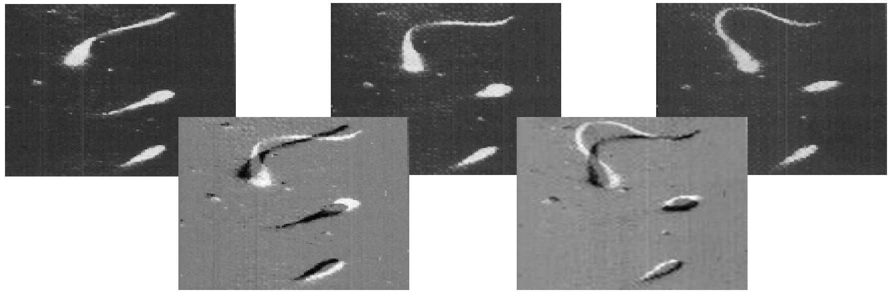


detail

Subtraction

- Background subtraction
  - Movement (or absence of it) detection
  - Remove the part of the image that does not change
  - Reduce information, sequence compression





Product

- Selection of Regions of Interest (ROI): we multiply the image by a mask with 1 in the region of interest and 0 in the rest

- Filtering in Fourier domain → product in frequency is equal to convolution in spatial domain

$$f * g = F^{-1}\{F\{f\} \cdot F\{g\}\}$$

- Decrease of border effects if we multiply the image by another one whose values decrease in the peripheral part

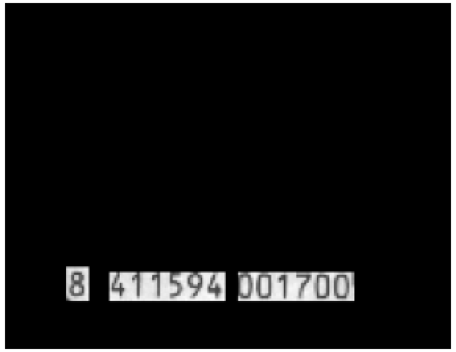
Product

- Regions of interest

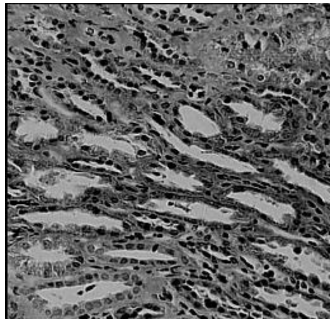
imagen

máscara

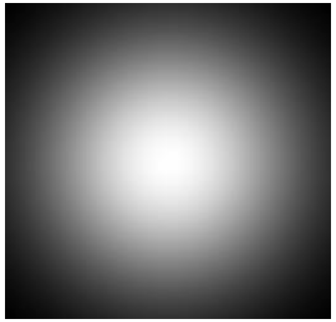
ROI's



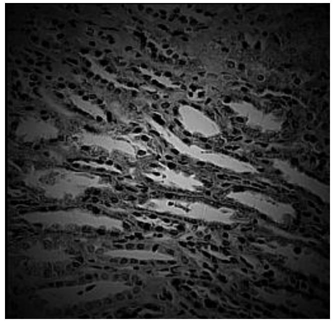
- Decrease of border effects.



original



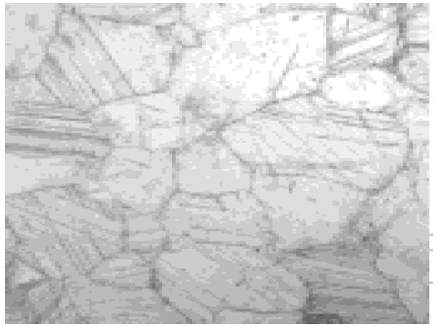
mask



result

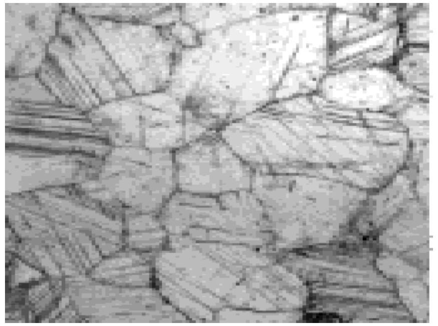
Division

- If we model a given process as the result of the producto of two factors (images), we can remove one with the division
- Example: microscopy image (transmitted light)
  - t_s sample transmision, t_b non homogeneous illumination that we want to remove, I_i constant illumination intensity.
  - I_s, I_b sample and background image.
  - $I_s=I_i\, t_b\, t_s$, $I_b=I_i\, t_b$.
  - We are interested in: $t_s=I_s/I_b$



sample

background



correction

Fundamentals of Computer Vision

Unit 3: Fundamentals of Image Processing

Jorge Bernal (Jorge.Bernal@uab.cat)