Mestrado em
Engenharia Informática

Image Segmentation

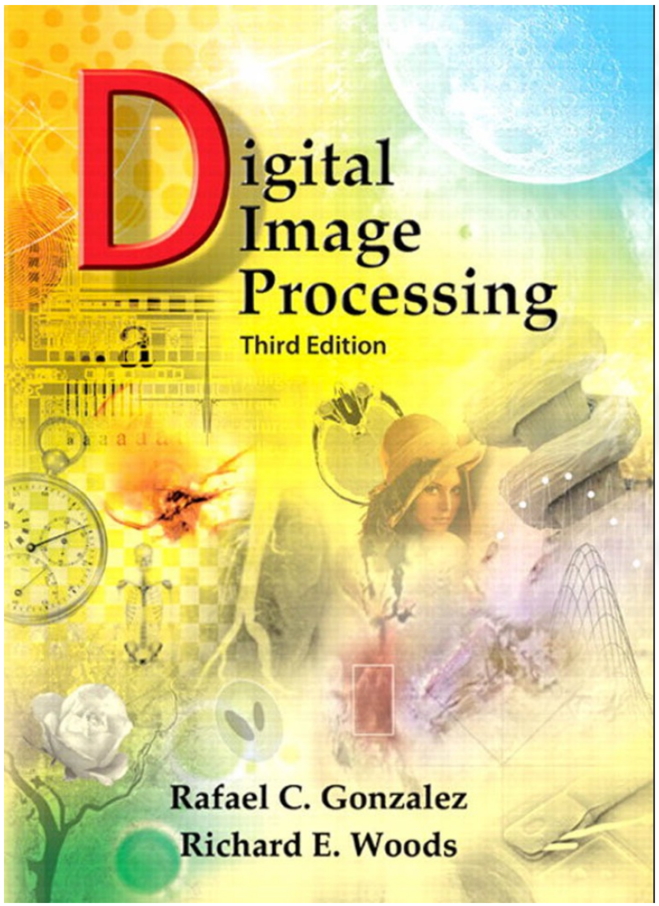
Visão por Computador

Luís Paulo Peixoto dos Santos

Mestrado em

Engª Informática

Bibliography



Pearson Prentice Hall
Pearson Education, Inc.
Upper Saddle River, New Jersey 07458

ISBN 0-13-168728-x

https://www.imageprocessingplace.com/

Capítulo 10

What is segmentation?

The process of partitioning the set of pixels R in an image into subregions $R_1, \ldots, R_n$, such that:

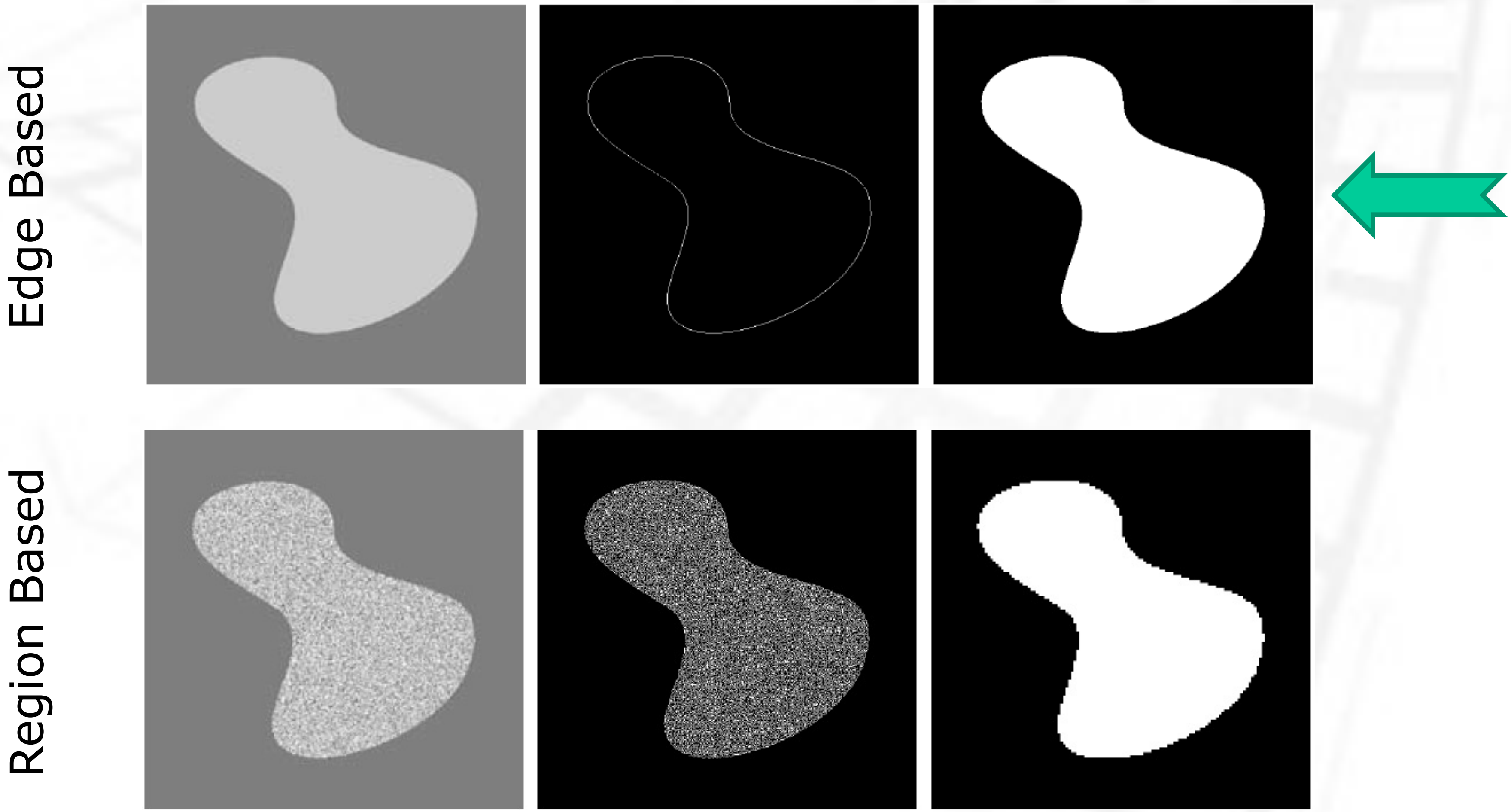
1. $\bigcup_{r=1}^{n} R_r = R$
2. R_r is a connected set
3. $R_i \cap R_j = \emptyset, \forall i, j = 1 \ldots n, i \neq j$
4. $Q(R_i) = TRUE$
5. $Q(R_i \cup R_j) = FALSE$ for any **adjacent** regions R_i , R_j

where $Q(R_i)$ is some logical predicate over the pixels in region R_i

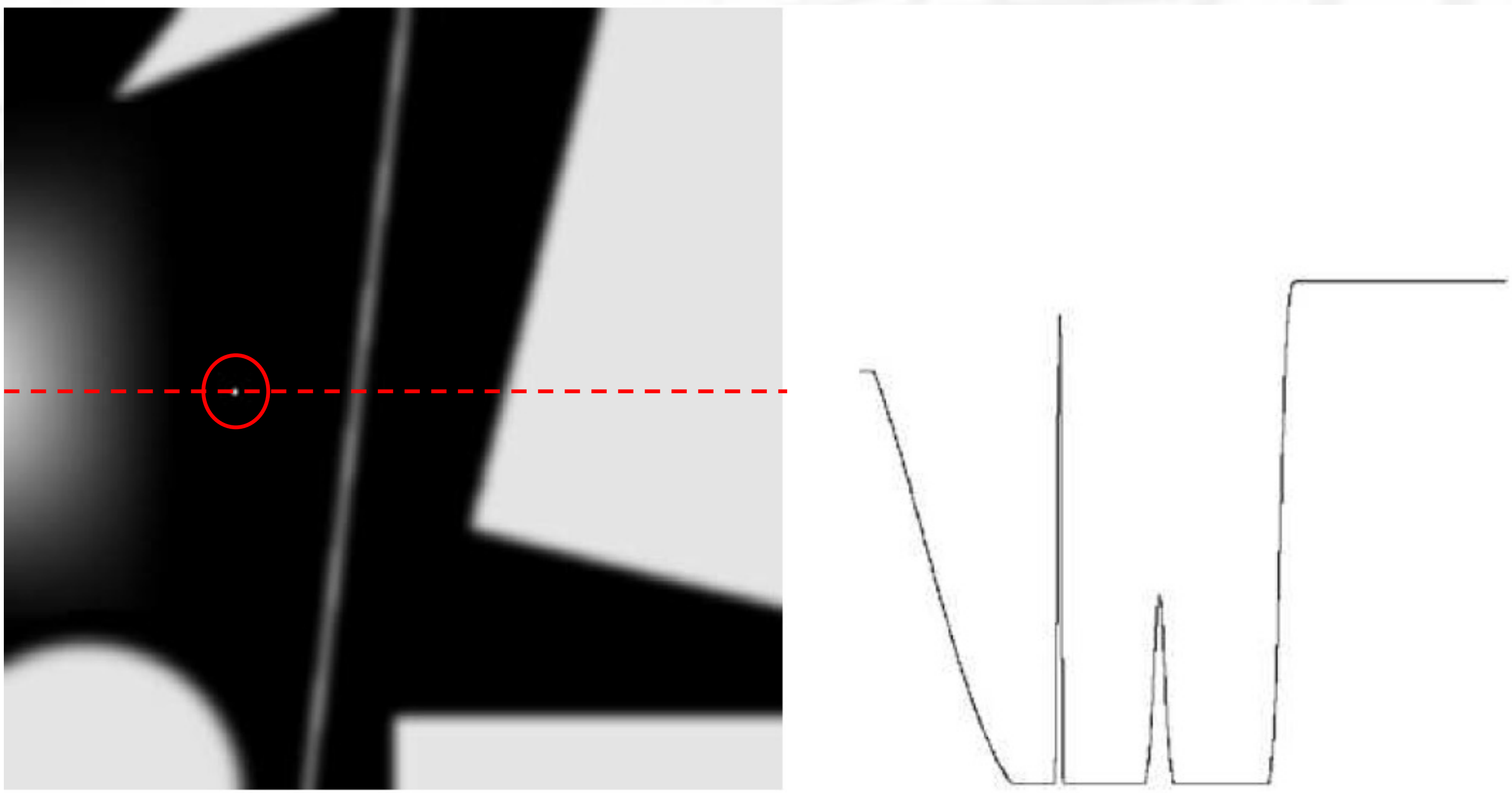
Image Segmentation

Mestrado em
Engª Informática

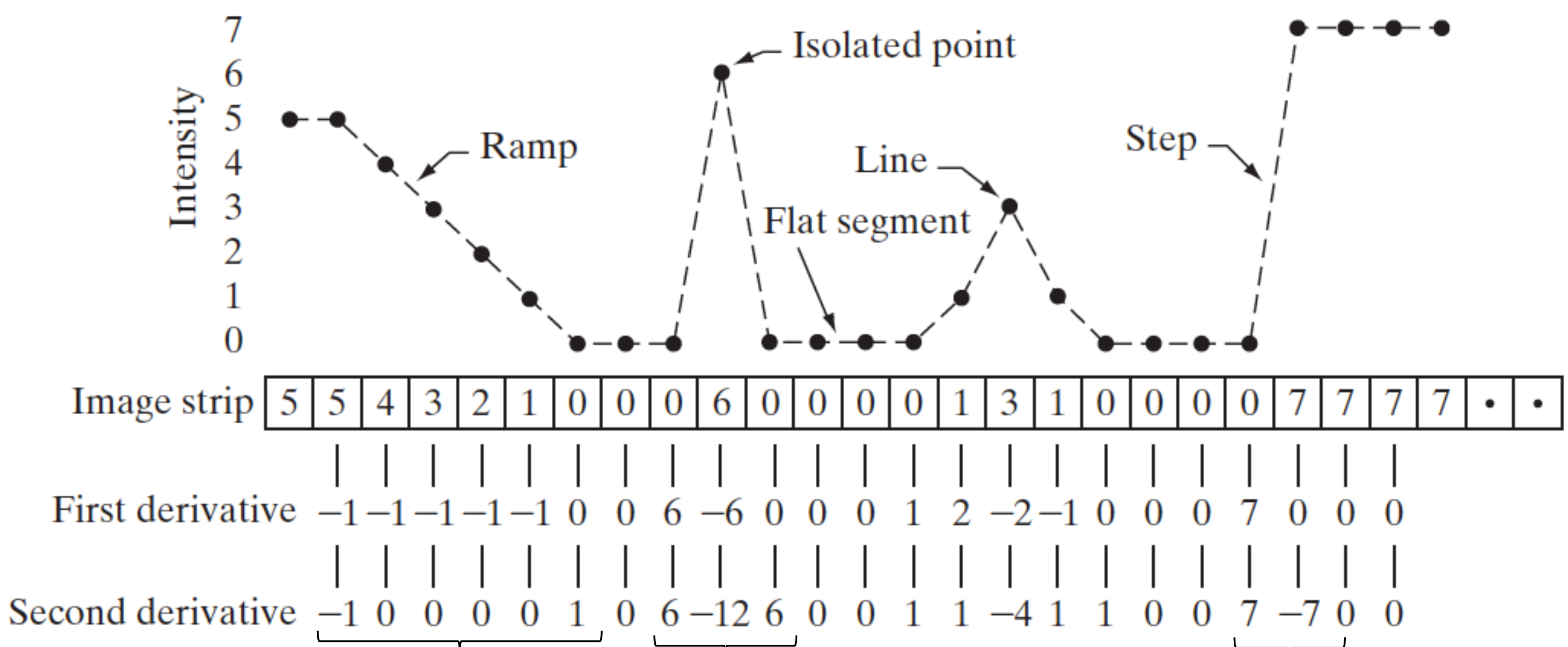
Image Segmentation



Edge Detection - Fundamentals



Edge Detection - Fundamentals



Ramp
$\nabla^2 f$ – zero crossing

Point
$|\nabla^2 f| \gg |\nabla f|$

Step
$\nabla^2 f$ – zero crossing

Laplacian Operator

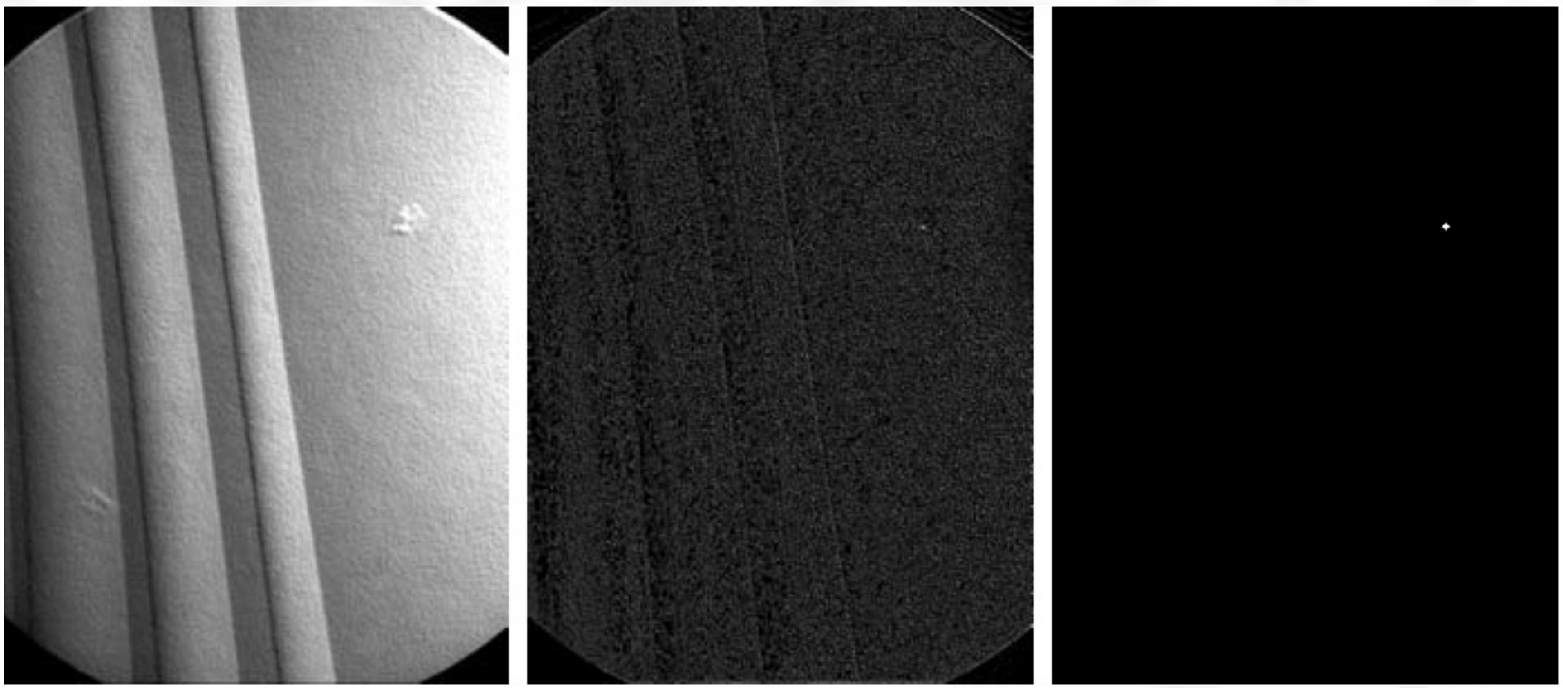
$$\nabla^2 f(x,y) = 8 * f(x,y) - \sum_{s=-a, s\neq 0}^{a} \sum_{t=-b, t\neq 0}^{b} f(x+s, y+t).$$

$$\nabla^2 f(x,y) = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Care must be taken with the sign and magnitude of $\nabla^2 f(x,y)$ if it is to be displayed as an image

Point Detection

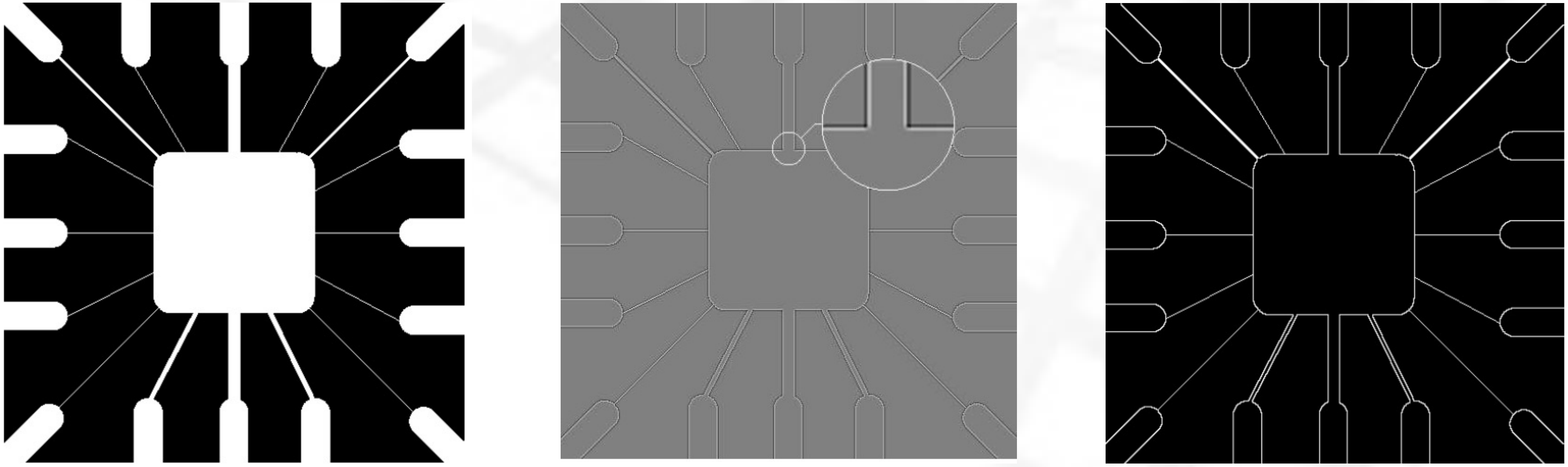
$$mask(x,y) = \begin{cases} 1 & \leftarrow |\nabla^2 f(x,y)| \geq T \\ 0 & \leftarrow \neg \end{cases}$$



Line Detection

$$mask(x,y) = \begin{cases} 1 & \leftarrow \nabla^2 f(x,y) \geq T \\ 0 & \leftarrow \neg \end{cases}$$

Only positive values to avoid doubling the lines' thickness.



Line Detection

- The previously seen Laplacian is isotropic (response independent of the lines' directions)
- Different anisotropic filters can be used to detect lines with given orientations

−1	−1	−1
2	2	2
−1	−1	−1

Horizontal

2	−1	−1
−1	2	−1
−1	−1	2

+45°

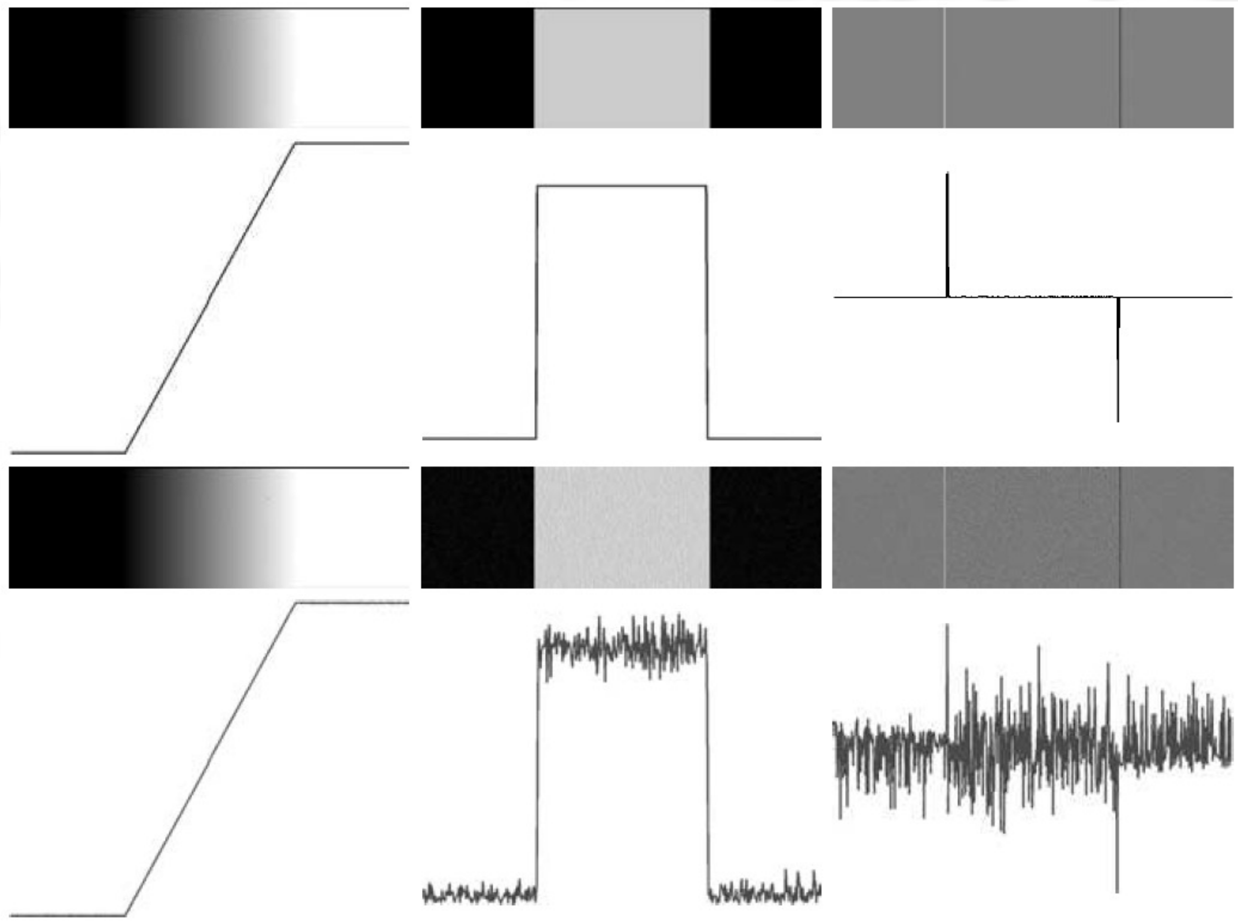
−1	2	−1
−1	2	−1
−1	2	−1

Vertical

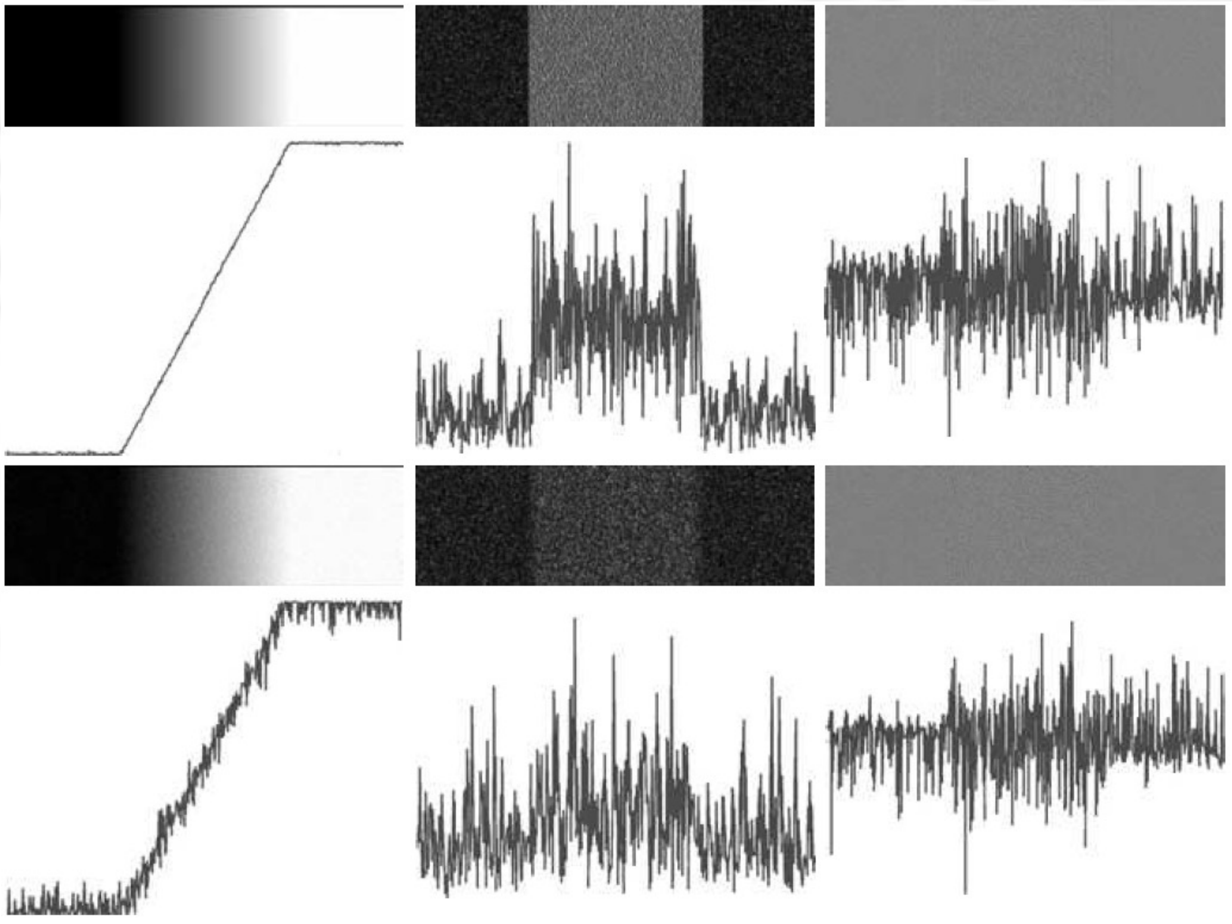
−1	−1	2
−1	2	−1
2	−1	−1

−45°

Impact of Noise (I)



Impact of Noise (II)



Gradient Operators

g_x | g_y

−1	−1	−1
0	0	0
1	1	1

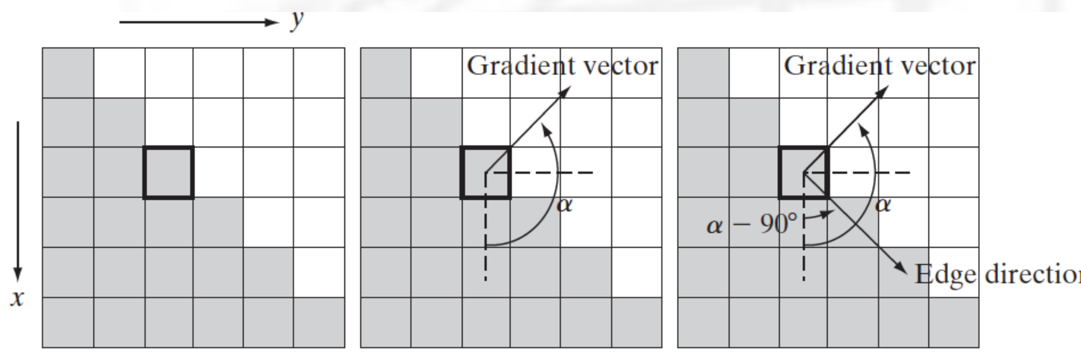
−1	0	1
−1	0	1
−1	0	1

Prewitt

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

Sobel



$$M(x,y) = |\nabla f(x,y)| = \sqrt{{g_x}^2 + {g_y}^2} \approx |g_x| + |g_y|$$

$$\alpha(x,y) = tan^{-1}\left(\frac{g_y}{g_x}\right)$$

Gradient Operators

- Application to raw image sensitive to noise and detail

Gradient Operators – Smoothing

- Smoothing diminishes sensitivity to noise and detail

5x5 average filtered

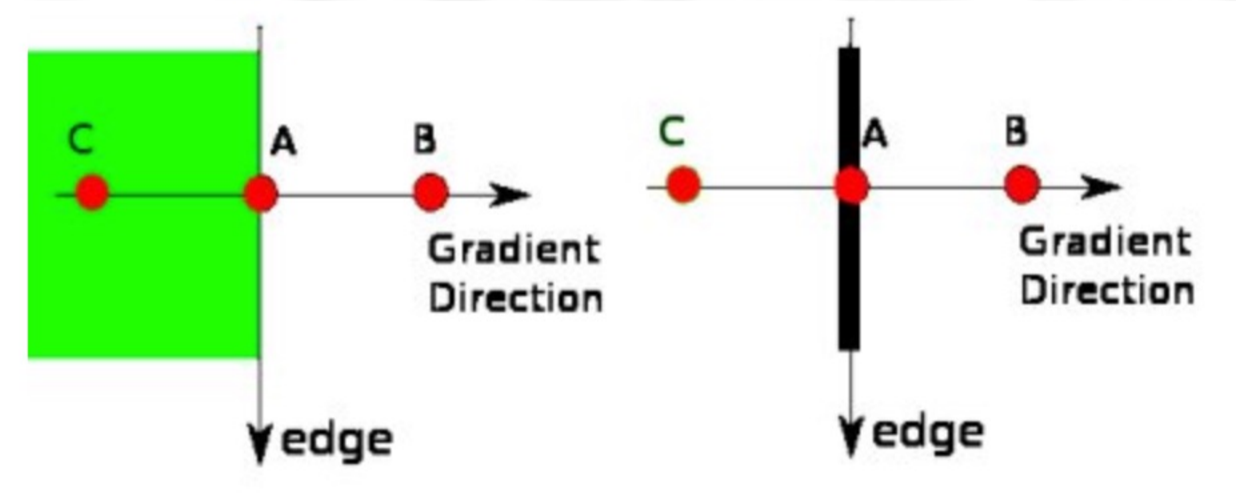
Gradient Operators – Thresholding

$$g(x, y) > 0.3 \max|\nabla f(x, y)|$$

smoothed
+
thresholded

Canny edge detection

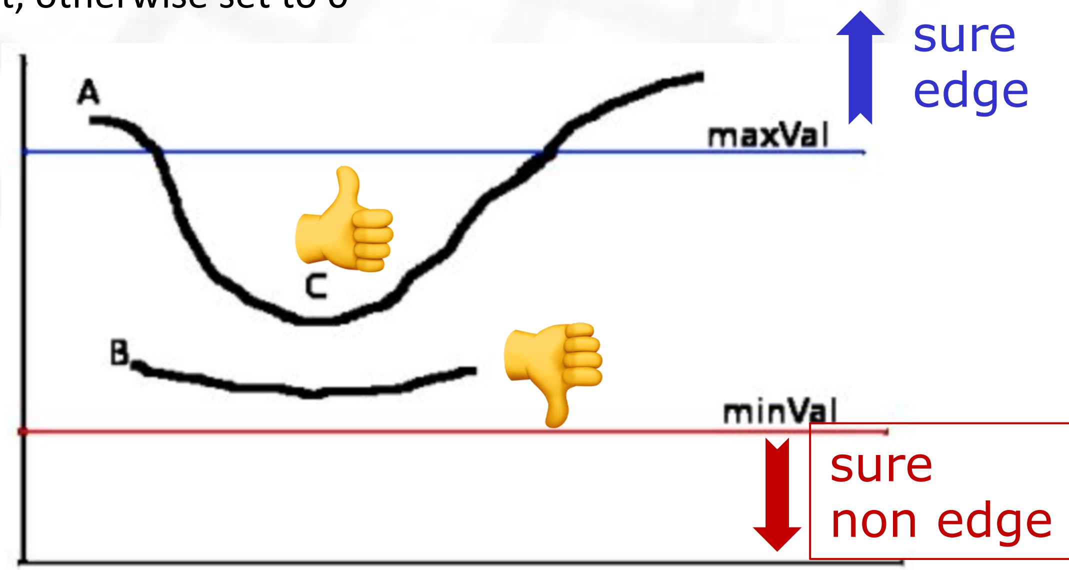
1. Noise reduction by Gaussian filtering
2. Compute the magnitude and direction of the gradient
3. Apply nonmaxima suppression to the gradient magnitude image:
    1. For each pixel g_p in the gradient magnitude map:
        1. Compare g_p with the 2 pixels in the negative and positive gradient directions
        2. If g_p is larger than both keep it, otherwise set to 0



Canny edge detection

1. Noise reduction by Gaussian filtering
2. Compute the magnitude and direction of the gradient
3. Apply nonmaxima suppression to the gradient magnitude image:
    1. For each pixel g_p in the gradient magnitude map:
        1. Compare g_p with the 2 pixels in the negative and positive gradient directions
        2. If g_p is larger than both keep it, otherwise set to 0
4. Hysteresis Thresholding

which pixels in the magnitude gradient are edges based on their connectivity to strong ("sure") edges



Canny edge detection

The original image

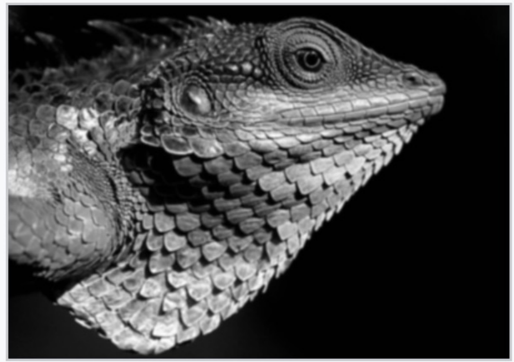
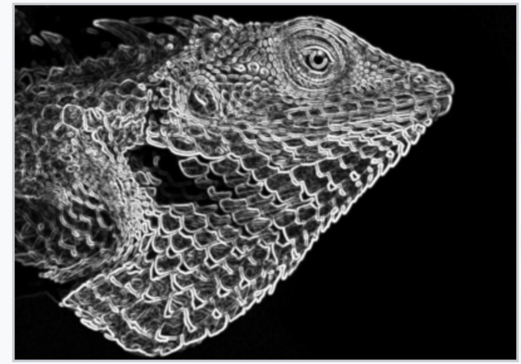
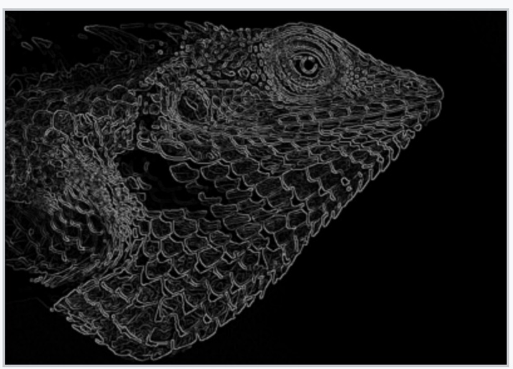


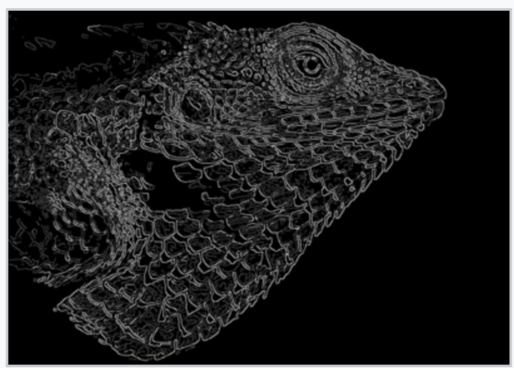
Image has been reduced to grayscale, and a 5x5 Gaussian filter with σ=1.4 has been applied



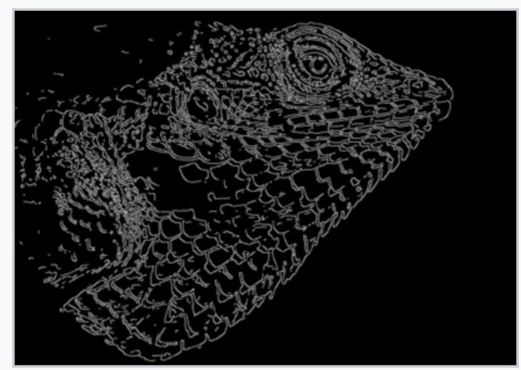
The intensity gradient of the previous image. The edges of the image have been handled by replicating.



Non-maximum suppression applied to the previous image.



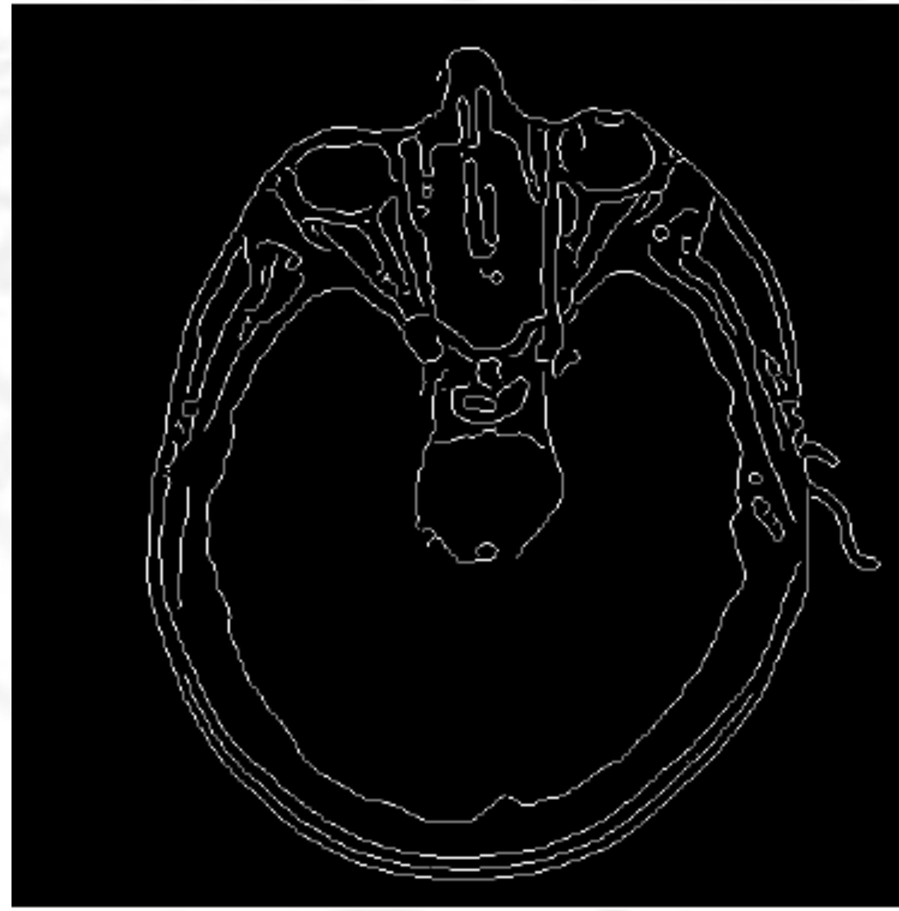
Double thresholding applied to the previous image. Weak pixels are



Hysteresis applied to the previous image

https://en.wikipedia.org/wiki/Canny_edge_detector

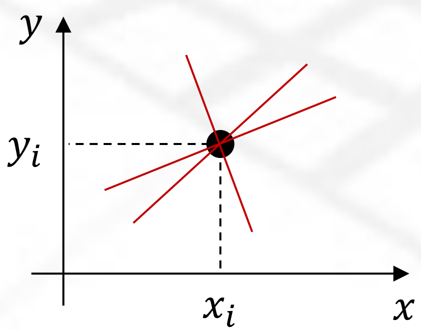
Canny edge detection

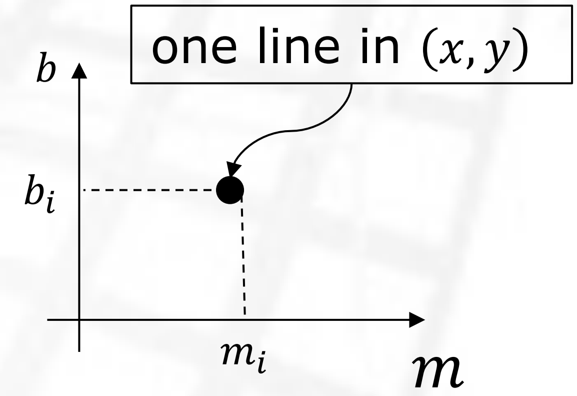


Edge Linking – Hough Transform

- A point in cartesian space $p_i = (x_i, y_i)$ belongs to an infinite number of lines to which the following expression holds:

$$y_i = m * x_i + b, \forall m, b \in \mathbb{R}$$





- A point in parameter space $l_i = (m_i, b_i)$ represents a line with an infinite number of points to which the following expression holds:

$$y = m_i * x + b_i, \forall x, y \in \mathbb{R}$$

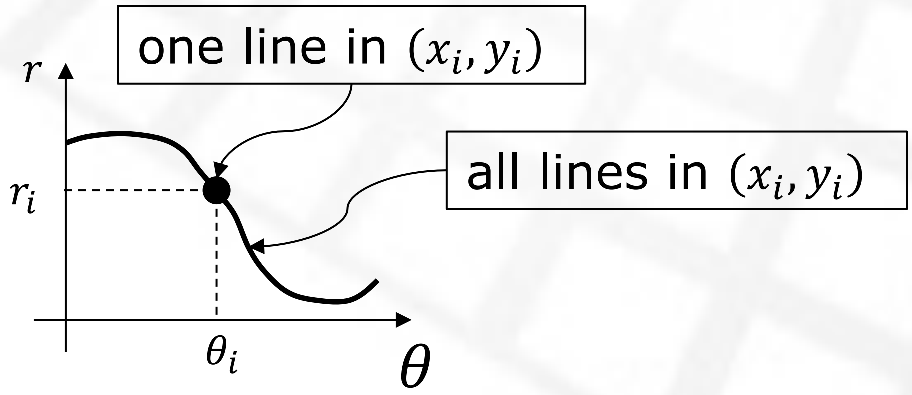
Edge Linking – Hough Transform

- Since m approaches ∞ as lines approach the vertical, polar coordinates are used instead:

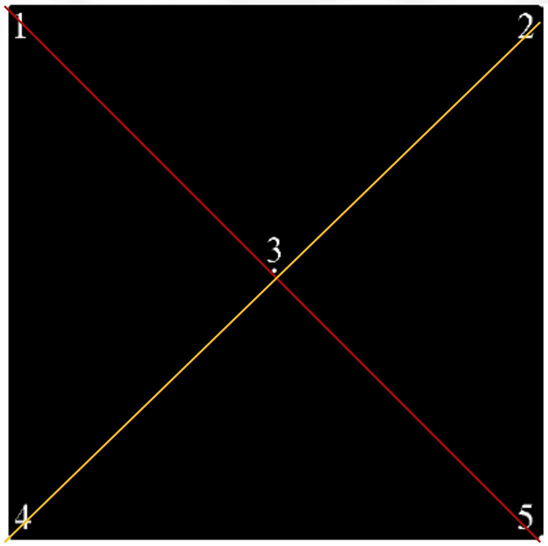
$$r = x_i * \cos\theta + y_i * \sin\theta$$

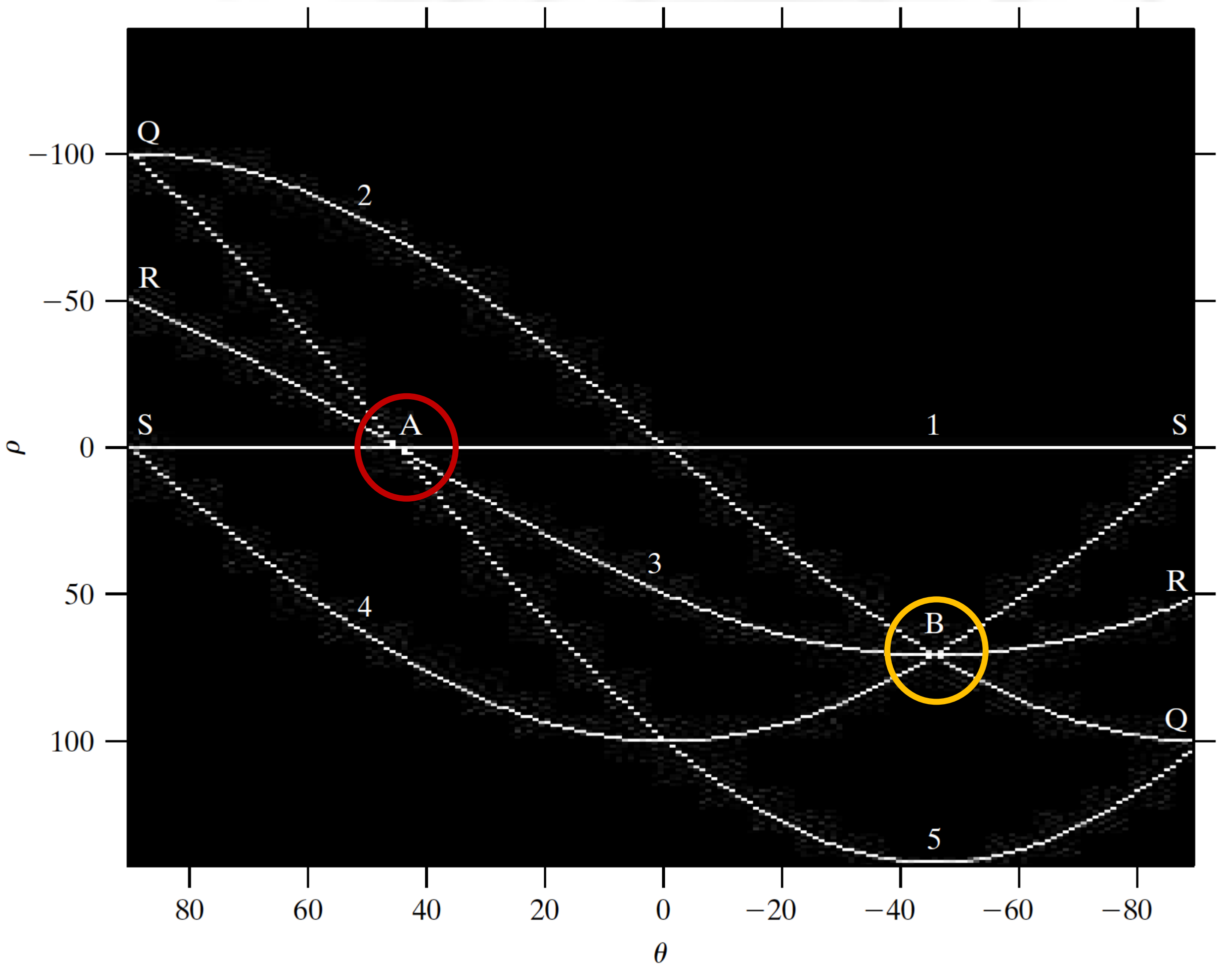
- where the parameters (r, θ) represent all lines passing in (x_i, y_i)

- For an image (r, θ) are discretized and positive ($\theta \in [0,2\pi[$)

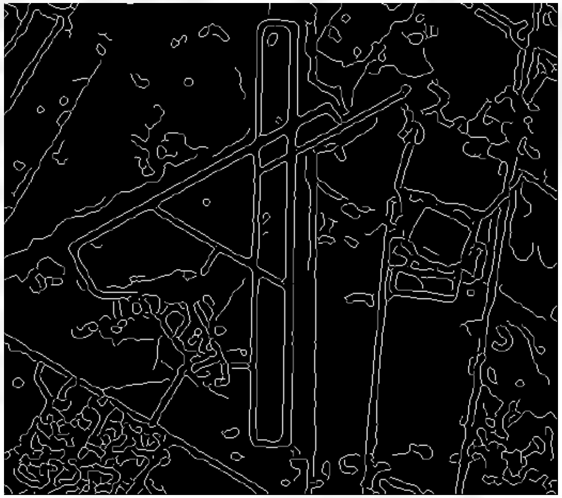


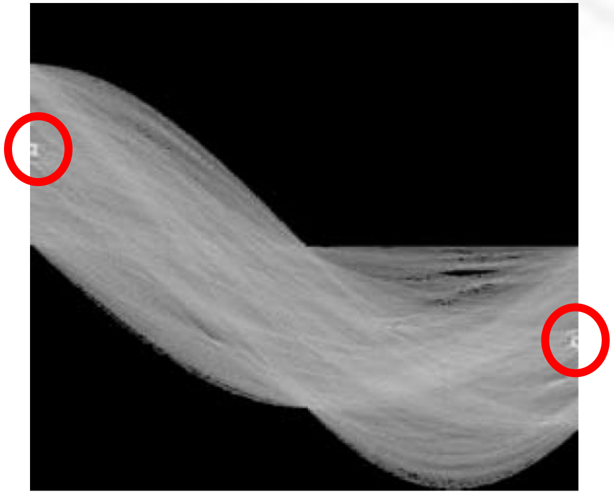
Edge Linking – Hough Transform





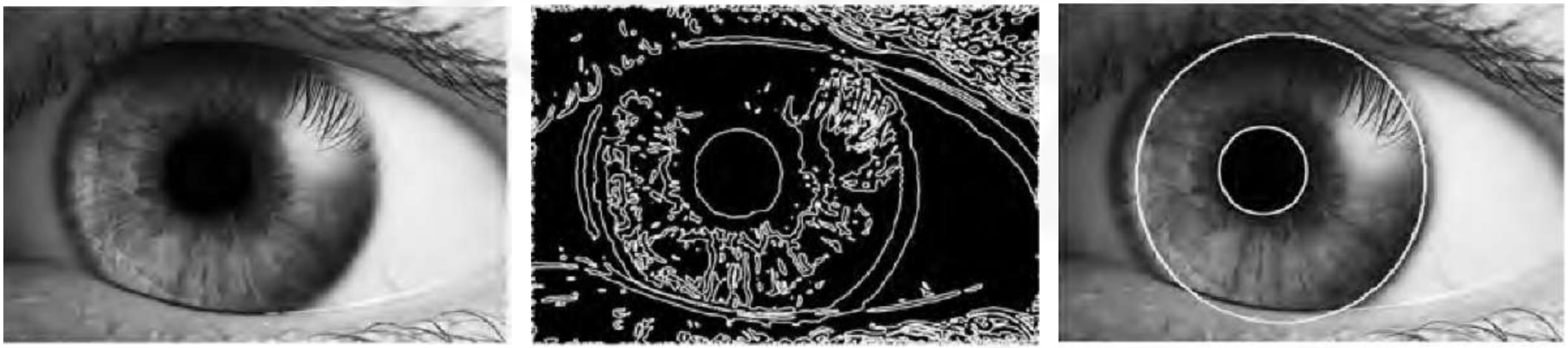
Edge Linking – Hough Transform





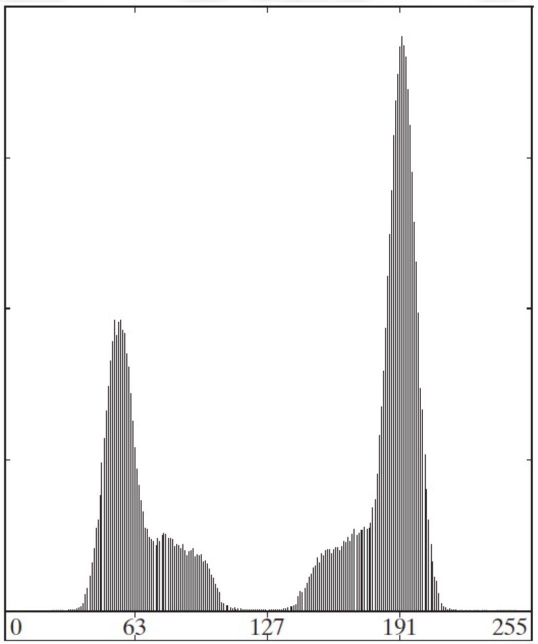
Edge Linking – Hough Transform

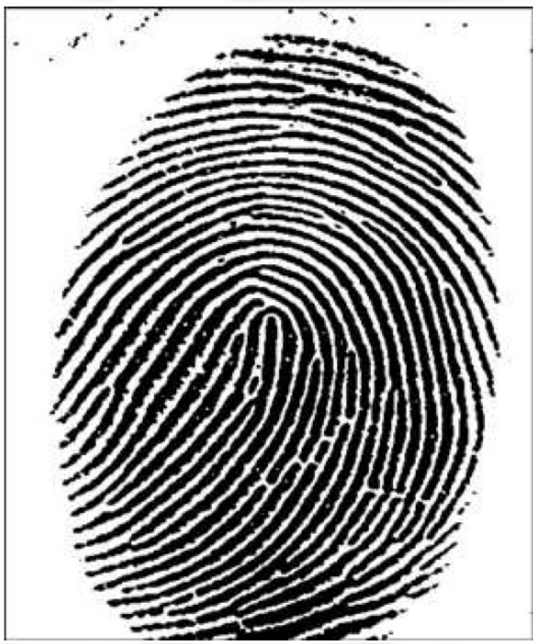
- The Hough transform can be applied to other curves described in closed analytical form.
- For example the circle: $(x - x_c)^2 + (y - y_c)^2 = r^2$
- The time complexity of the algorithm increases with the number of parameters



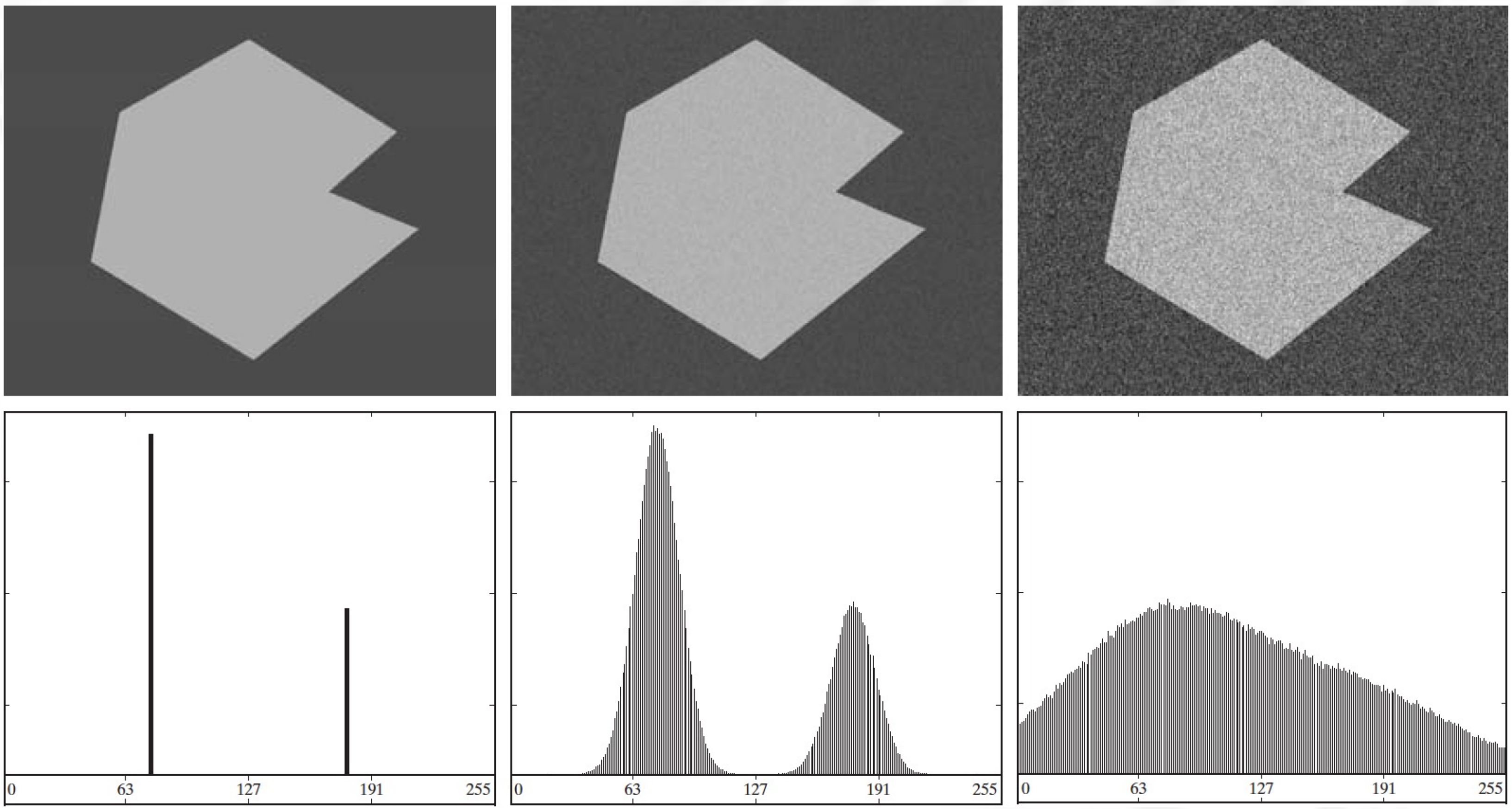
Thresholding

$$g(x,y) = \begin{cases} 1 & \leftarrow f(x,y) \geq T \\ 0 & \leftarrow \neg \end{cases}$$

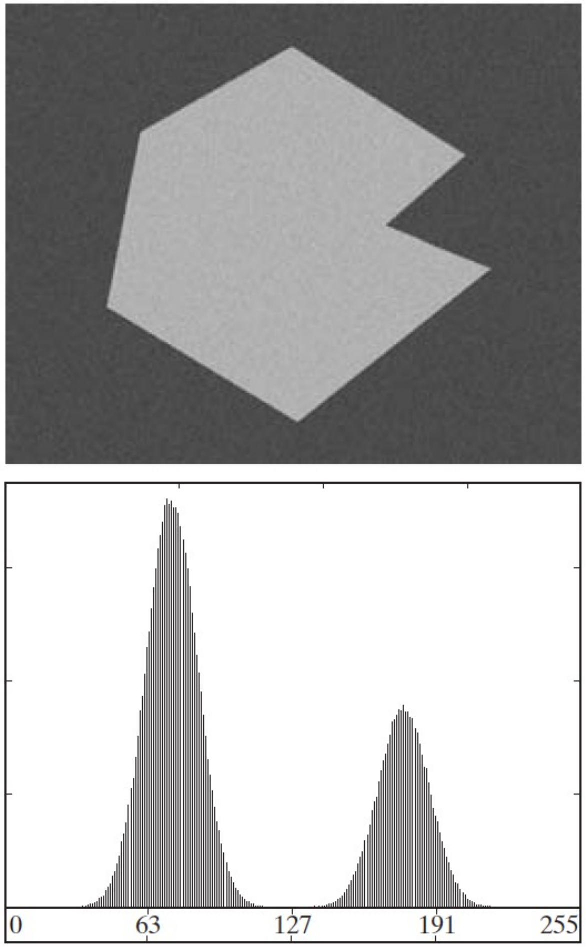


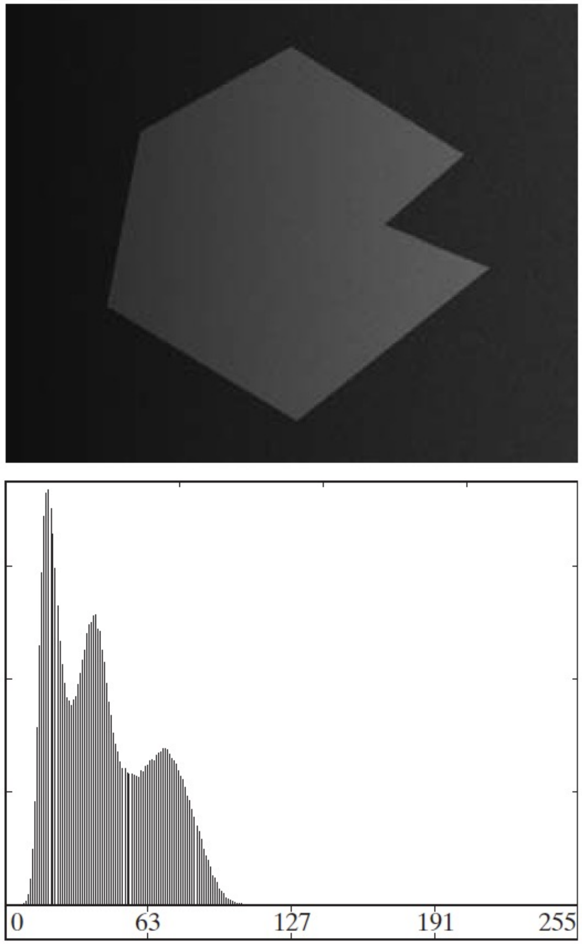


Thresholding: noise



Thresholding: illumination





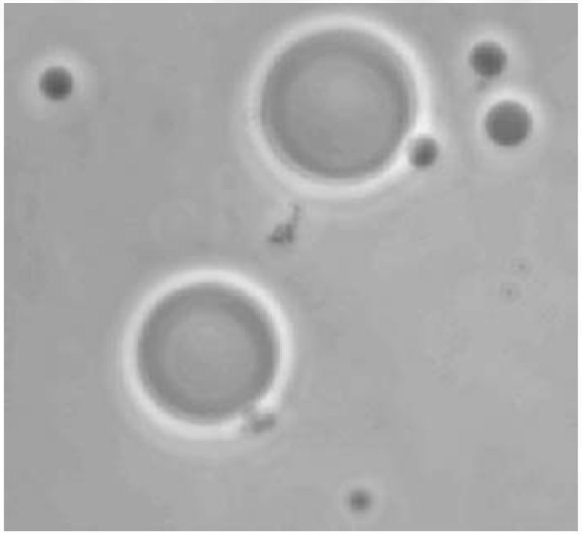
Otsu Optimal Global Thresholding

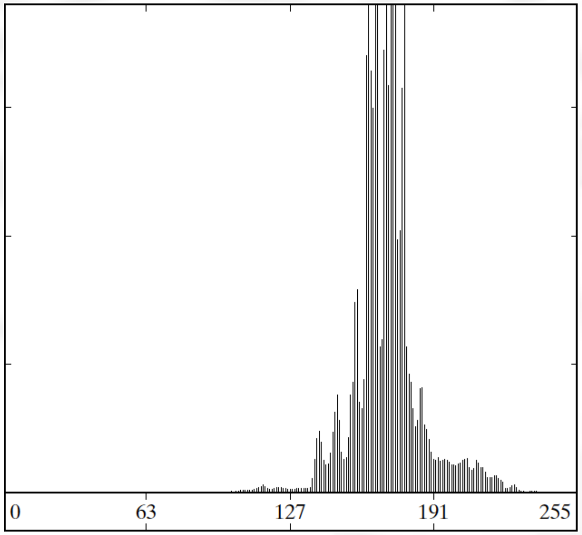
- Consider the normalized histogram with $p_i = {}^{n_i}/_{(M*N)}$, i $= 0, \dots, L-1$, for an image with $M * N$ pixels and L illumination levels
- A threshold k divides the pixels into 2 classes:
  - C_1 containing the pixels with values in $\{0, \cdots, k\}$
  - C_2 containing the pixels with values in $\{k+1, \cdots, L-1\}$
- Let $P_1(k)$ be the probability that a pixel belongs to C_1
- Let $m(k) = \sum_{i=0}^{k} i p_i$ be the average intensity of C_1
- Let $m_G = \sum_{i=0}^{L-1} i p_i$ be the average global intensity (whole image)

- Then the optimal threshold is given by

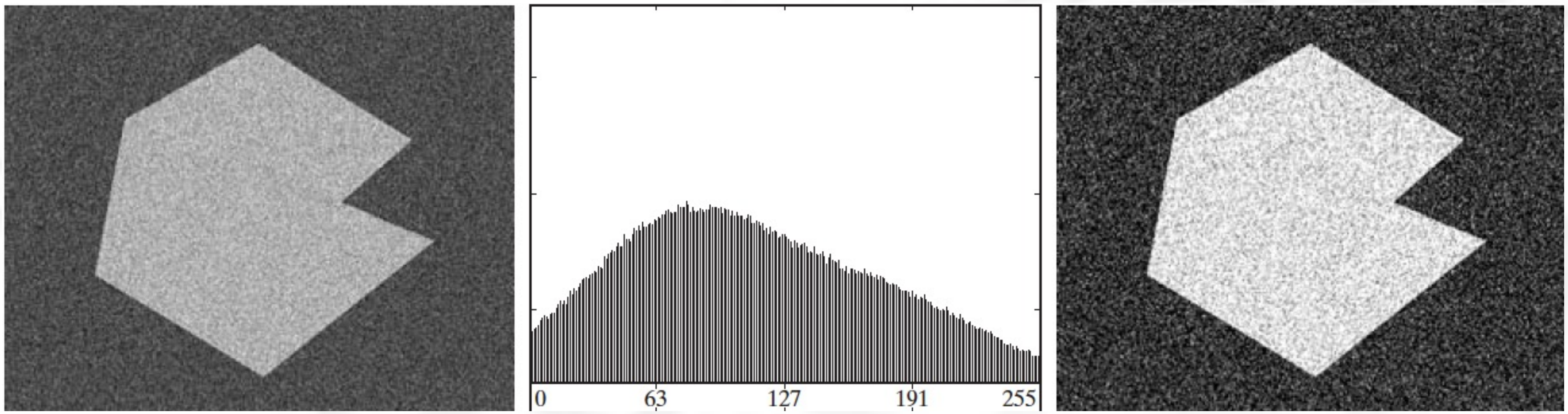
$$k = \underset{0 \le k < L}{\operatorname{argmax}} \left(\frac{[m_G P_1(k) - m(k)]^2}{P_1(k)[1 - P_1(k)]} \right)$$

Otsu Optimal Global Thresholding

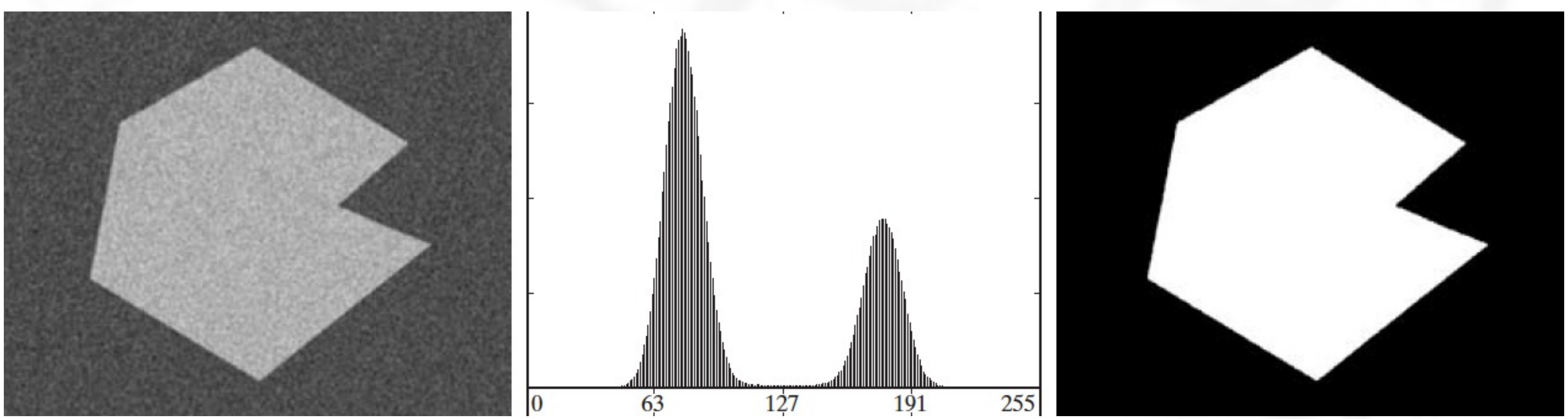




Thresholding: smoothing noise



5x5 smoothing filter



Thresholding: Edges

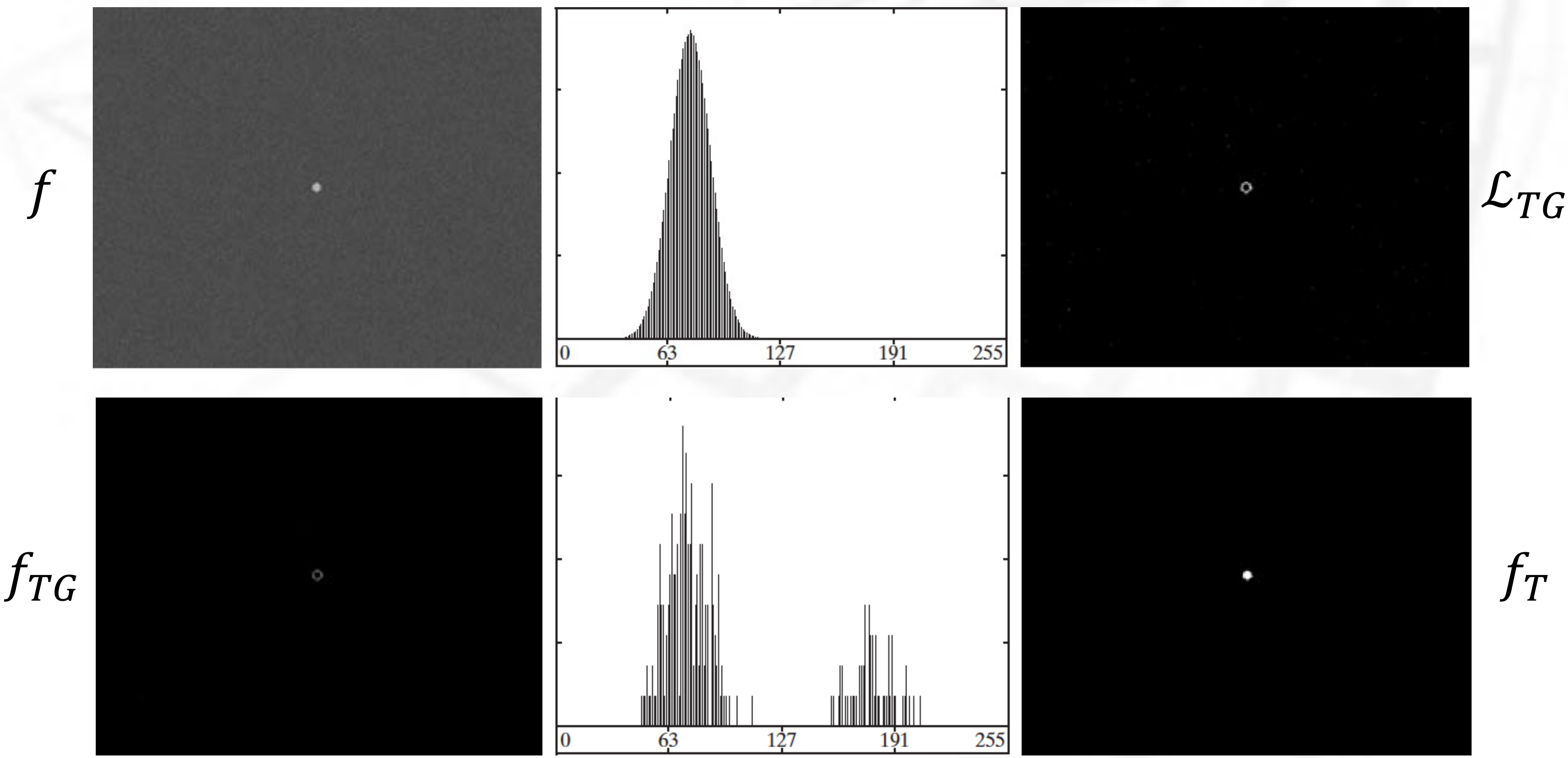
- Select the threshold using the edges' pixels instead of all pixels:

1. $\mathcal{L}(x,y) = |\nabla^2 f(x,y)|$
2. $\mathcal{L}_{TG}(x,y) = \begin{cases} 0 & \leftarrow \mathcal{L}(x,y) < TG \\ 1 & \leftarrow \neg \end{cases}$ `// mask strong edges`
3. $f_{TG}(x,y) = \begin{cases} 0 & \leftarrow \mathcal{L}_{TG}(x,y) = 0 \\ f(x,y) & \leftarrow \mathcal{L}_{TG}(x,y) = 1 \end{cases}$ `// select strong edges`
4. $T = Otsu(f_{TG})$
5. $f_T(x,y) = \begin{cases} 0 & \leftarrow f(x,y) < T \\ 1 & \leftarrow \neg \end{cases}$ `// threshold the original image`

Mestrado em

Engª Informática

Thresholding: Edges



Local Thresholding

- Select a per pixel threshold $T(x, y)$

- Most commonly

$$T(x, y) = a * \sigma(x, y) + b * m(x, y)$$

or

$$T(x, y) = a * \sigma(x, y) + b * m_G$$

- m_G - mean image value
- $m(x, y)$ – mean value in a neighbourhood of (x, y)
- $\sigma(x, y)$ – standard deviation in a neighbourhood of (x, y)

Local Thresholding