Chapter 10 Image Segmentation

10.1 Fundamentals

10.2 Point, Line, and Edge Detection

10.3 Thresholding

10.4 Region-Based Segmentation

10.1 Fundamentals

R: entire image – partition into n subregions $\{R_1, R_2, \ldots, R_n\}$

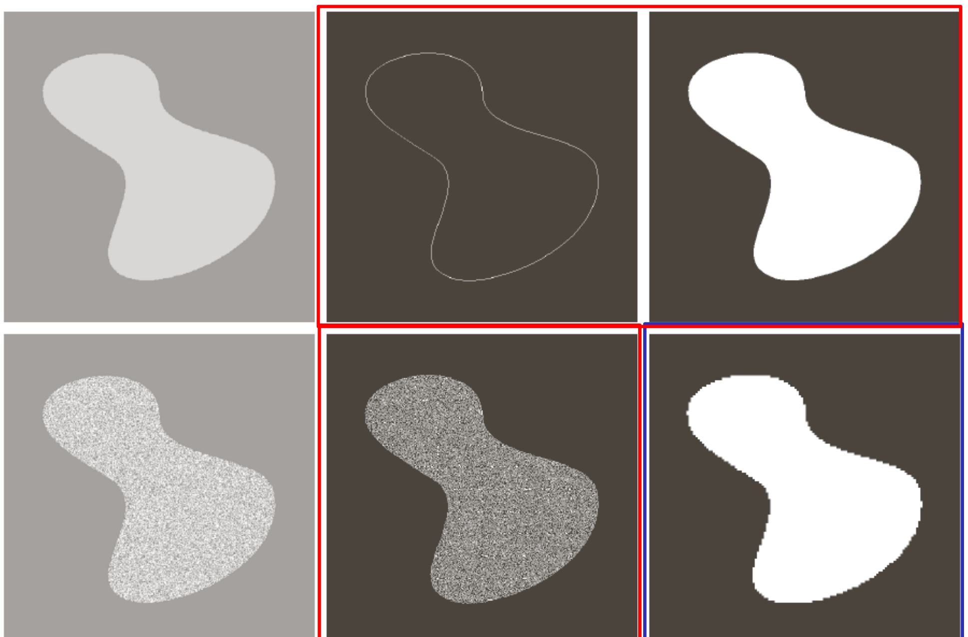
$$\bigcup_{i=1}^{n} R_i = R$$

R_i is a connected set, $i = 1,2,\ldots,n$

$R_i \cap R_j = \varnothing$ for all i and $j, i \neq j$

$Q(R_i) = True$ for all $i = 1,2,\ldots,n$

$Q(R_i \cup R_i) = False$ for any adjacent regions R_i and R_i



Edge-based segmentation algorithm

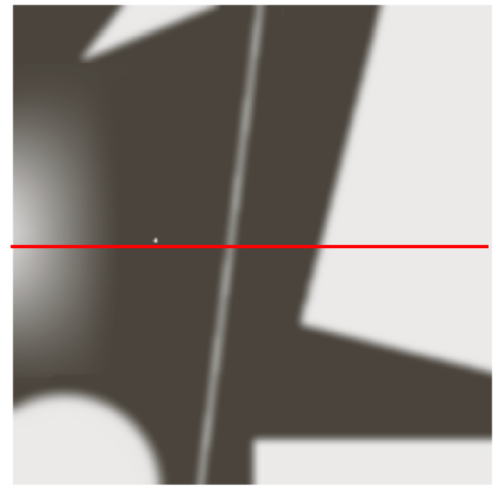
Region-based segmentation algorithm

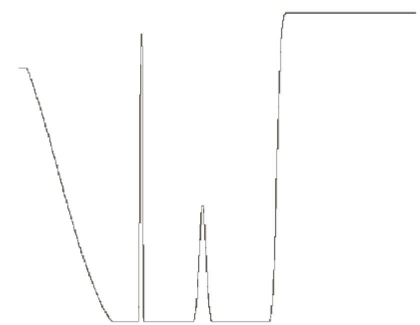
10.2 Point, Line, and Edge Detection

Detect using derivatives

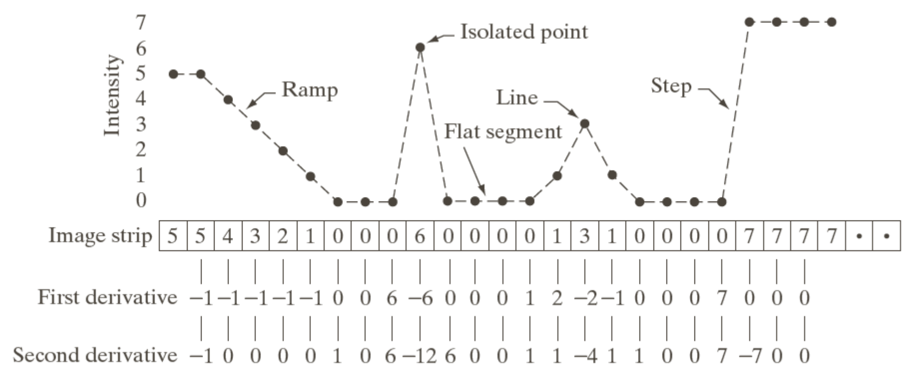
$$\frac{\partial f}{\partial x} = f'(x) = f(x+1) - f(x)$$

$$\frac{\partial^2 f}{\partial x^2} = f''(x) = f(x+1) + f(x-1) - 2f(x)$$





w_1	w_2	w_3
w_4	$\boxed{w_5}$	w_6
w_7	w_8	w_9

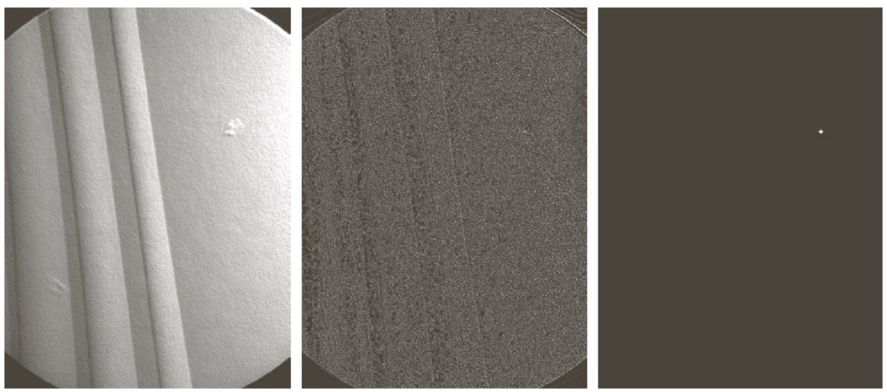


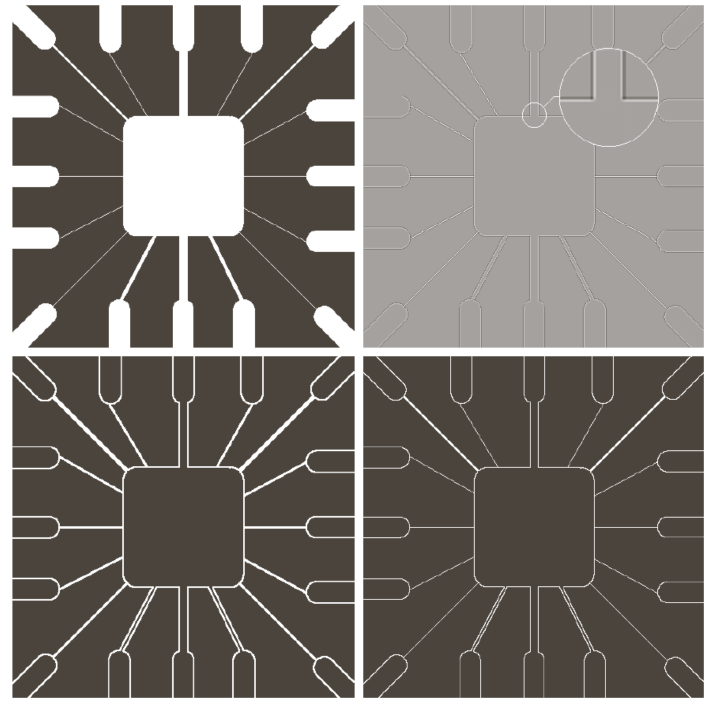
$$R = \sum_{k=1}^{9} w_k z_k = w_1 z_2 + w_2 z_2 + \ldots + w_9 z_9$$

Point, Line Detection

Mask

1	1	1
1	−8	1
1	1	1





Threshold

$|R| \geq T$

Line Detection

−1	−1	−1
2	2	2
−1	−1	−1

Horizontal

2	−1	−1
−1	2	−1
−1	−1	2

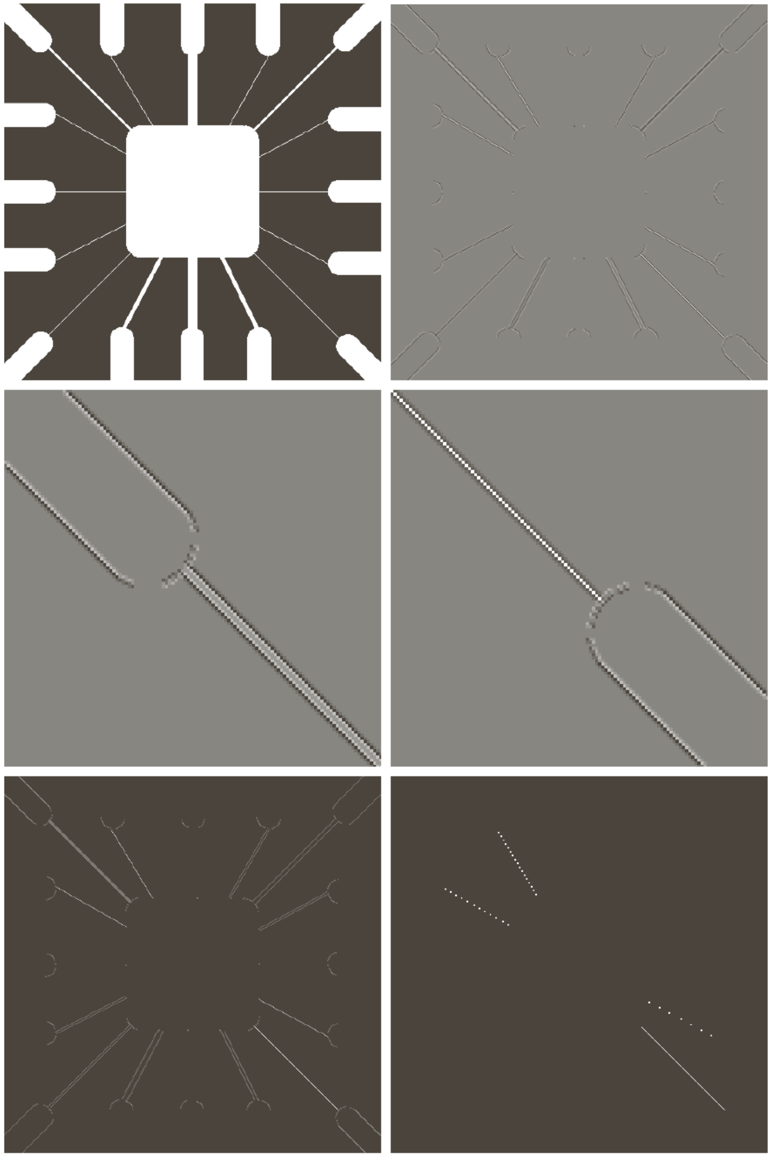
+45°

−1	2	−1
−1	2	−1
−1	2	−1

Vertical

−1	−1	2
−1	2	−1
2	−1	−1

−45°

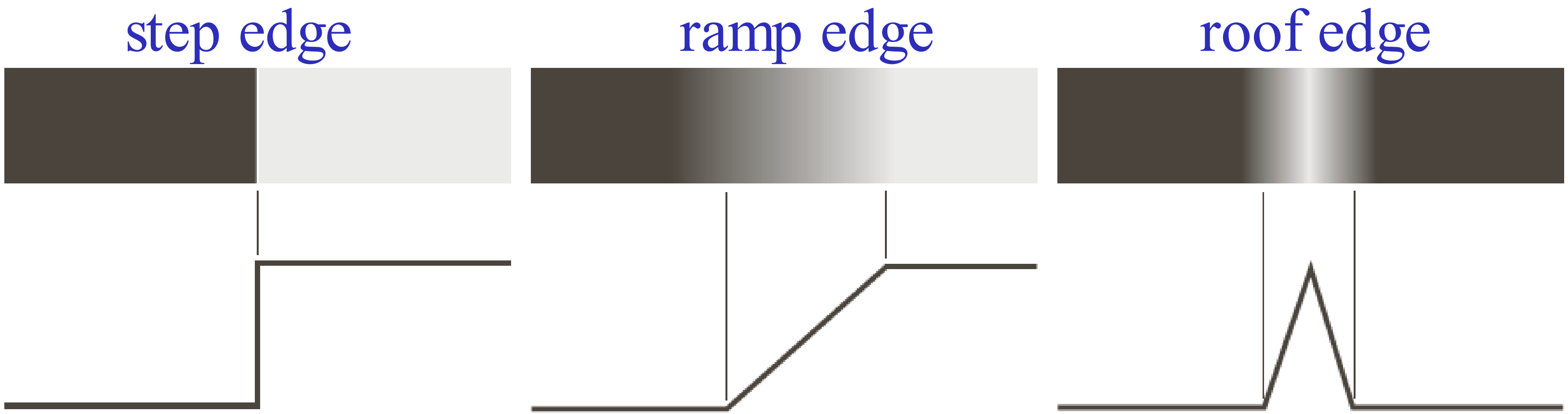


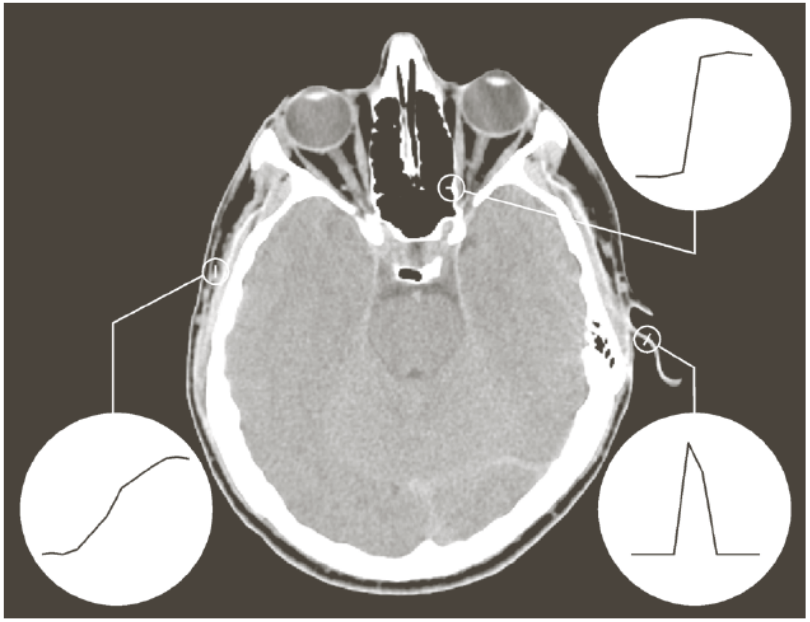
Detect line that are one pixel thick

45° mask

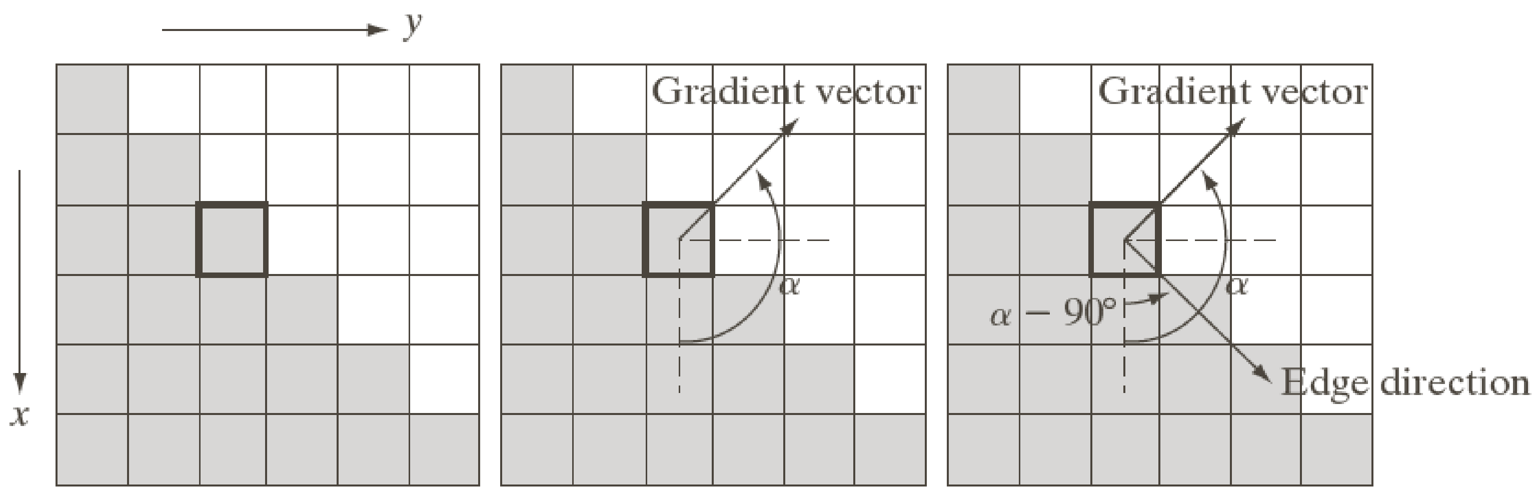
$|R| \geq T$

Edge Models





Edge Detection Methods – using image gradient



$$\nabla f \equiv grad(f) \equiv \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

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

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

Gradient operators

$$g_x = \frac{\partial f(x,y)}{\partial x} = f(x+1,y) - f(x,y)$$

$$g_y = \frac{\partial f(x,y)}{\partial y} = f(x,y+1) - f(x,y)$$

−1
1

−1	1

1-D Masks

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

2-D Masks

−1	0
0	1

0	−1
1	0

Roberts

Symmetric about the center points

0	1	1
−1	0	1
−1	−1	0

−1	−1	0
−1	0	1
0	1	1

Prewitt

0	1	2
−1	0	1
−2	−1	0

−2	−1	0
−1	0	1
0	1	2

Sobel

−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

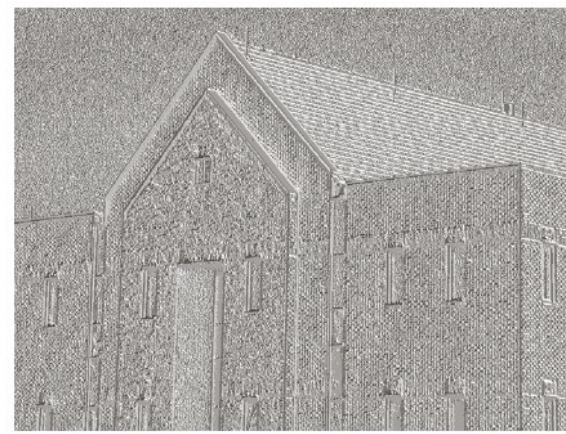
Sobel

Sobel mask edge detection example

$|g_x|$

$|g_y|$

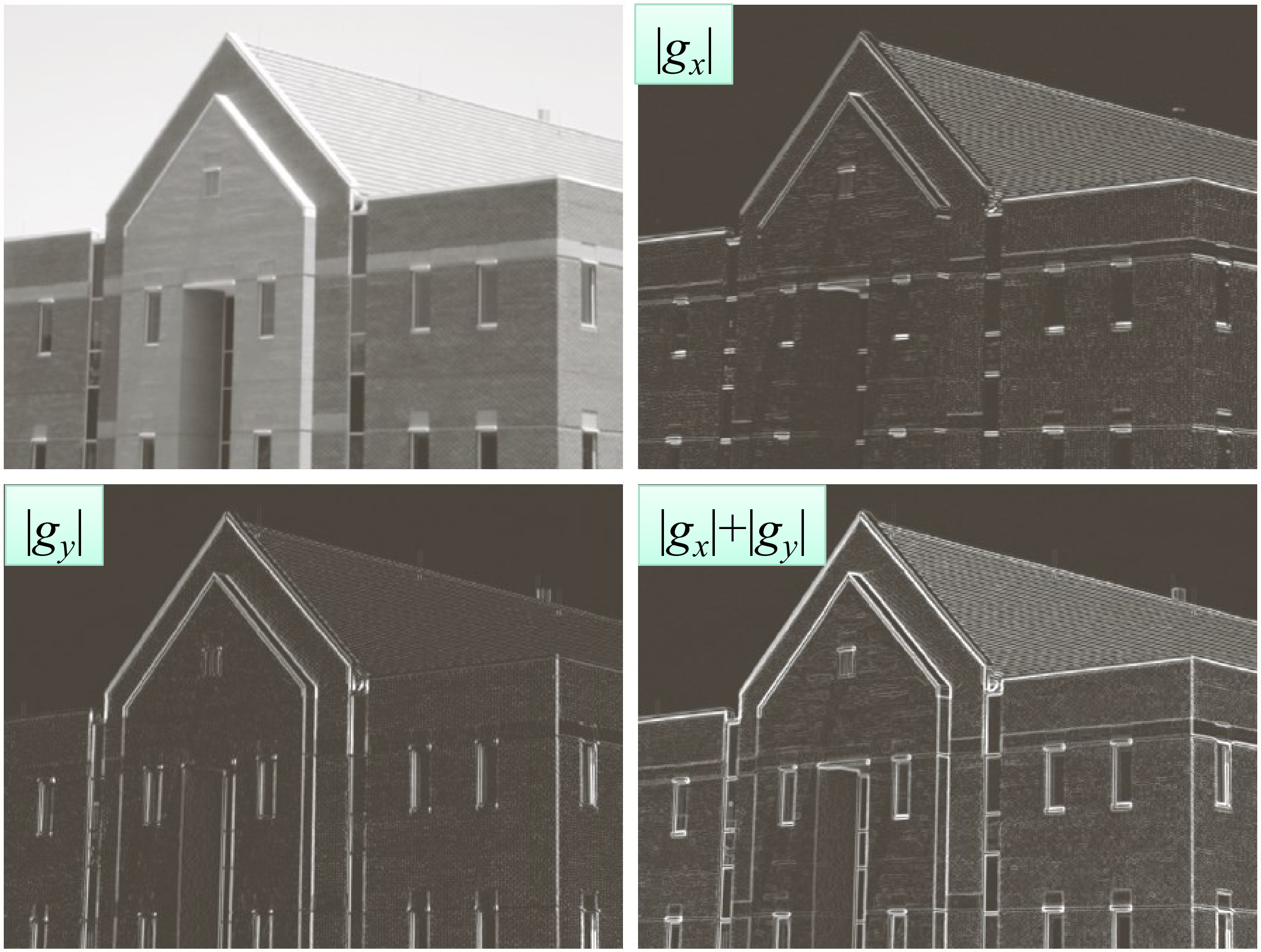
$|g_x|+|g_y|$



Gradient angle

Reduce noise: Smoothing

Smooth with 5×5 average filter

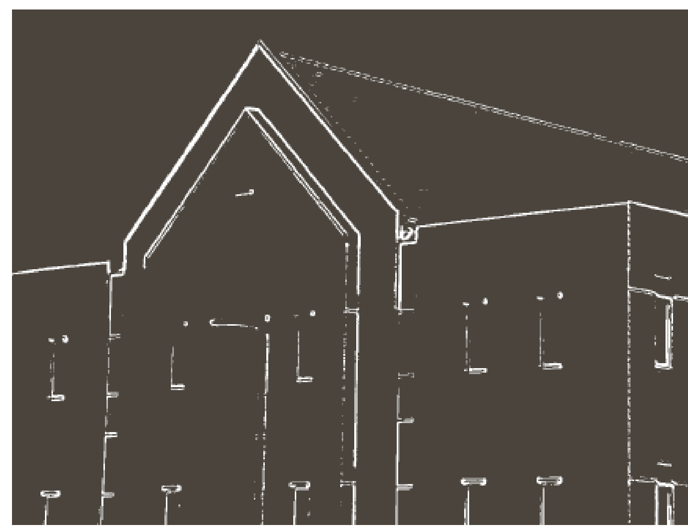


Diagonal edge detection

0	1	2
-1	0	1
-2	-1	0

-2	-1	0
-1	0	1
0	1	2

Edge detection + threshold: 33% highest value



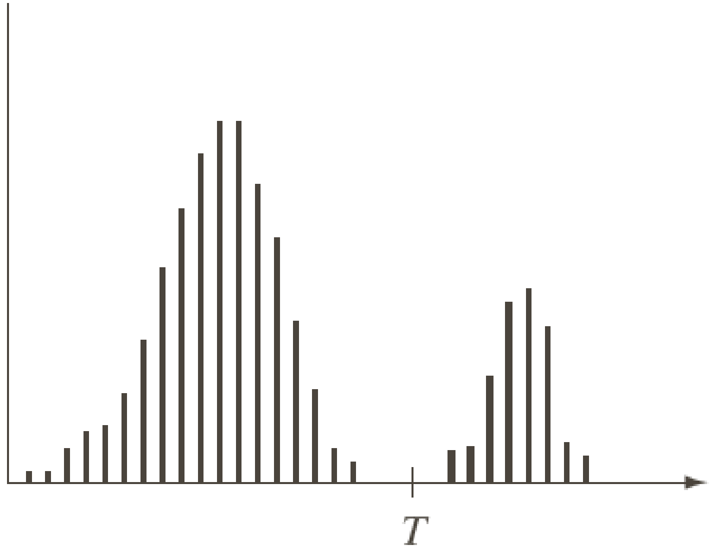
10.3 Thresholding

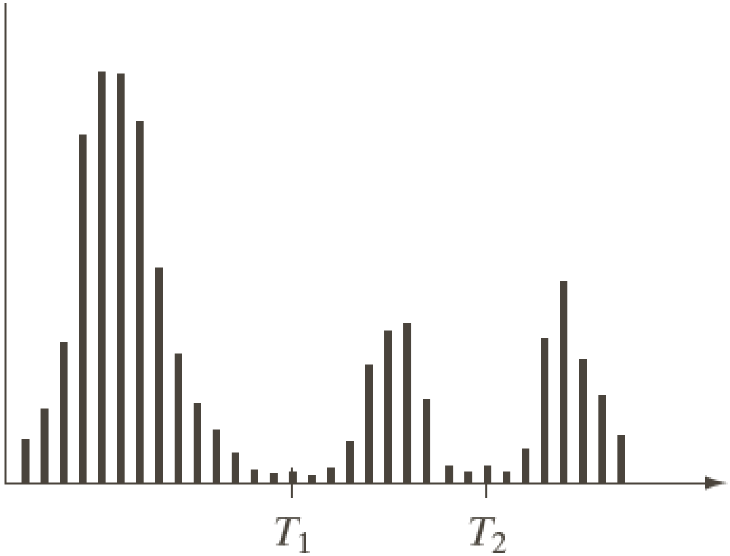
The basics of intensity thresholding

Multiple thresholding

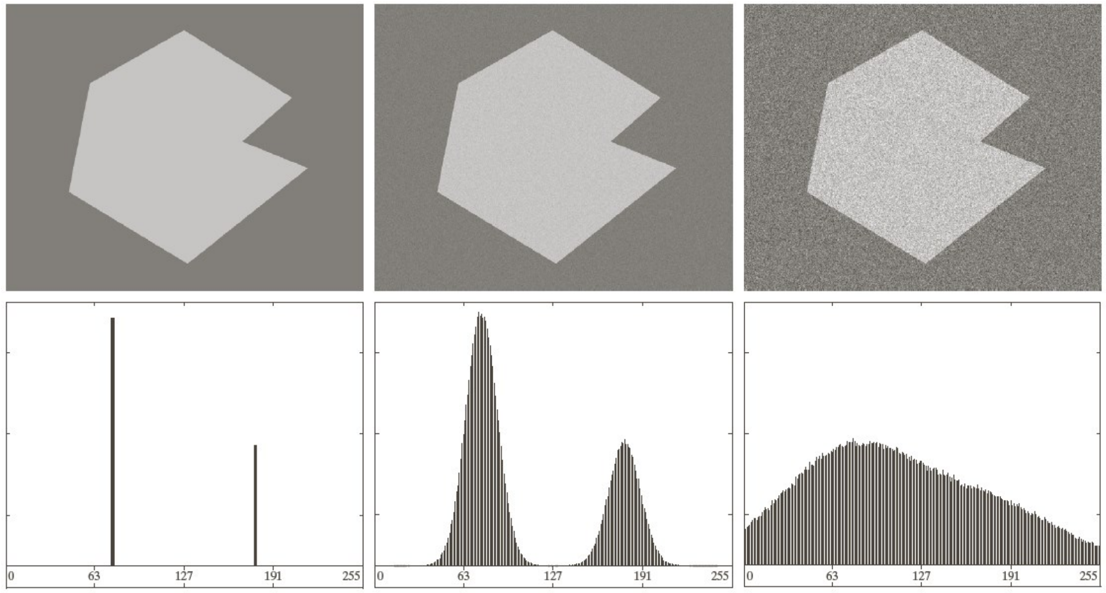
$$g(x,y)=\begin{cases}1 & \text{if } f(x,y)>T\\ 0 & \text{if } f(x,y)\le T\end{cases}$$

$$g(x,y)=\begin{cases}a & \text{if } f(x,y)>T_2\\ b & \text{if } T_1\le f(x,y)\le T_2\\ c & \text{if } f(x,y)\le T_1\end{cases}$$

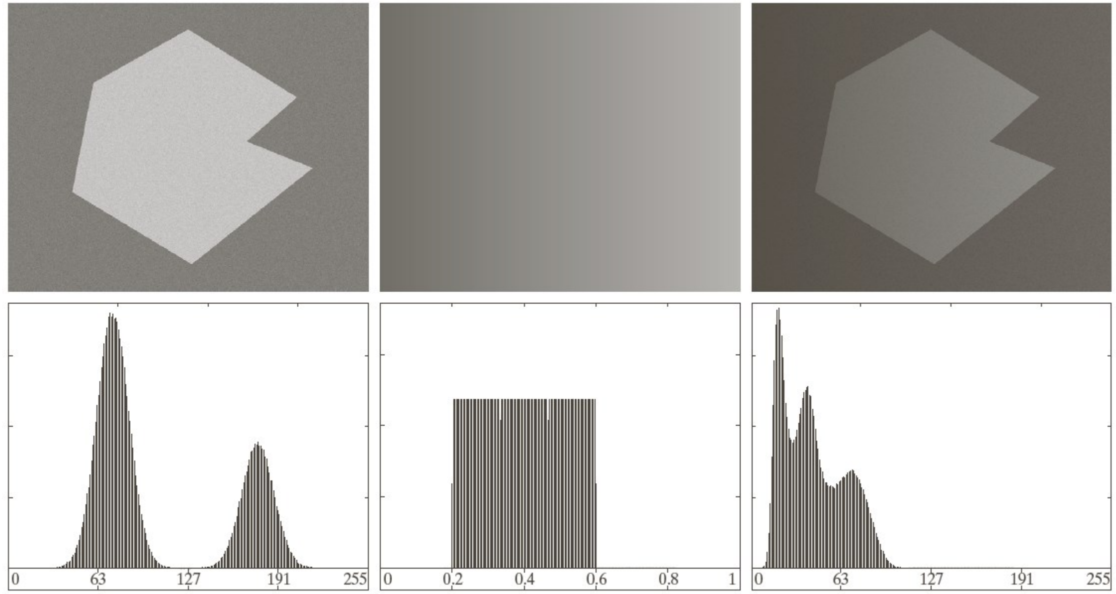




The role of noise in image thresholding



The role of illumination and reflectance



play a central role in image segmentation

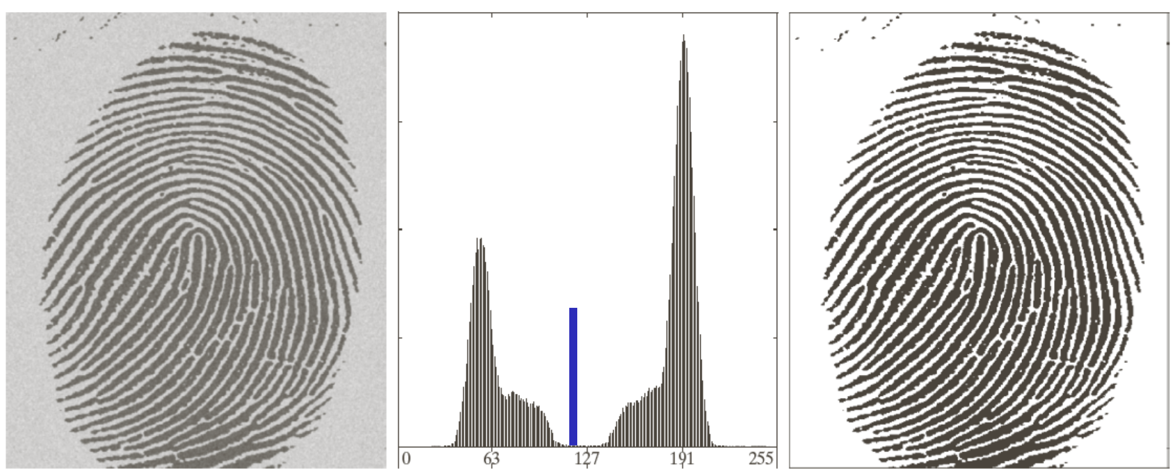
Basic Global Thresholding – when the intensity distributions of objects and background pixels are sufficiently distinct

Iterative thresholding algorithm

1. Select initial threshold T
2. Segment image into $\{G_1, G_2\}$ using T using

$$g(x,y)=\begin{cases}1 \text{ if } & f(x,y)>T\\ 0 \text{ if } & f(x,y)\le T\end{cases}$$

3. Compute mean values m_1 and m_2 for the pixels in G_1 and G_2
4. Computer new threshold: $T=(m_1+m_2)/2$
5. Repeat steps 2 through 4 until the difference between values of T in successive iterations is smaller than ΔT



T=125

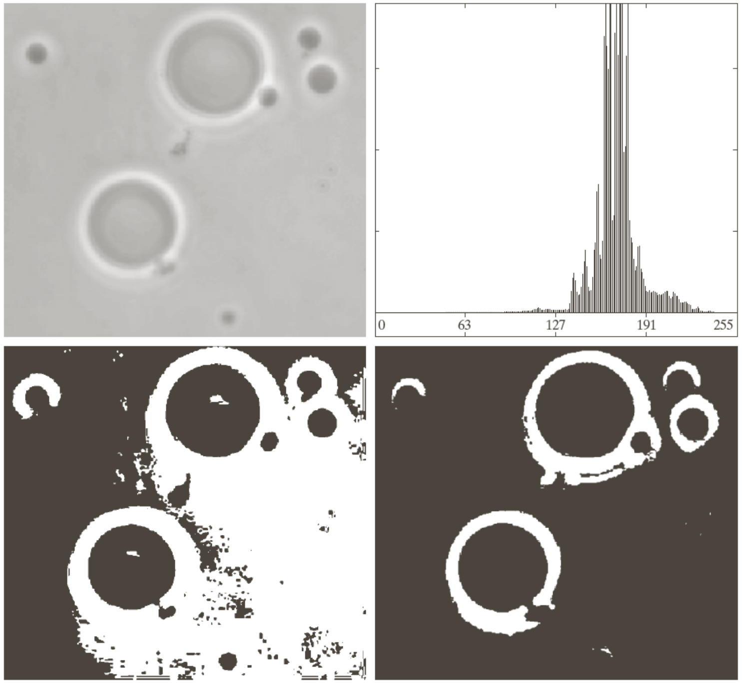
Optimal Global Thresholding Using Otsu's Method

– maximize the between-class variance

– perform on histogram $k = 0,1,2,\ldots,L-1$

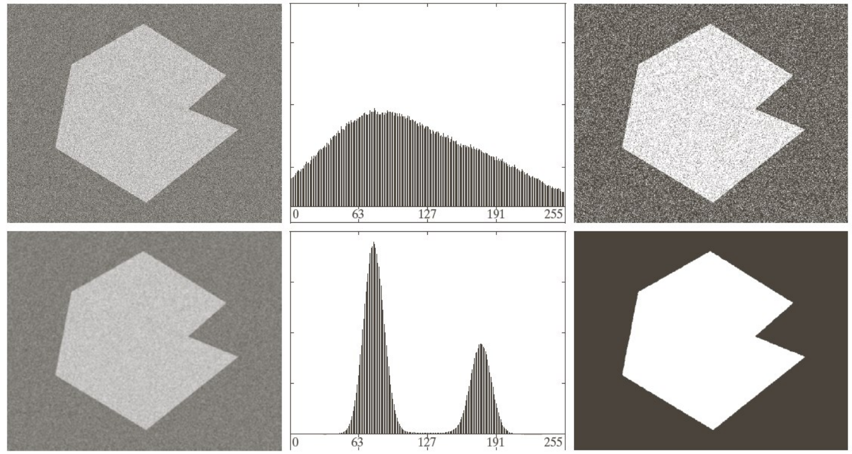
$$g(x,y) = \begin{cases} 1 & \text{if } f(x,y) > k^* \\ 0 & \text{if } f(x,y) \le k^* \end{cases}$$

1. Compute normalized histogram of input image $p_i, i = 0,1,\ldots,L-1$
2. Compute the cumulative sums $P_1(k)$ using $P_1(k) = \sum_{i=0}^{k} p_i$
3. Compute the cumulative means $m(k)$ using $m(k) = \sum_{i=0}^{k} i p_i$
4. Compute the global intensity mean m_G, using $m_G = \sum_{i=0}^{L-1} i p_i$
5. Compute the between-class variance $\sigma_B^2(k)$ using $\sigma_B^2(k) = \dfrac{[m_G P_1(k) - m(k)]^2}{P_1(k)[1 - P_1(k)]}$
6. Obtain the Otsu threshold, k^*, as the value of k for which $\sigma_B^2(k)$ is maximum. If the maximum is not unique, obtain k^* by averaging the values of k corresponding to the various maxima detected.
7. Obtain the separability measure, η^*, by evaluating $\eta(k) = \dfrac{\sigma_G^2(k)}{\sigma_G^2}$ at $k = k^*$.



Basic global Otsu's method

Using Image Smoothing to Improve Global Thresholding

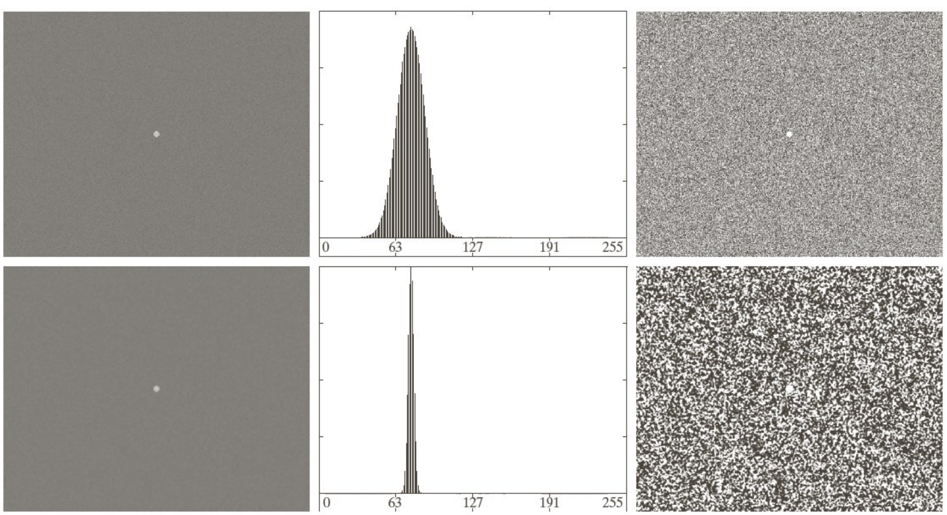


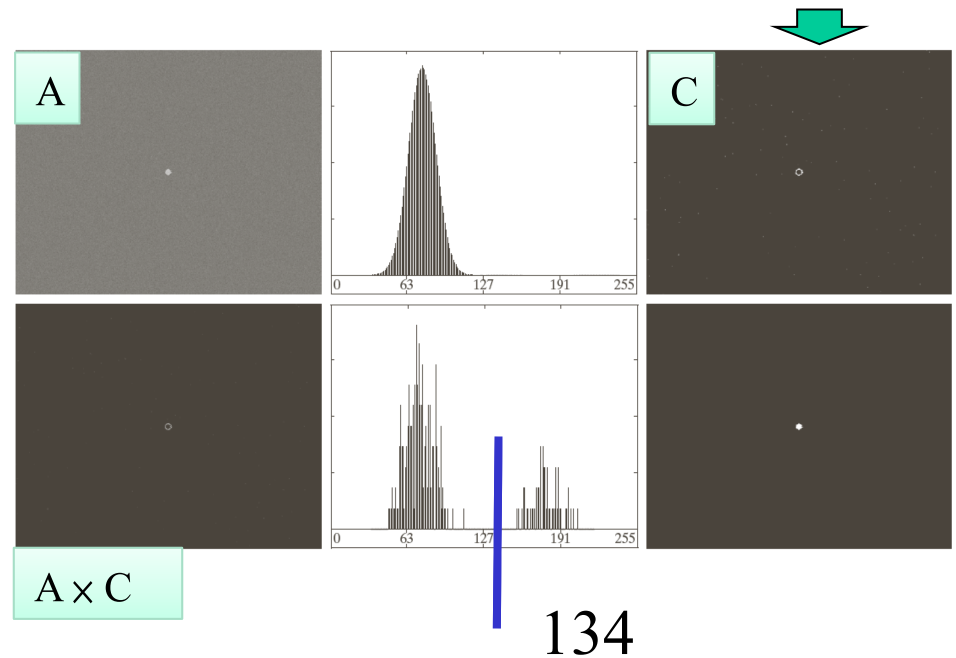
Otsu’s method

5×5 averaging

Using Edges to Improve Global Thresholding

Gradient magnitude image threshold at 99.7%

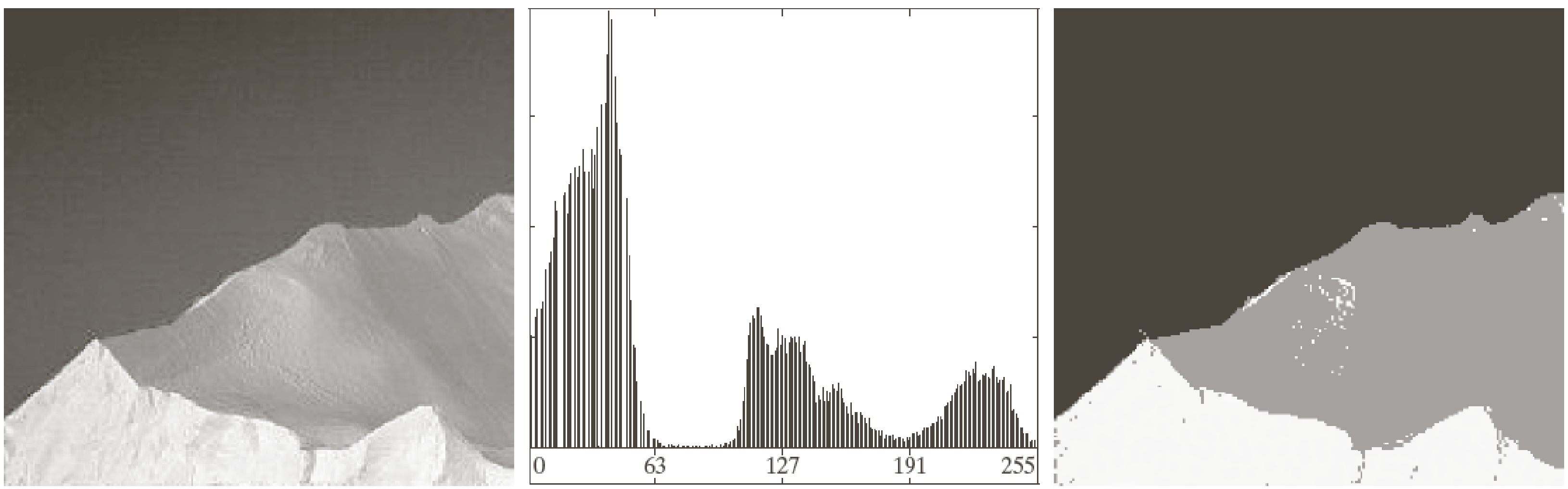




5×5 averaging

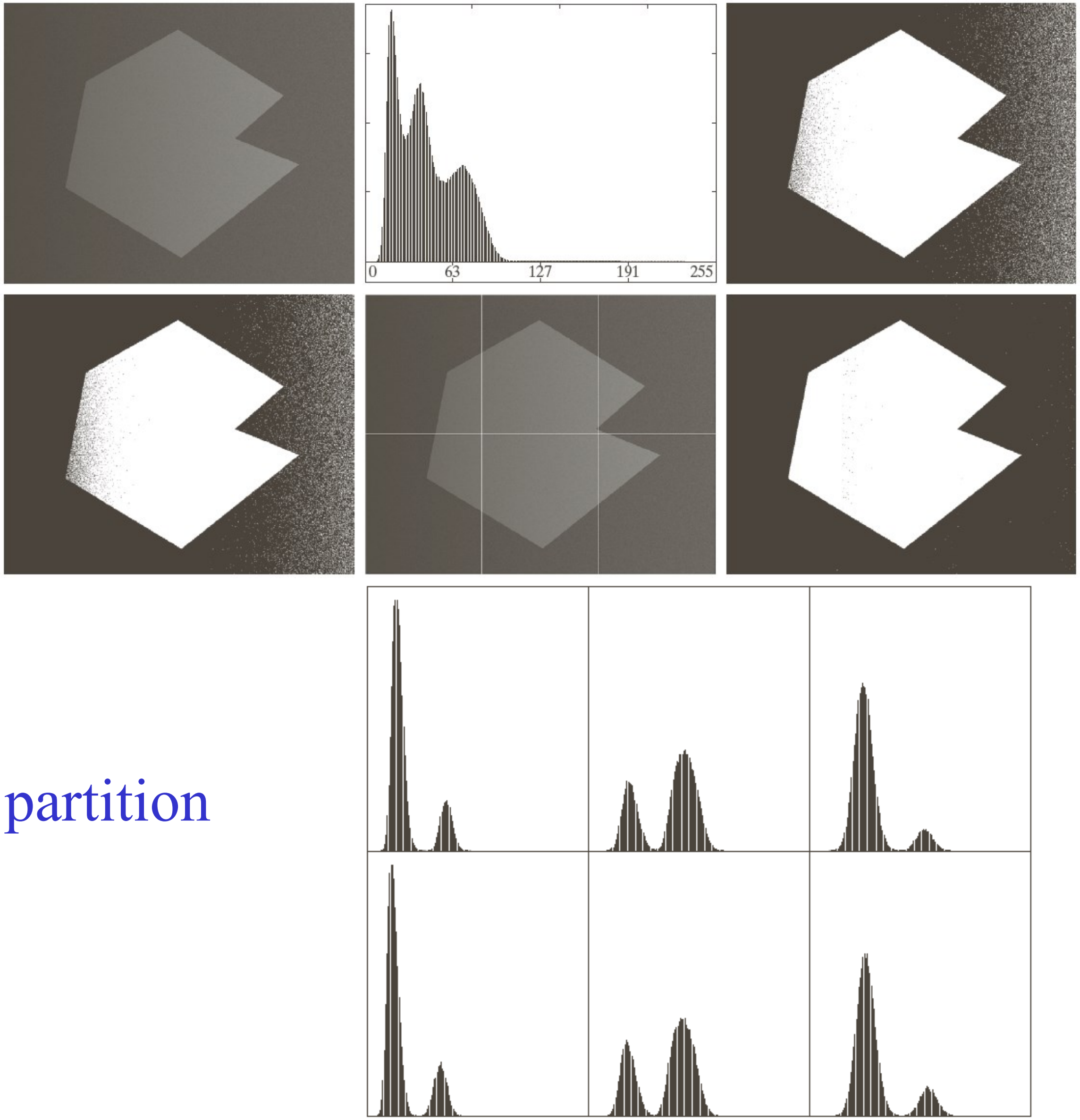
Multiple Thresholding

$$g(x,y)=\begin{cases} a \text{ if } f(x,y)\le k_1^* \\ b \text{ if } k_1^* < f(x,y)\le k_2^* \\ c \text{ if } f(x,y) > k_2^* \end{cases}$$



Variable Thresholding

Iterative global



Otsu's method

Otsu's method

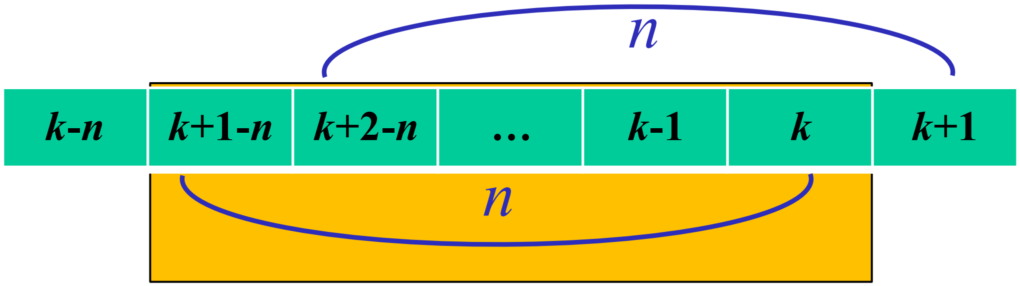
Image partition

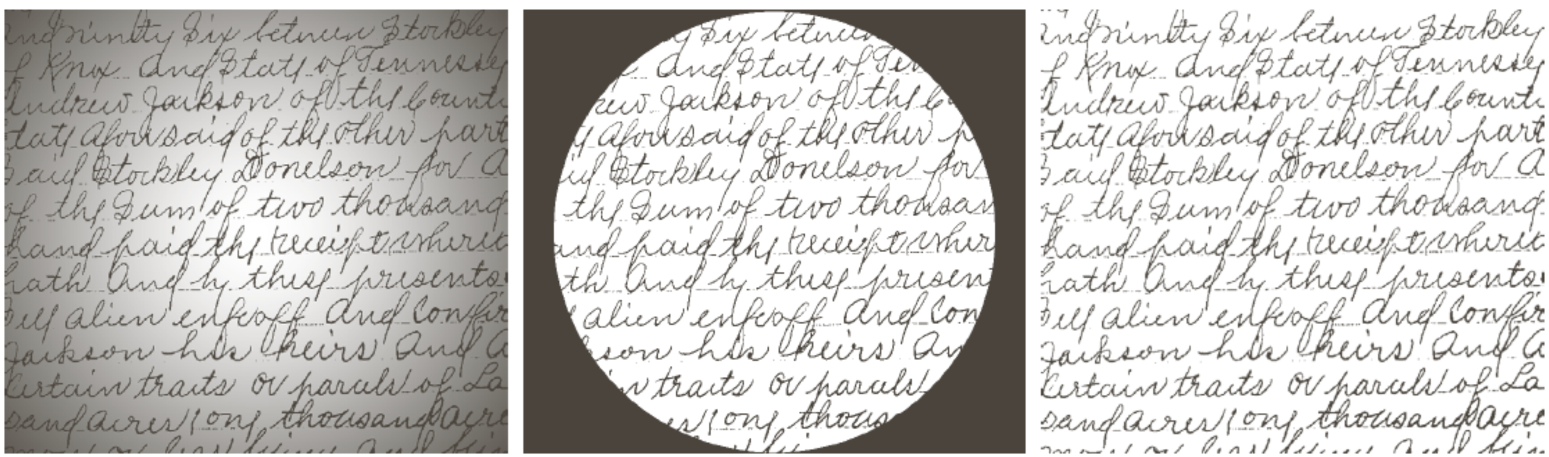
Variable thresholding based on local image properties

Using moving averages

$$m(k+1) = \frac{1}{n} \sum_{i=k+2-n}^{k+1} z_i = m(k) + \frac{1}{n}(z_{k+1} - z_{k-n})$$

$$m(1) = z_1 / n$$





Otsu's method

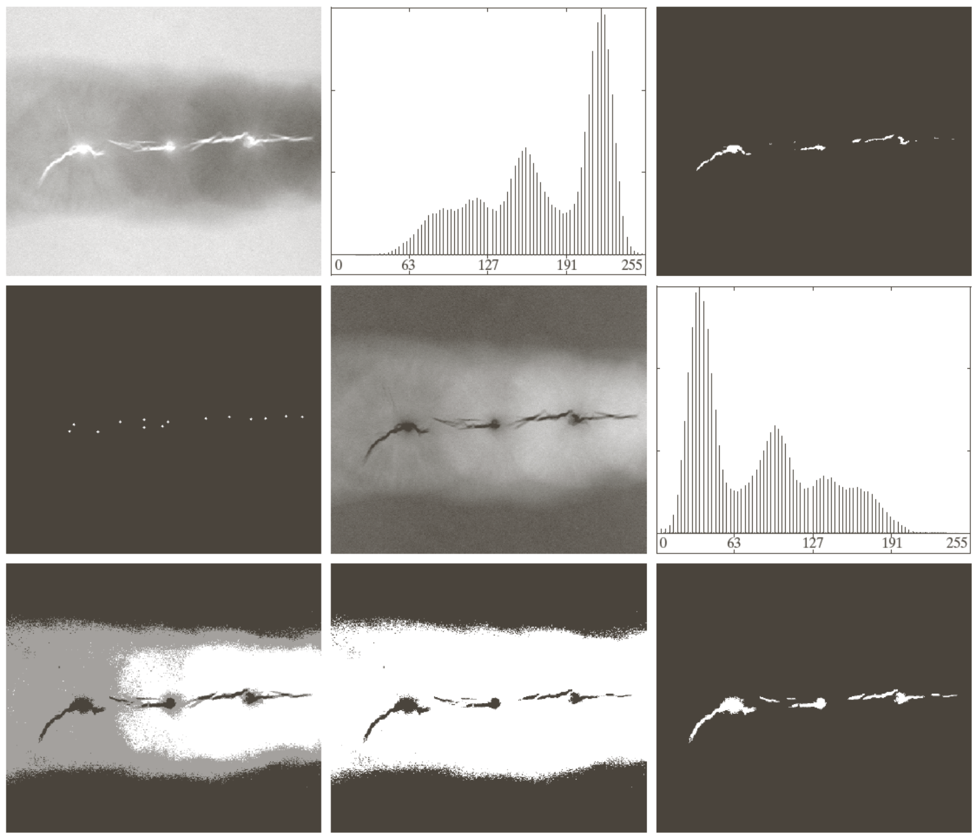
moving averaging

10.4 Region-Based Segmentation

10.4.1 Region growing – a procedure that groups pixels or subregions into larger regions based on predefined criteria (e.g. similar color) for growth.

Threshold
99% = 254

Seeds:
connected
component



Region growing

Dual threshold

10.4 Region-Based Segmentation

10.4.1 Region Splitting and Merging

1. Split into four disjoint quadrants any region R_i for $Q(R_i)$=false
2. When no further splitting is possible, merge any adjacent regions R_j and R_k for which $Q(R_j \cup R_k)$=true
3. Stop when no further merging is possible