Edge Points and their Detection

Detection Of Discontinuities

- Are based on detecting sharp changes in intensity in neighbouring pixels
- Discontinuities are
  - Edges

    Intensity changes abruptly from current pixel to next pixels along horizontal/ vertical/diagonal lines

    Connected set of edge pixels belong to edge
  - Lines

    Intensity of background on either side of the edge pixel is either higher or lower than edge pixels
  - Isolated points

    Intensity changes abruptly from current pixel to next pixel

    Line whose length and width is 1 pixel

Detection of isolated points using Laplacian

Laplacian mask, w_k

1	1	1
1	-8	1
1	1	1

Center pixel of filtered image, $R(x, y)$

If $|R(x,y)| >= T$ then $g(x,y) = 255$

else $g(x,y) = 0$

image, g_k

4	4	4
3	10	1
4	1	5

Filtered image

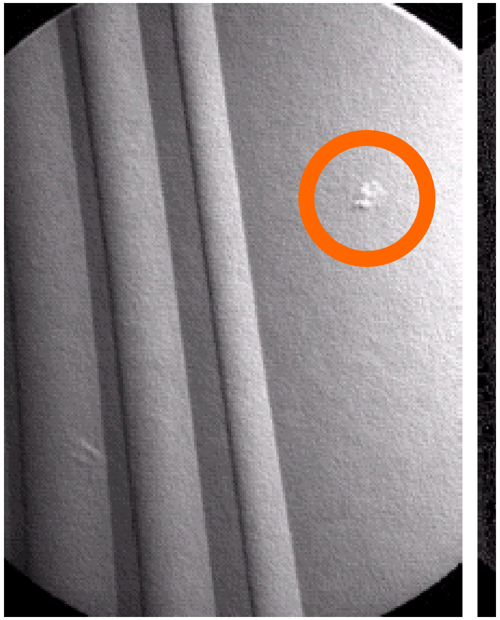
4	4	4
3	-54	1
4	1	5

For T = 15

4	4	4
3	255	1
4	1	5

Isolated point

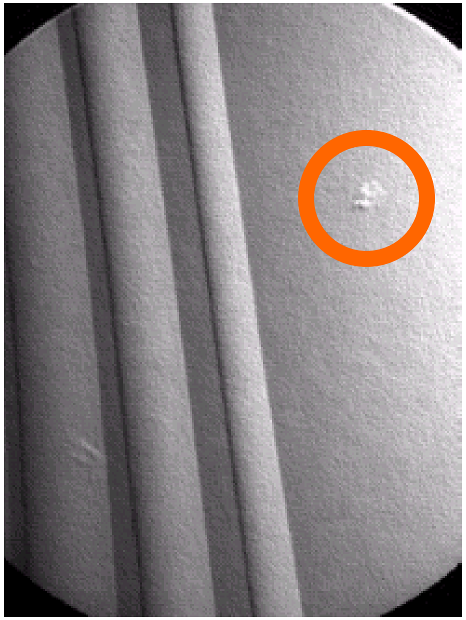
−1	−1	−1
−1	8	−1
−1	−1	−1



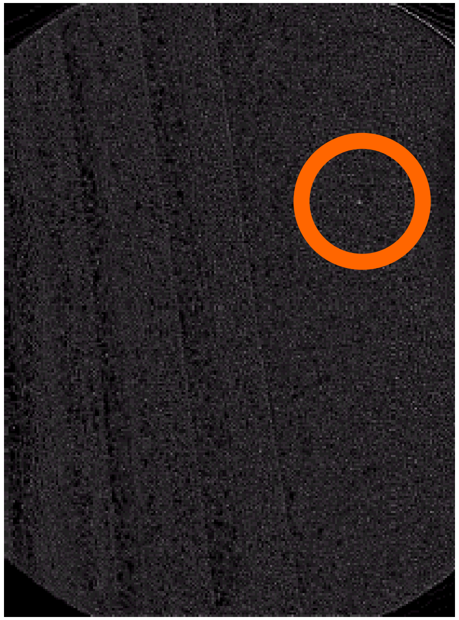
X-ray image of
a turbine blade

Isolated points

−1	−1	−1
−1	8	−1
−1	−1	−1



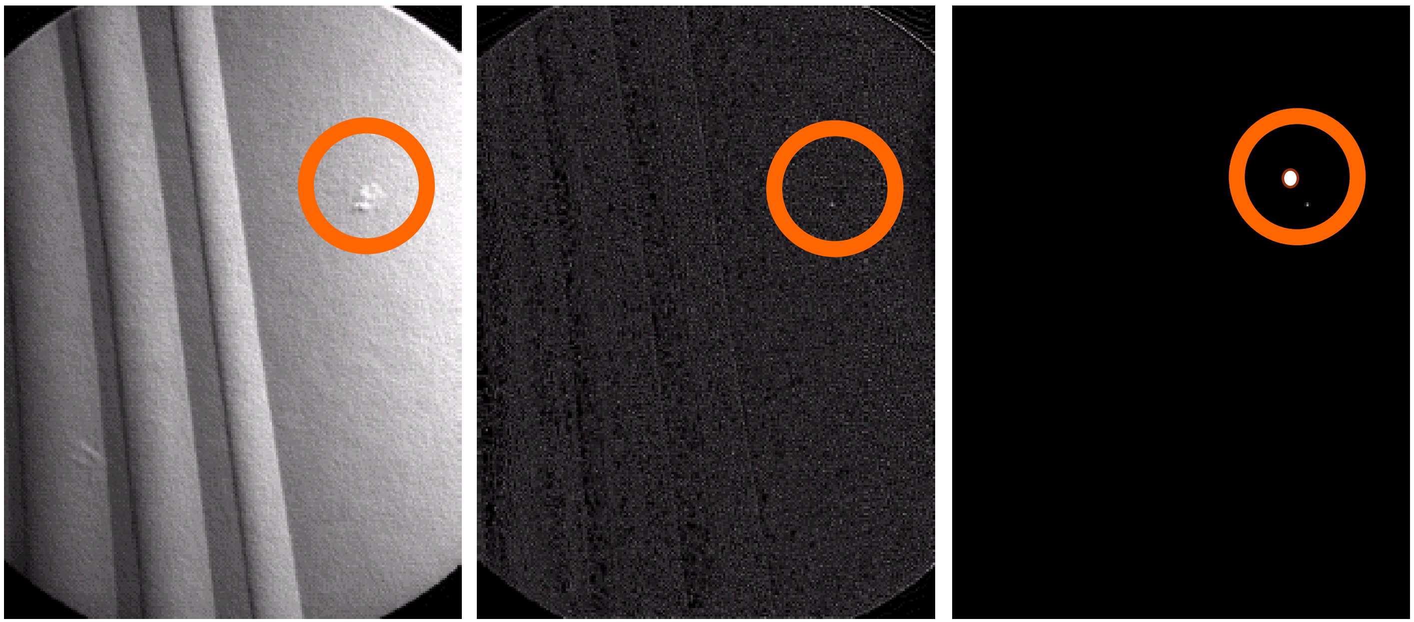
X-ray image of a turbine blade



Result of point detection

Isolated points

−1	−1	−1
−1	8	−1
−1	−1	−1



X-ray image of a turbine blade

Result of point detection

- Binary image
- Threshold at 90% of the highest intensity

Edge Detection using gradient operators

- The gradient of an image $f(x,y)$ at location (x,y) is defined as the vector:

$$\nabla \mathbf{f} = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

- The magnitude of this vector: $\nabla f = \text{mag}(\nabla \mathbf{f}) = \left[g_x^2 + g_y^2\right]^{1/2}$

- The direction of this vector: $\alpha(x, y) = \tan^{-1}\left(\frac{g_y}{g_x}\right)$

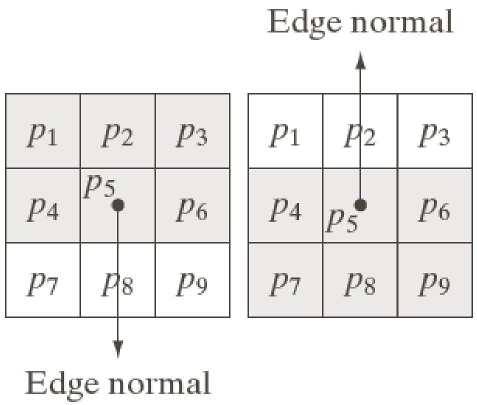
$$\frac{\partial f}{\partial x} = g_x = f(x+1) - f(x) \qquad \frac{\partial f}{\partial y} = g_y = f(y+1) - f(y)$$

Gradient Vector

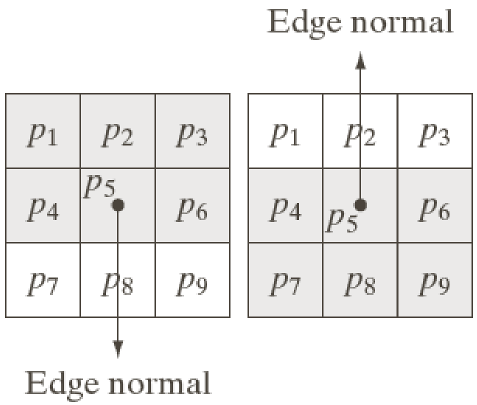
$$\nabla f = \text{mag}(\nabla \mathbf{f}) = \left[g_x^2 + g_y^2\right]^{1/2} \qquad \alpha(x, y) = \tan^{-1}\left(\frac{g_y}{g_x}\right)$$

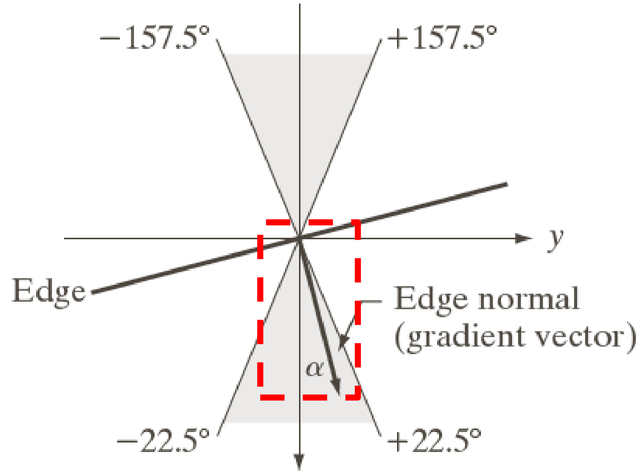
- Points in the direction of the greatest rate of change intensity at location (x,y)
- Magnitude is the value of rate of change in the direction of gradient vector
- Direction is measured with respect to vertical axis
- Direction of gradient vector is also called edge normal

Gradient angle



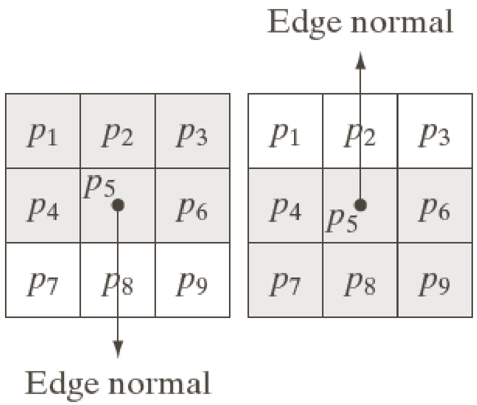
Gradient angle

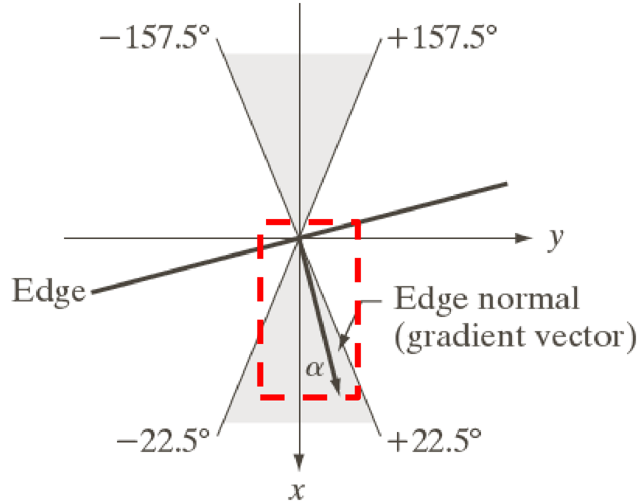


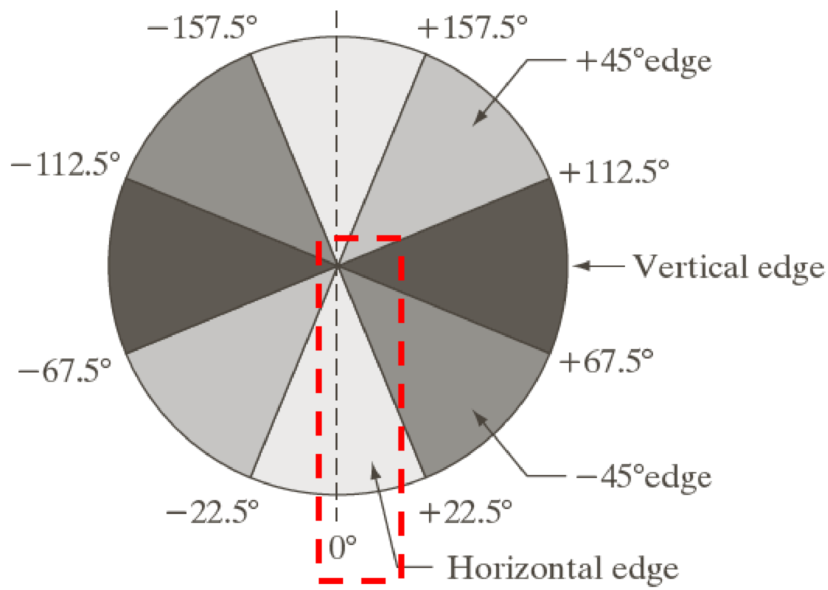


Angle of gradient vector = angle of edge with horizontal axis

Gradient angle

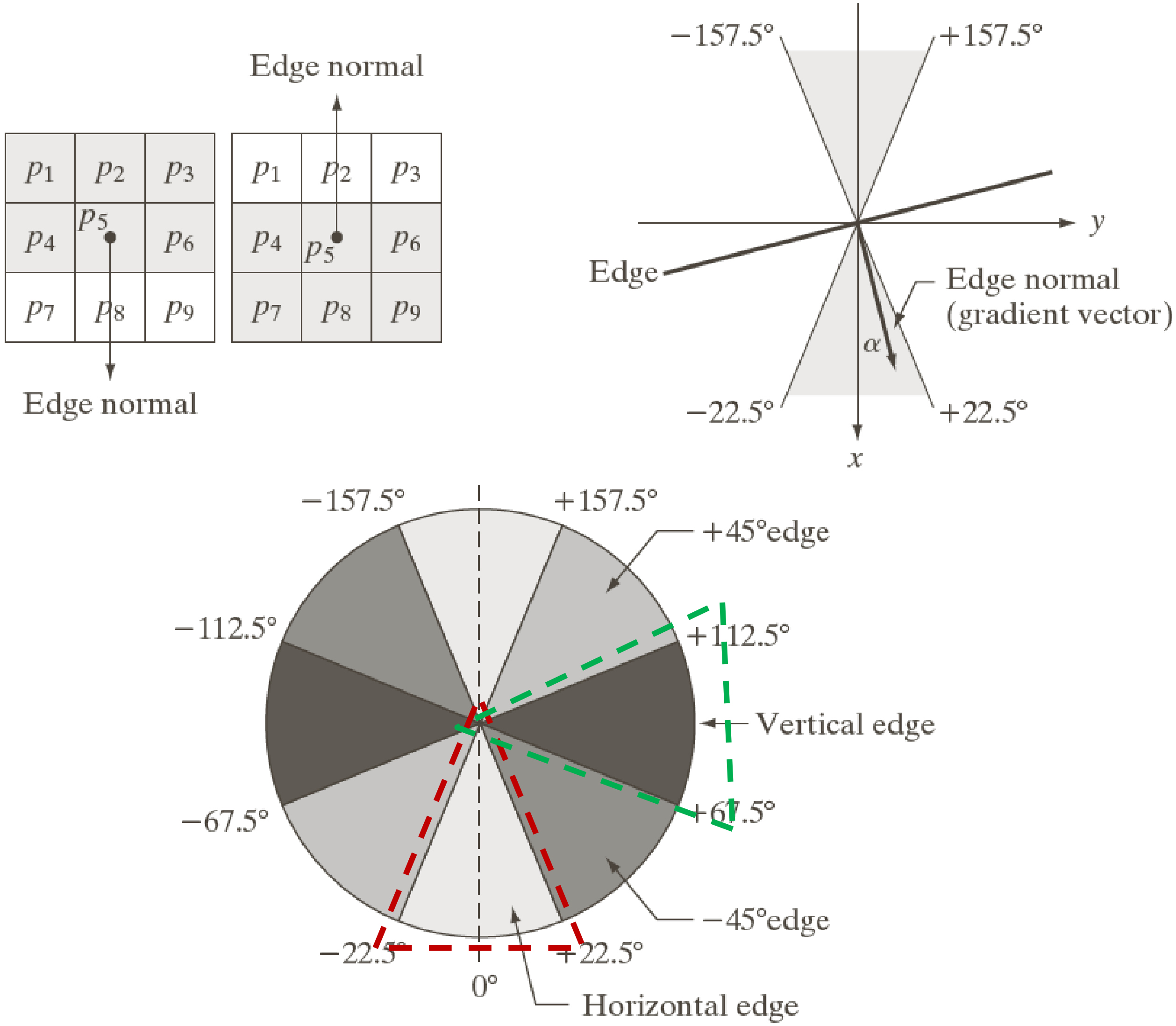






Angle of gradient vector = angle of edge with horizontal axis

Gradient angle



Angle of gradient vector = angle of edge with horizontal axis

Gradient vectors of Image

- Magnitude of gradient vector at pixel can be shown in the form of image
- Magnitude of gradient vector (gradient image) is given by

$$
\begin{aligned}
M(x, y) &= mag(\nabla \mathrm{f}) \\
&= \left[G_x^2 + G_y^2\right]^{1/2} \\
&= \left[\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2\right]^{1/2}
\end{aligned}
$$

- Size of $M(x,y)$ is same as image
- For practical reasons magnitude can be simplified as

$$M(x, y) \approx |G_x| + |G_y|$$

Common Edge Detectors

- Edge detection filters in spatial domain

−1	−1	−1		−1	0	1
0	0	0		−1	0	1
1	1	1		−1	0	1

Prewitt

−1	−2	−1		−1	0	1
0	0	0		−2	0	2
1	2	1		−1	0	1

Sobel

Sobel Operators (gradient vector)

image

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

Mask for g_x

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

Mask for g_y

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

$$g_x = \left|(z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)\right|$$

$$g_y = \left|(z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)\right|$$

$$M(x, y) \approx |g_x| + |g_y|$$

Sobel Operators for gradient vector

$$g_x = \left|(z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)\right|$$

$$g_y = \left|(z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)\right|$$

image

10	10	50	10	10
10	10	50	10	10
10	10	50	10	10
10	10	50	10	10

Vertical mask, gy

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

Horizontal mask, gx

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

g_y

10	10	50	10	10
10	160	0	160	10
10	160	0	160	10
10	10	50	10	10

+

g_x

10	10	50	10	10
10	0	0	0	10
10	0	0	0	10
10	10	50	10	10

=

$g_y + g_x$

10	10	50	10	10
10	160	0	160	10
10	160	0	160	10
10	10	50	10	10

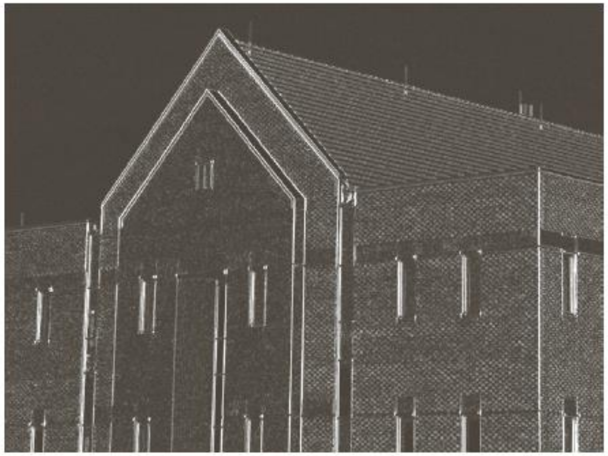
Edge detection

Edge detection

horizontal edges using sobel mask

Edge detection

horizontal edges using sobel mask



vertical edges using sobel mask

Edge detection

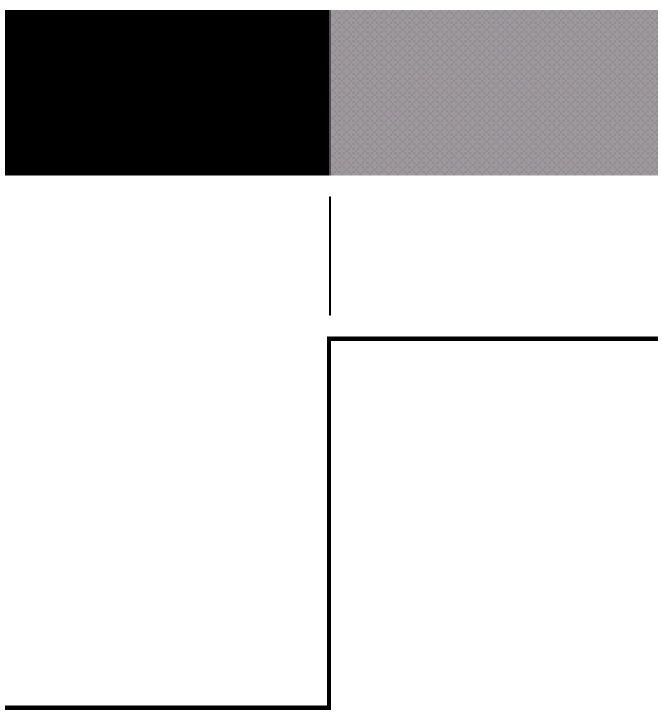
horizontal edges using sobel mask

vertical edges using sobel mask

Addition of vertical and horizontal edges

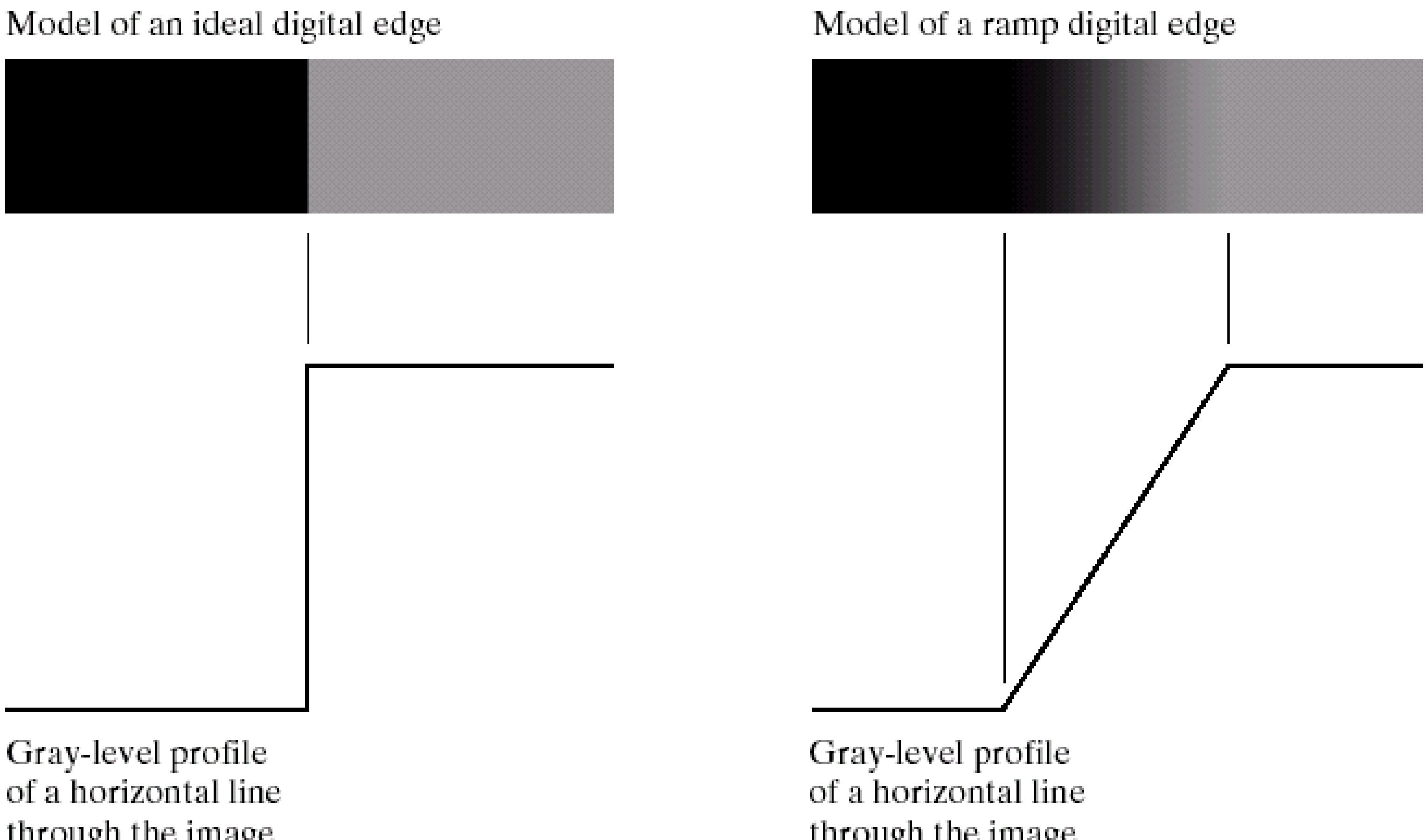
Edge Detection

Model of an ideal digital edge

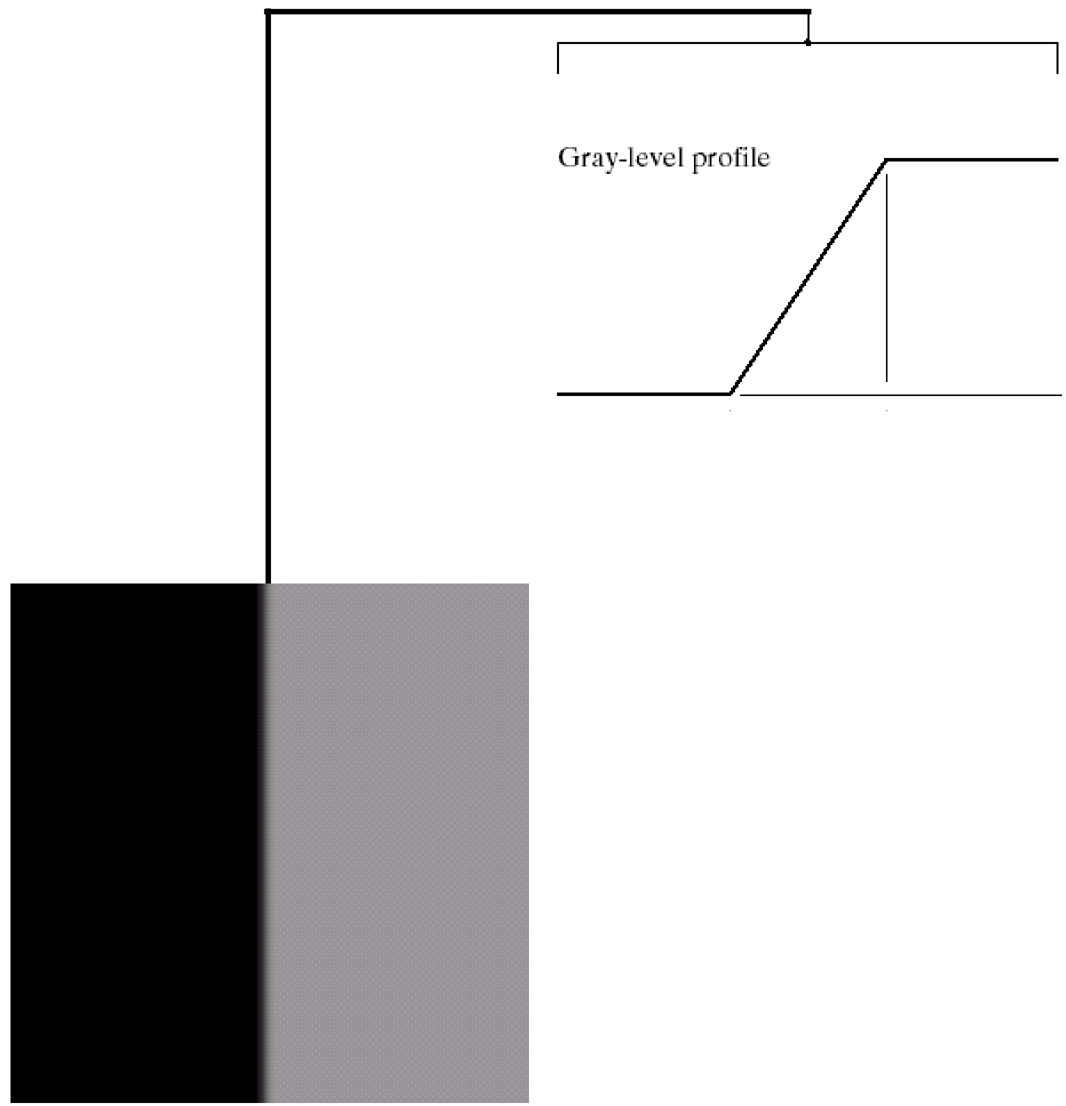


Gray-level profile
of a horizontal line
through the image

Edge Detection

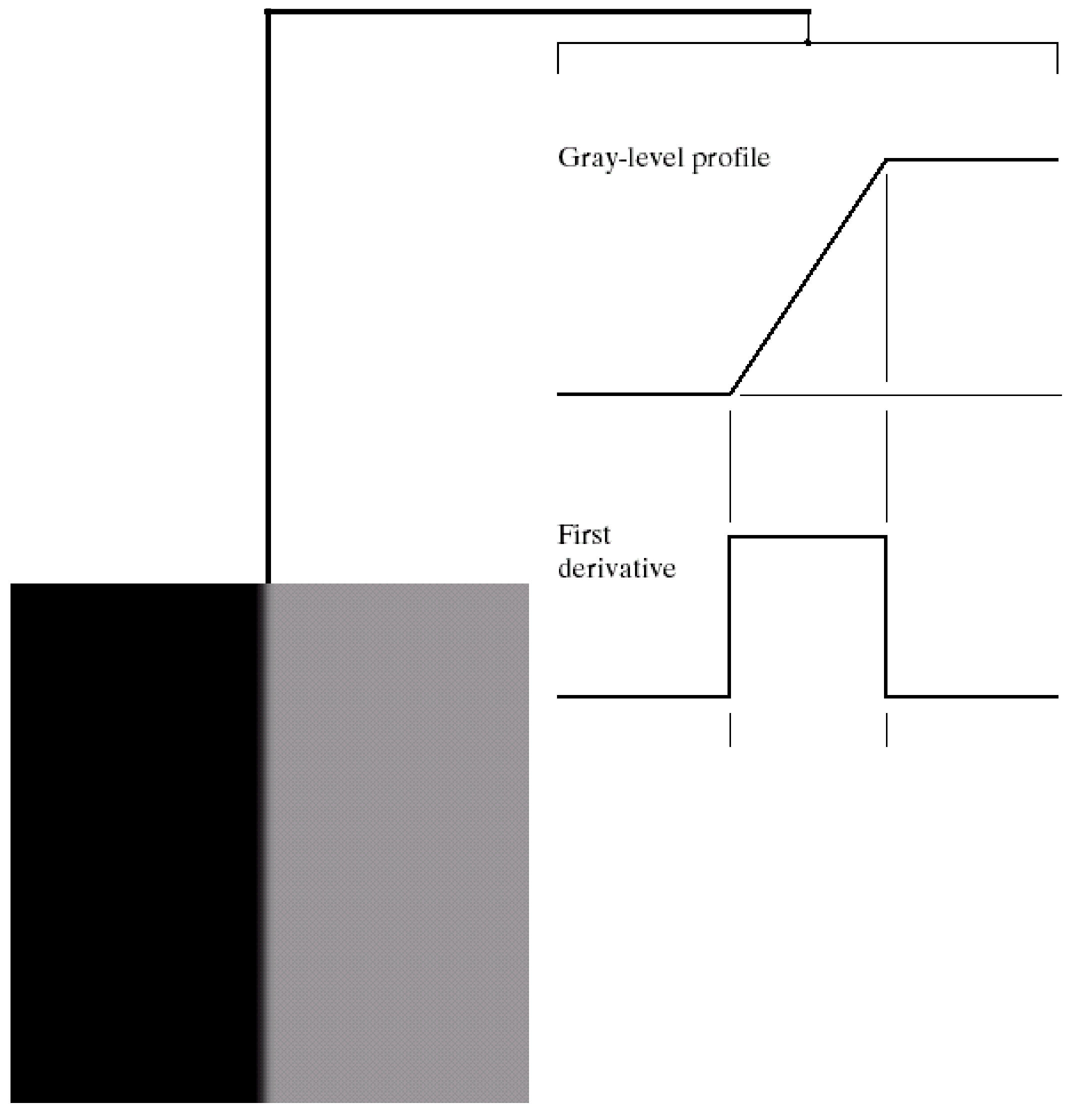


Edges & Derivatives



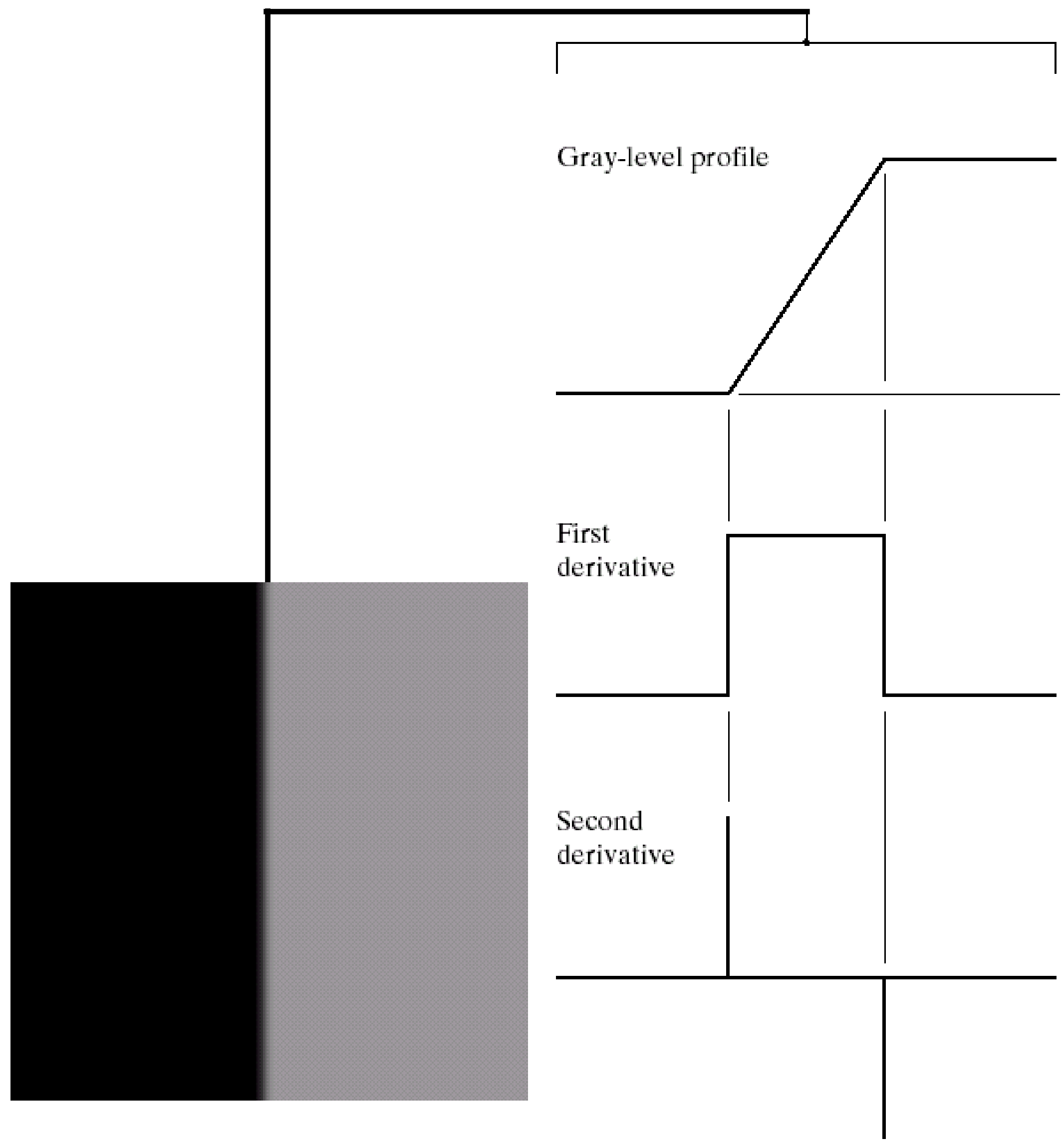
Edges & Derivatives

- 1^{st} derivative denotes the location of edge



Edges & Derivatives

- 1st derivative denotes the location of edge
- 2nd derivative shows location of edge and its direction



Derivatives & Noise

- Derivative based edge detectors are sensitive to noise

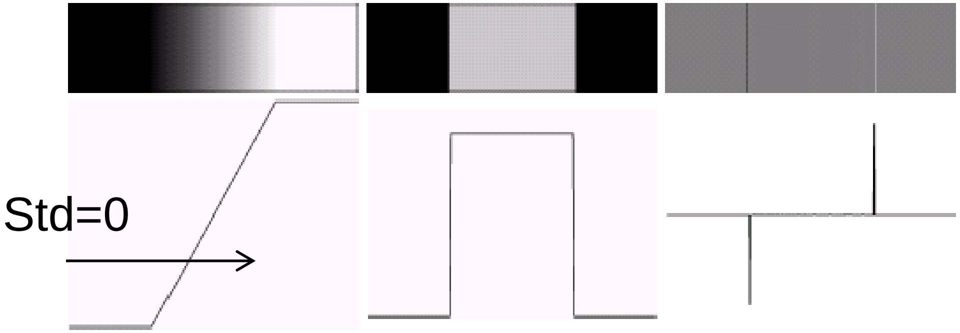
Derivatives & Noise

- Derivative based edge detectors are sensitive to noise

Image
with noise

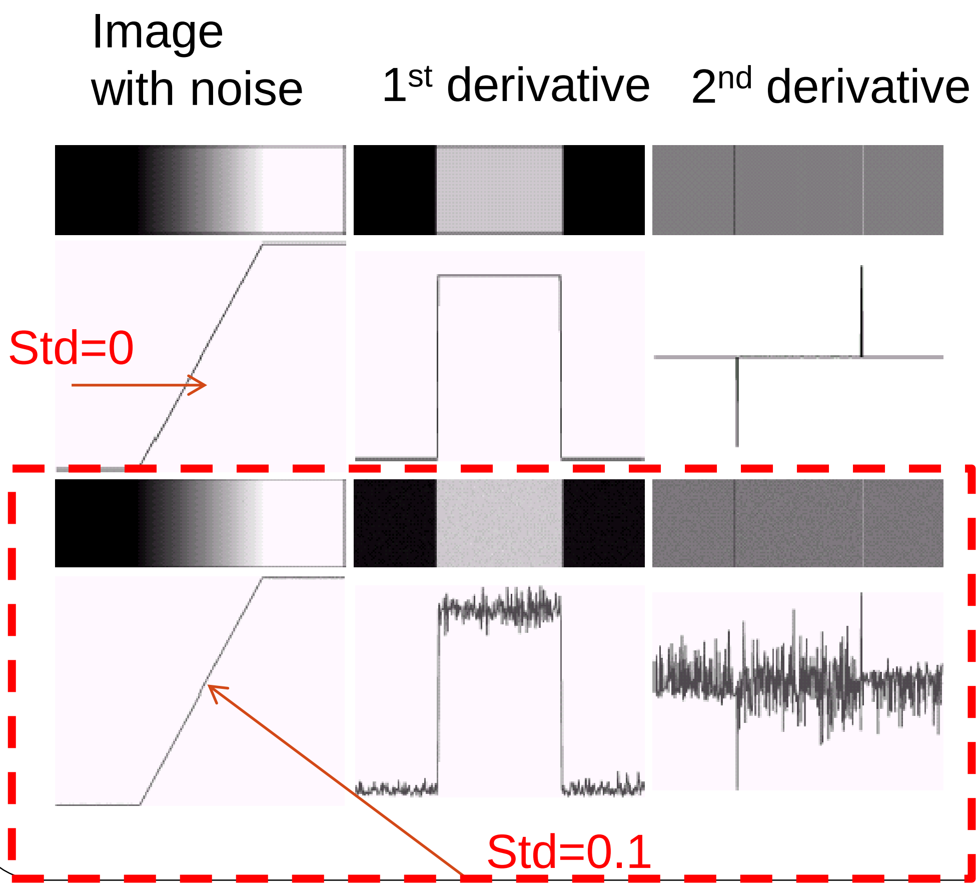
1^{st} derivative

2^{nd} derivative



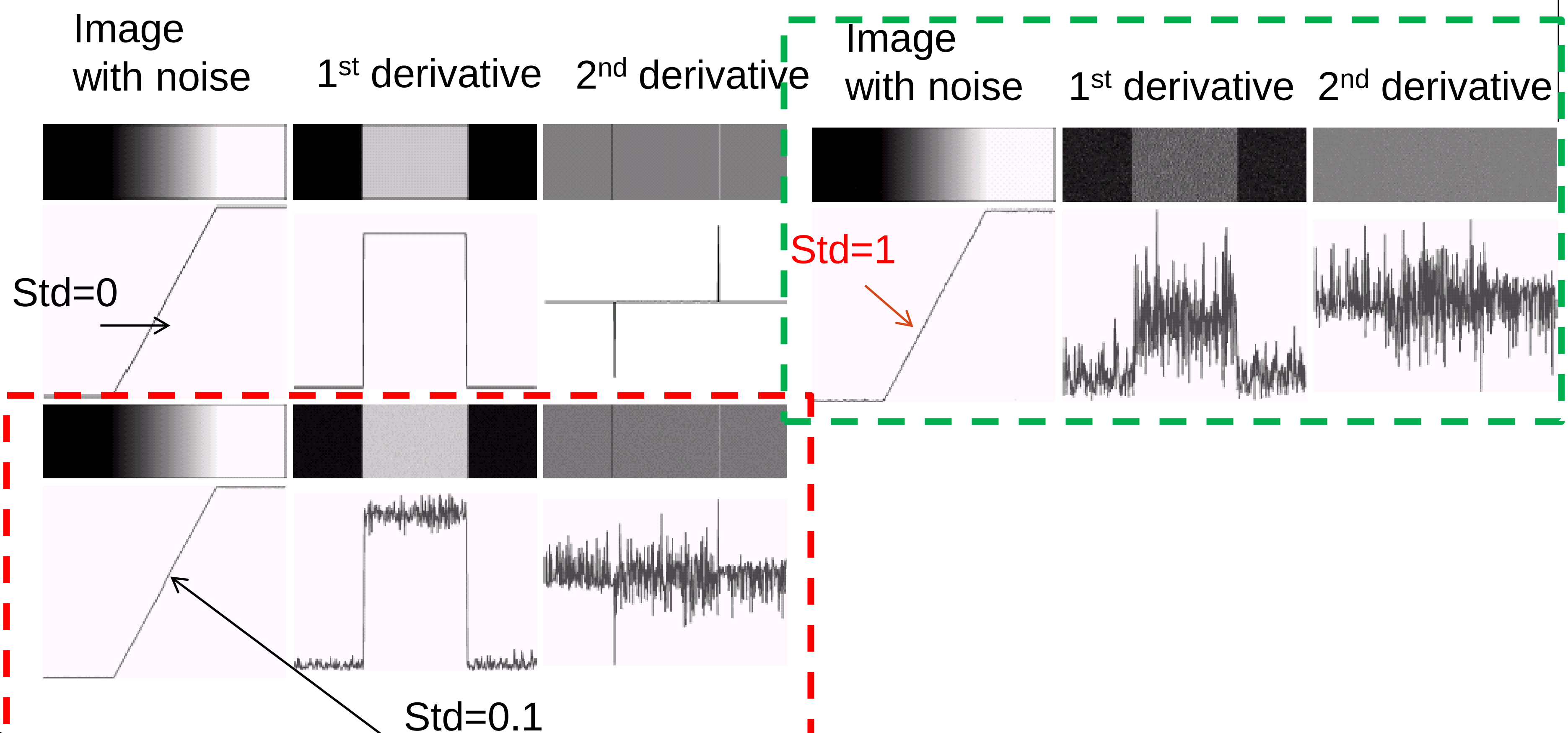
Derivatives & Noise

- Derivative based edge detectors are extremely sensitive to noise



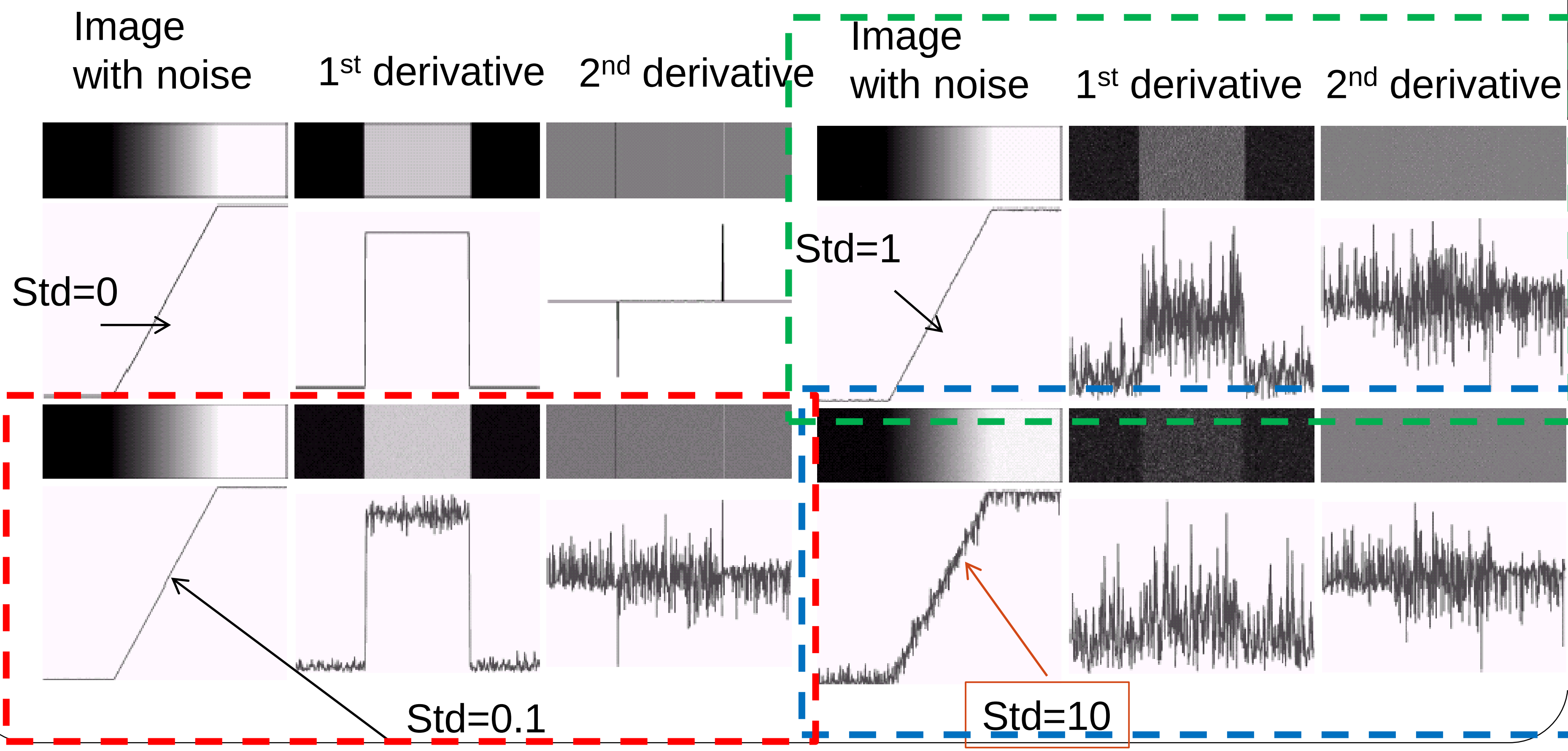
Derivatives & Noise

- Derivative based edge detectors are extremely sensitive to noise



Derivatives & Noise

- Derivative based edge detectors are extremely sensitive to noise



Edge Detection of Noisy Images

- Images can have noise pixels
- Image can also have too many details
- These details are considered as noise
- Ex: Brickwork of the house is considered as noise
- To overcome this, smooth(blur) images prior to edge detection

edge detection with and without smoothing

Apply 5x5 averaging filter before edge detection

without smoothing

with smoothing

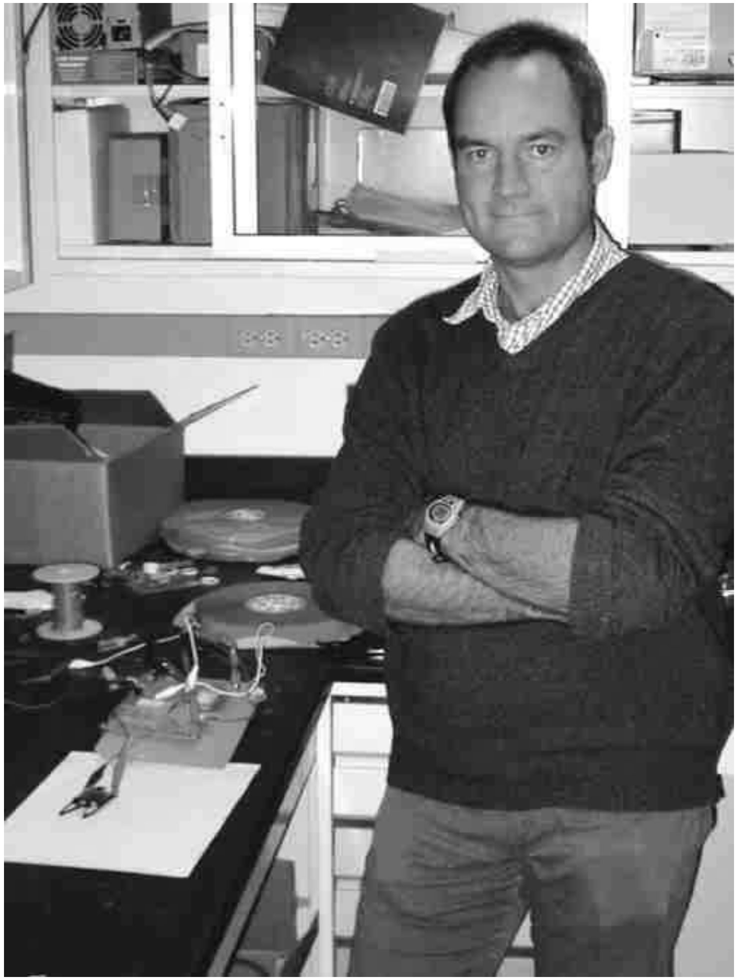
Canny edge detection algorithm

1. Smoothen the image by Gaussian low pass filter
2. Compute the gradient magnitude and angle at each pixel
3. Detect edge points if magnitude of gradient is above a threshold
4. If above threshold, determine edge direction
5. If neighboring pixel is also edge point and has same edge direction then add it as a edge point
6. Link edge points

Canny edge detector

- Low error rate
  - All edges are detected
  - There are no spurious edges
- Edge points are localized
  - Detected edges are close to the true edge
- Single edge point response
  - Detector return only one point for each true edge point
  - Does not identify multiple edge pixels where only single edge point exists

Canny edge detection



Canny edge detection

Edge Linking

- Edge detection characterize edge points
- Breaks may be present in the edges
- Edge linking is required to link edge points with gaps between two edge points
- Linking assembles edge pixels into meaningful edges and region boundaries

Edge Linking

- If edge point are close to each other, they may belong to one edge
- Two properties of edge points are useful for edge linking:
  - the strength (or magnitude) of the detected edge points
  - their directions (determined from gradient directions)
- Adjacent edge points with similar magnitude and direction are linked
- Example: Given an edge pixel with coordinates (x_0,y_0)
- Neighborhood of pixel at (x,y) is also an edge pixel if-

$$\left| M(x, y) - M(x_0, y_0) \right| \le E, \quad E: \text{a nonnegative threshold}$$

$$\left| \alpha(x, y) - \alpha(x_0, y_0) \right| < A, \quad A: \text{a nonegative angle threshold}$$

Edge Linking

- Compute strength and direction of gradient vector for pixel (x,y) on edge

$$\nabla \mathrm{f} = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix}$$

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

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

- S_{xy} is a set of neighboring pixels of (x,y)
- Compute M(a,b) for each pixel of S_{xy}

		(a,b)
	(x,y)	

Edge Linking

- If $|M(a, b) - M(x, y)| <= E$
  and $\alpha(a, b) - \alpha(x, y) < = A$
- then the pixel with coordinates (a, b) in S_{xy} is linked to the pixel at (x, y)

Edge Linking

1. Compute gradient magnitude and angle arrays, $M(x, y)$ and $\alpha(x, y)$ of image $f(x, y)$
2. Form binary image $g(x,y)$ such that
3. if $M(x,y) > E$ and $\alpha(x,y) = A \pm T_A$

   $g(x,y) = 1$

   else $g(x,y) = 0$

T is a threshold, A is specified angle direction and T_A is a band of acceptable range

1	1	0	1	0	1
1	0	0	0	1	0
0	0	0	1	1	1

Edge Linking

1	1	0	1	0	1
1	0	0	0	1	0
0	0	0	1	1	1

4. Assume link value, k
5. Scan each row of g(x,y) and identify 0s
6. If number consecutive zeros < = k
   then fill gaps with ‘1’

k = 2

1	1	1	1	1	1
1	0	0	0	1	0
0	0	0	1	1	1

Edge detection

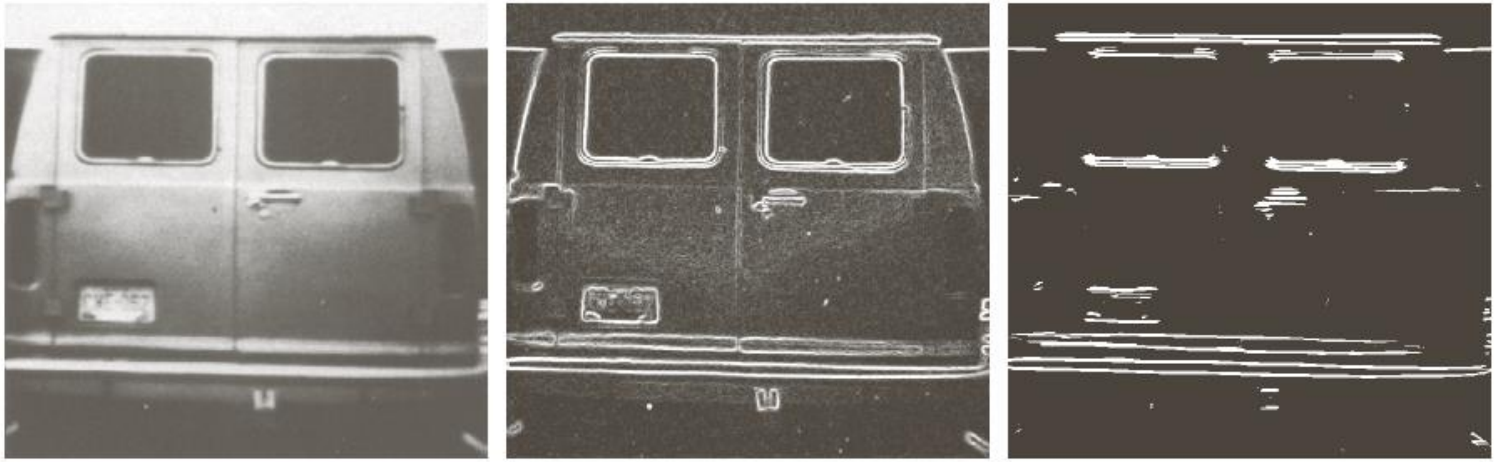
Sobel filter

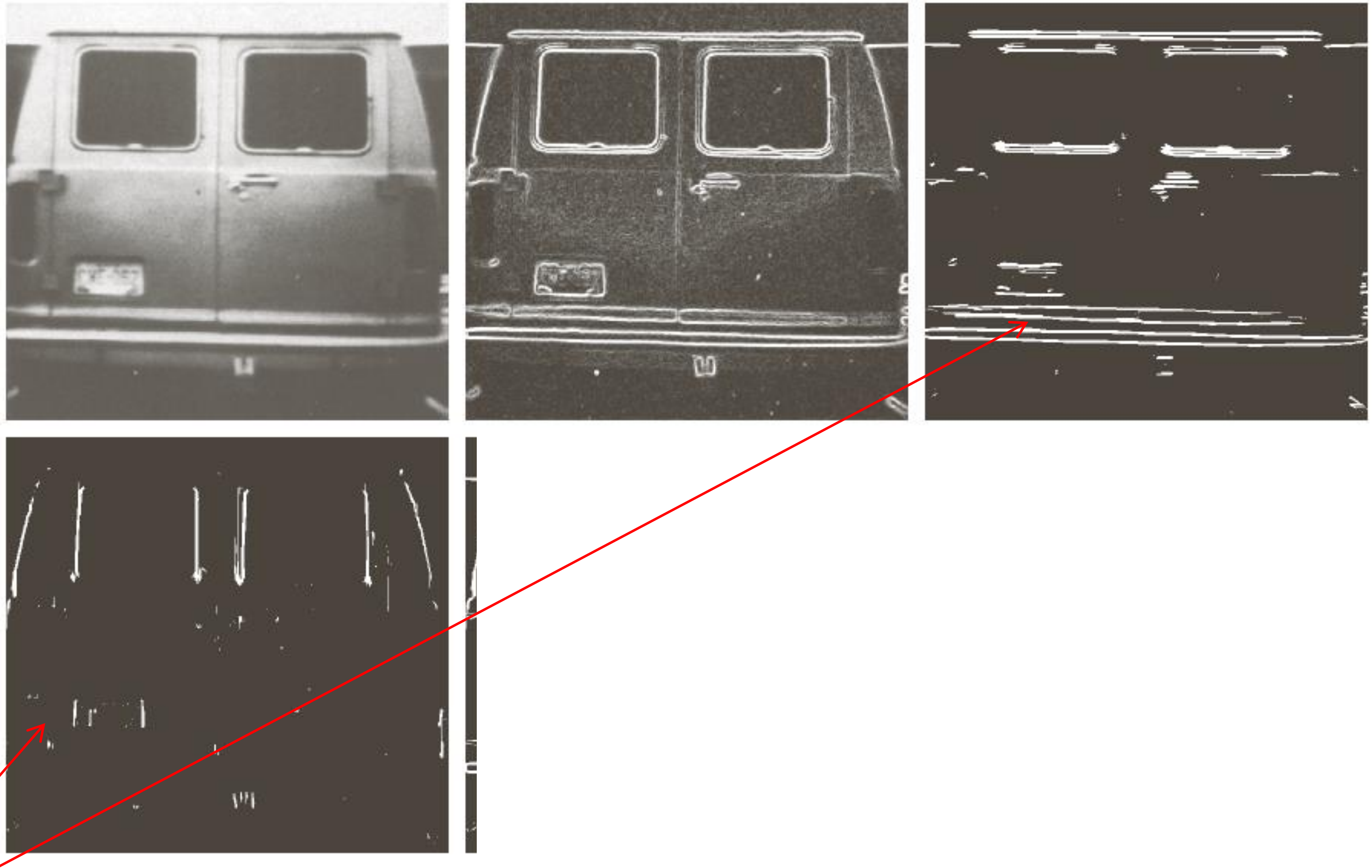
canny

Edge linking to fill gaps

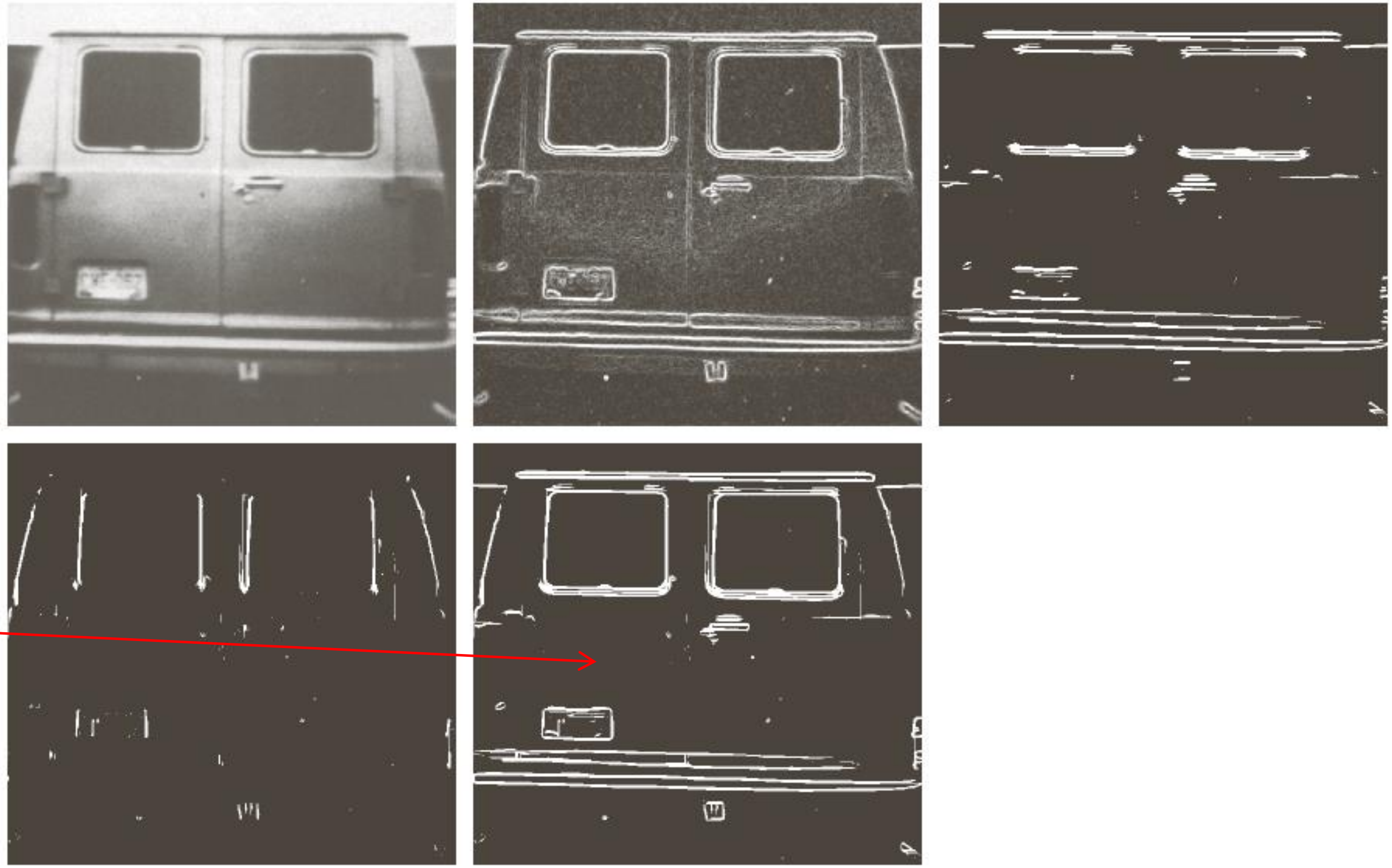
Edge Detection and Linking

Edge Detection and Linking



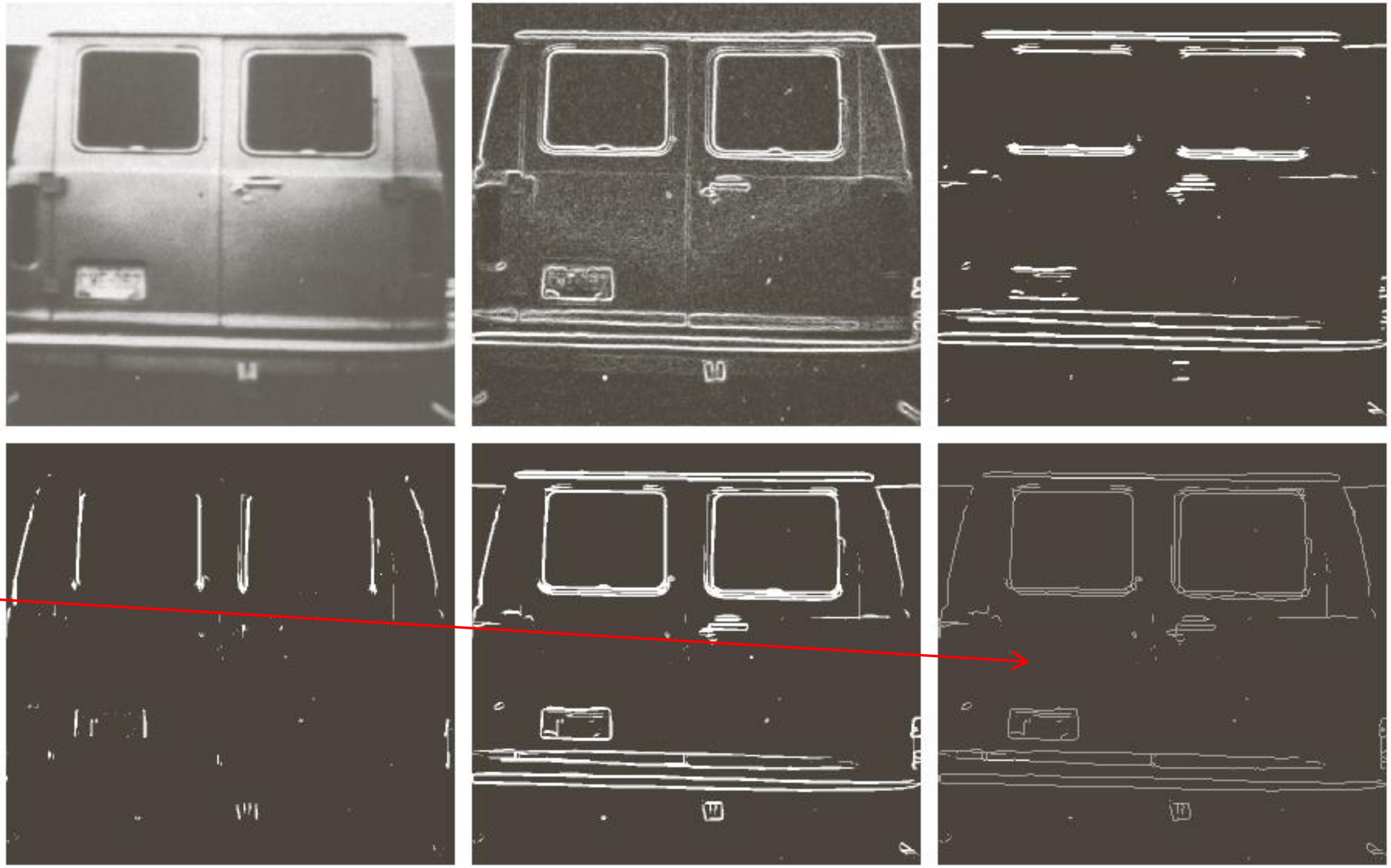


E=30% of max gradient magnitude
A = 90^0
$T_A = 45^0$
K = 25 for horizontal direction and vertical direction



Addition of horizontal and vertical linked edges

Thinning using Image morphology



What is Image segmentation?

- Refers to the process of partitioning an image into groups of pixels which are homogeneous with respect to some criterion.
- Different groups must not intersect with each other, and adjacent groups must be heterogeneous
- Segmentation algorithms are area oriented instead of pixel-oriented
- The result of segmentation is the splitting up of the image into connected areas
- Thus segmentation is concerned with dividing an image into meaningful regions

CLASSIFICATION OF IMAGE-SEGMENTATION TECHNIQUES

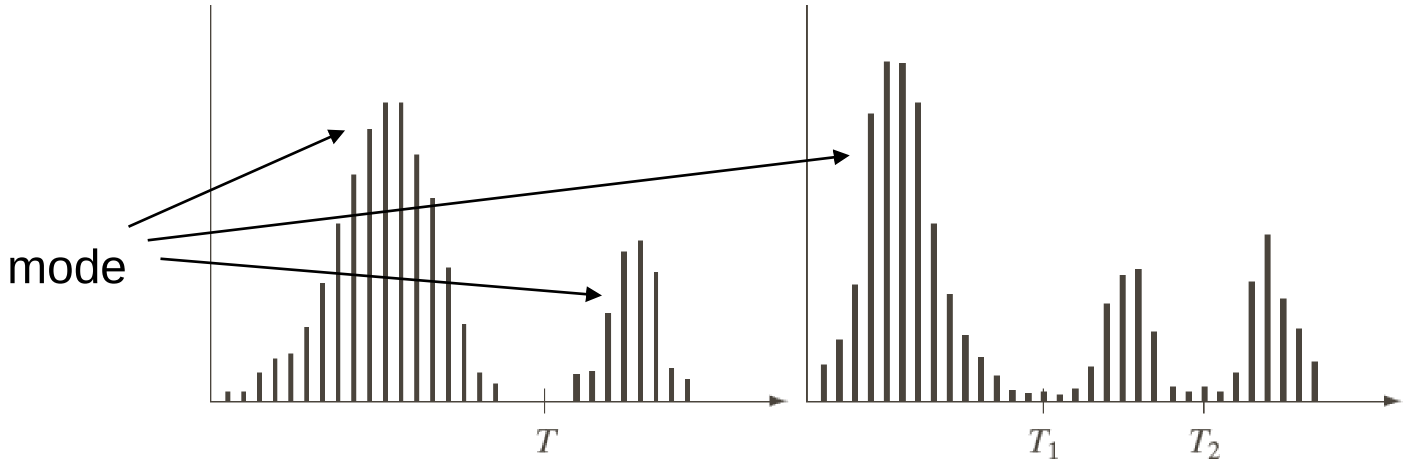
- Local Segmentation
  - Deals with segmenting sub-images which are small windows on a whole image
  - The number of pixels available to local segmentation is much lower than global segmentation
- Global Segmentation
  - is concerned with segmenting a whole image
  - deals mostly with segments consisting of a relatively large number of pixels
  - This makes estimated parameter values for global segments more robust

Approaches for segmentation

1. Region approach
   - Regions in an image are a group of connected pixels with similar properties
   - Each pixel is assigned to a particular object or region
2. Boundary approach
   - locates boundaries that exist between the regions.
3. Edge approach
   - edges are identified first, and then they are linked together to form required boundaries

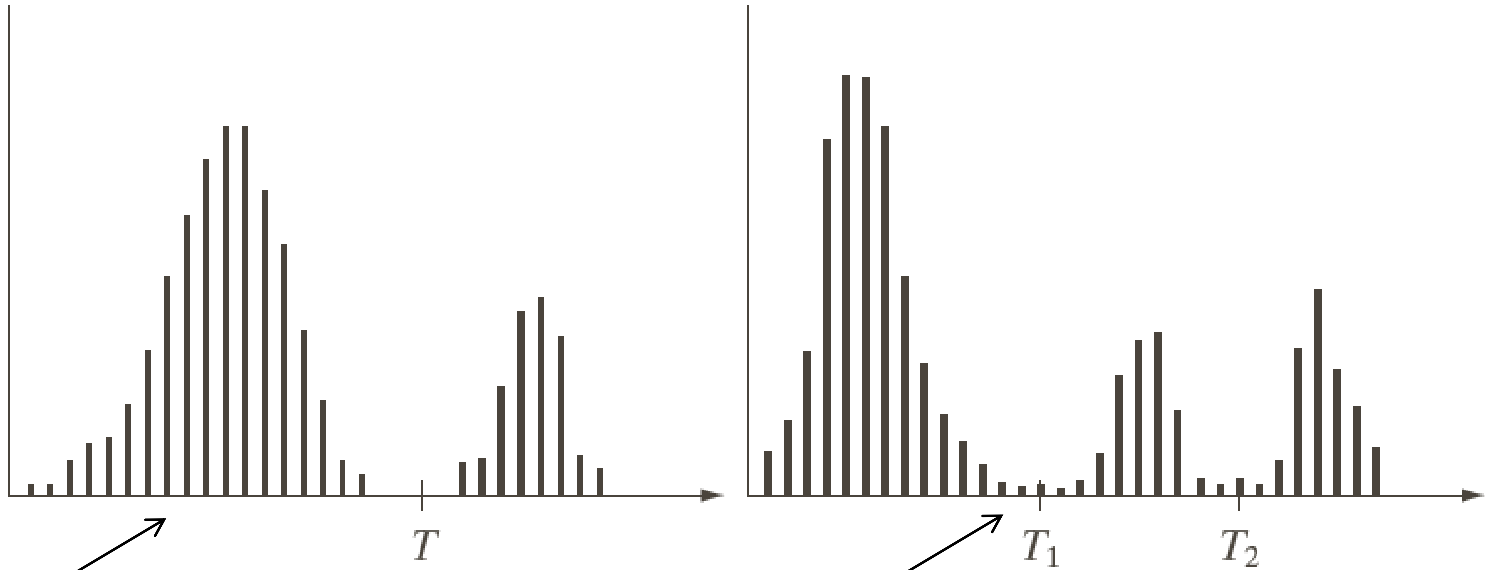
Intensity Thresholding for segmentation

- Based on histogram
- Depends on width and depth of the valleys separating the modes in histogram
- Valleys depend on
  - Separation between peaks
  - Relative size of object against background
  - Noise contents
  - Uniformity of the illumination source



Thresholding

histograms



Two dominant modes

Three dominant modes

For thresholding,

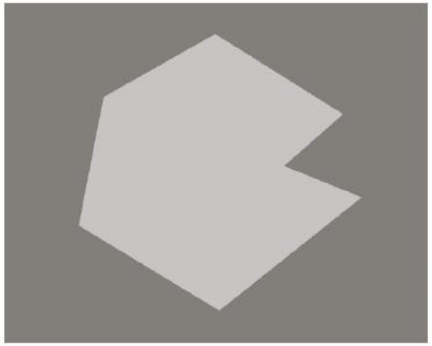
$$g(x,y)=\begin{cases} 1\, if\;\; f(x,y)>T \\ 0\, if\;\; f(x,y)\le T \end{cases}$$

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

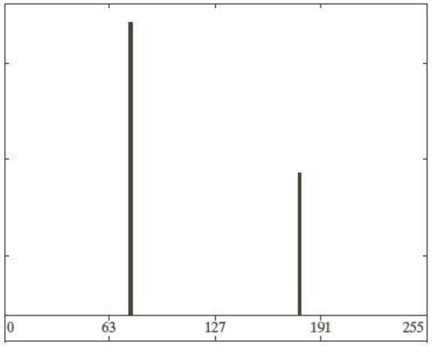
Effect of noise in image thresholding

Gaussian noise, std = 0

Image



histograms



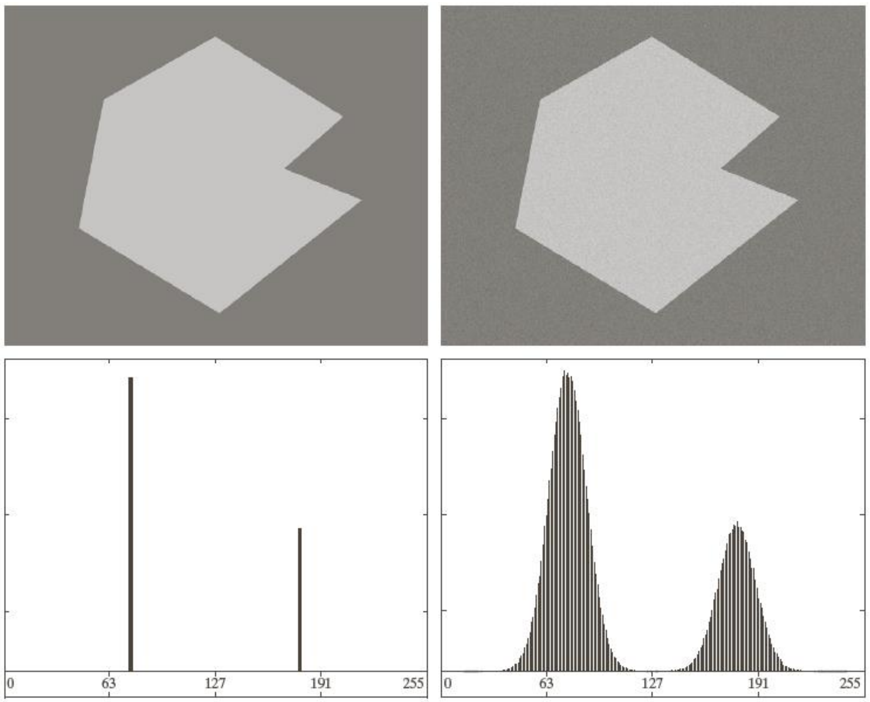
Threshold can be set as two modes can be separated easily

Effect of noise in image thresholding

Gaussian noise, std = 0 Gaussian noise, std 10

Image

histograms



Threshold can be set as two modes can be separated easily

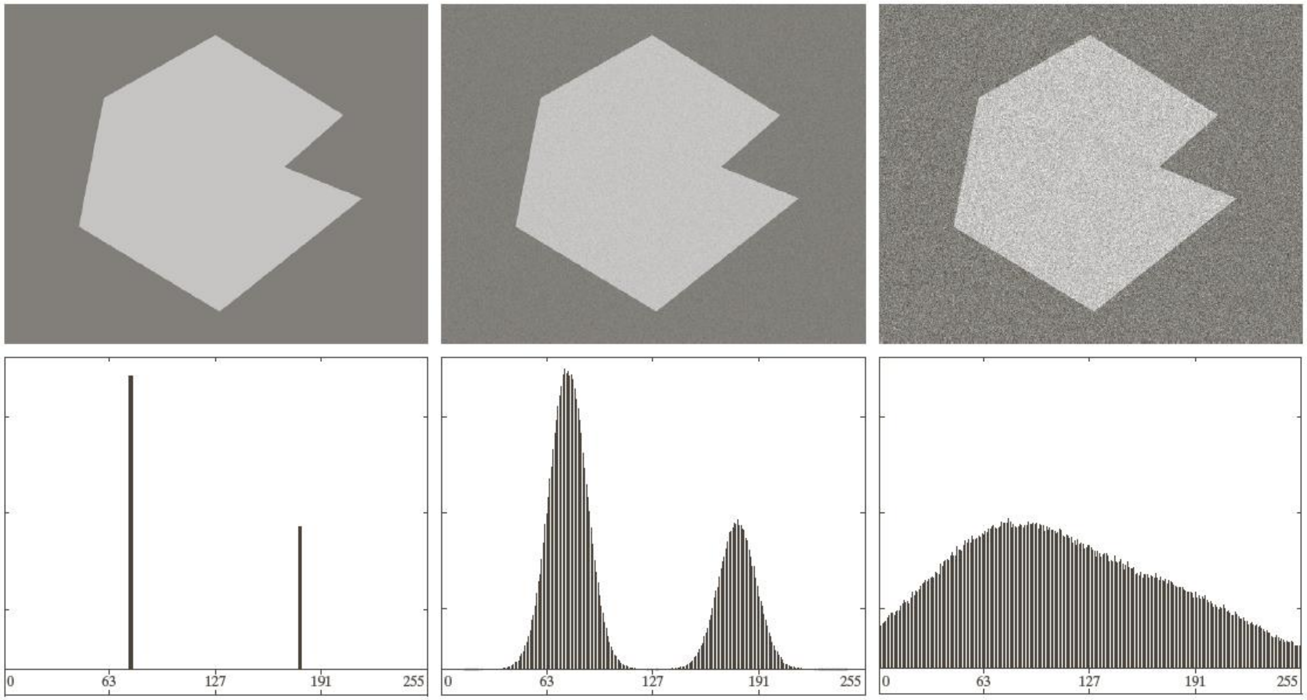
Threshold can be set as two modes can be separated though image is noisy

Effect of noise in image thresholding

Gaussian noise, std = 0 Gaussian noise, std 10 Gaussian noise std, 50

Image

histograms



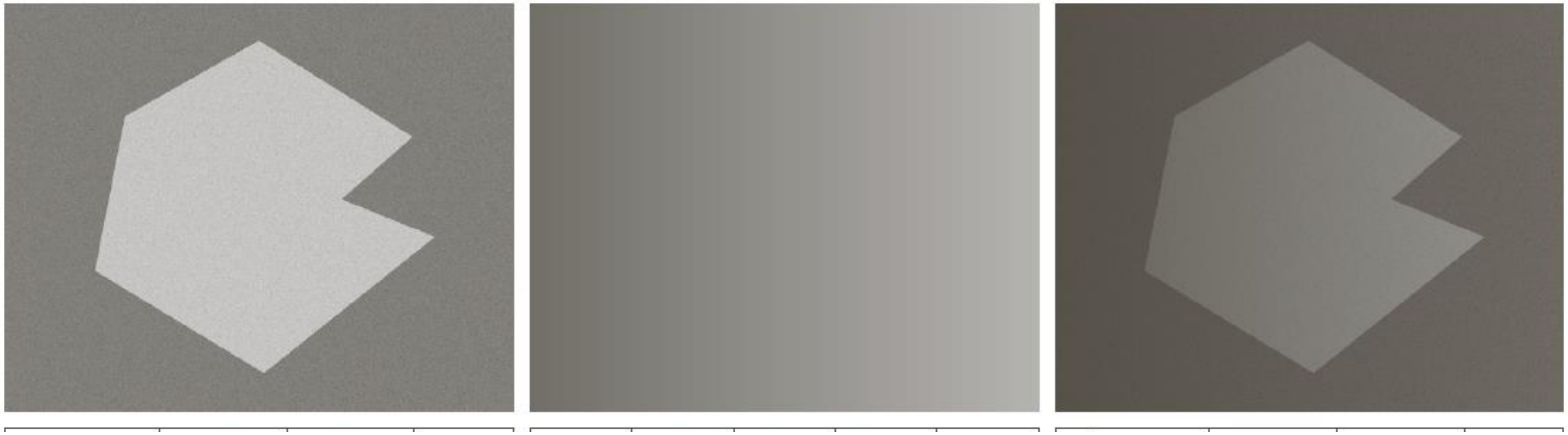
Threshold can be set as two modes can be separated easily

Threshold can be set as two modes can be separated though image is noisy

Thresholding is not possible as modes have merged

Role of illumination in image thresholding

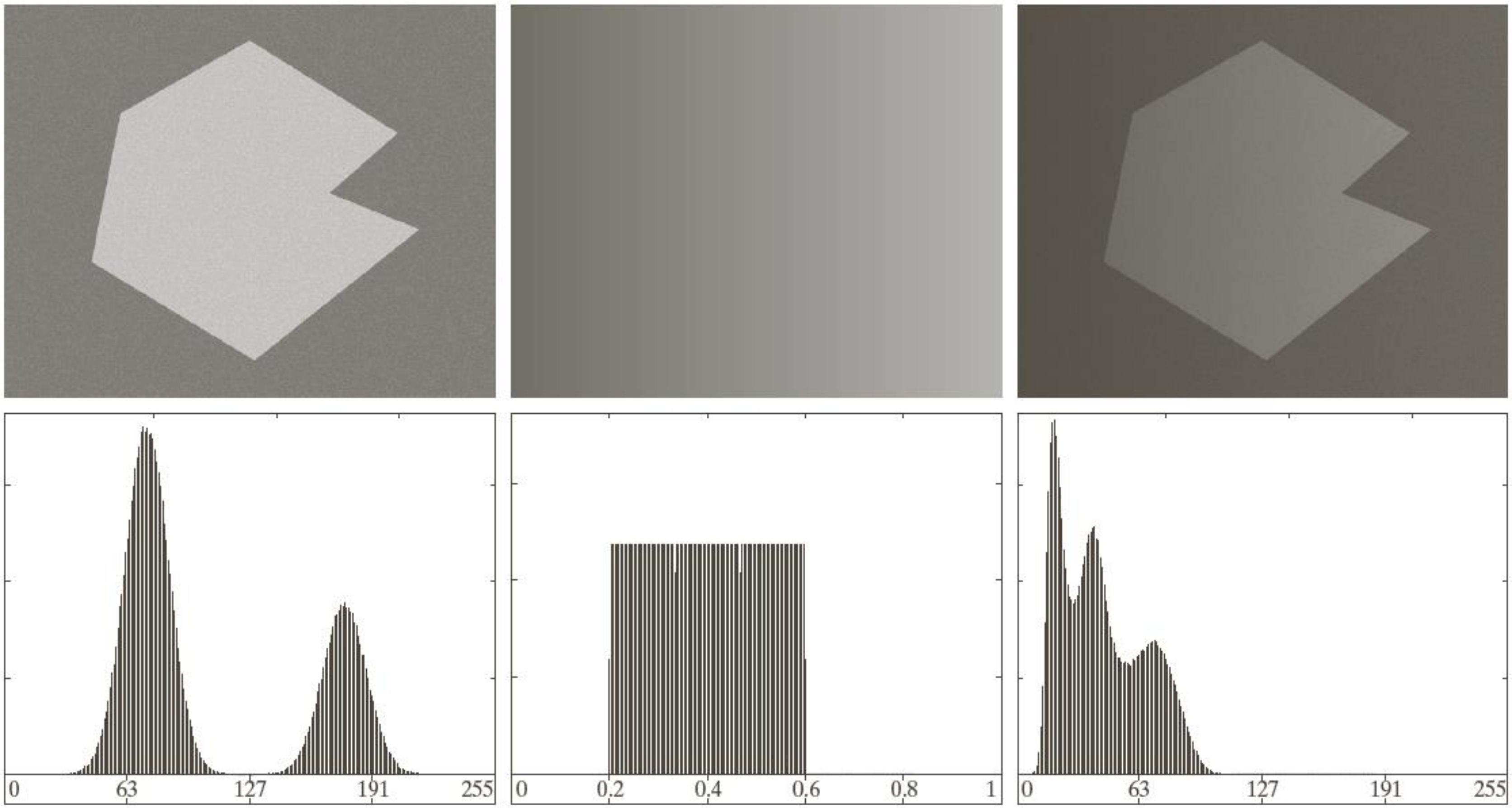
Image



Role of illumination in image thresholding

Image

histog
rams



Threshold can be set as two modes can be separated

Thresholding is not possible as modes have merged

Thresholding is not possible as modes have merged

Basic Global Thresholding

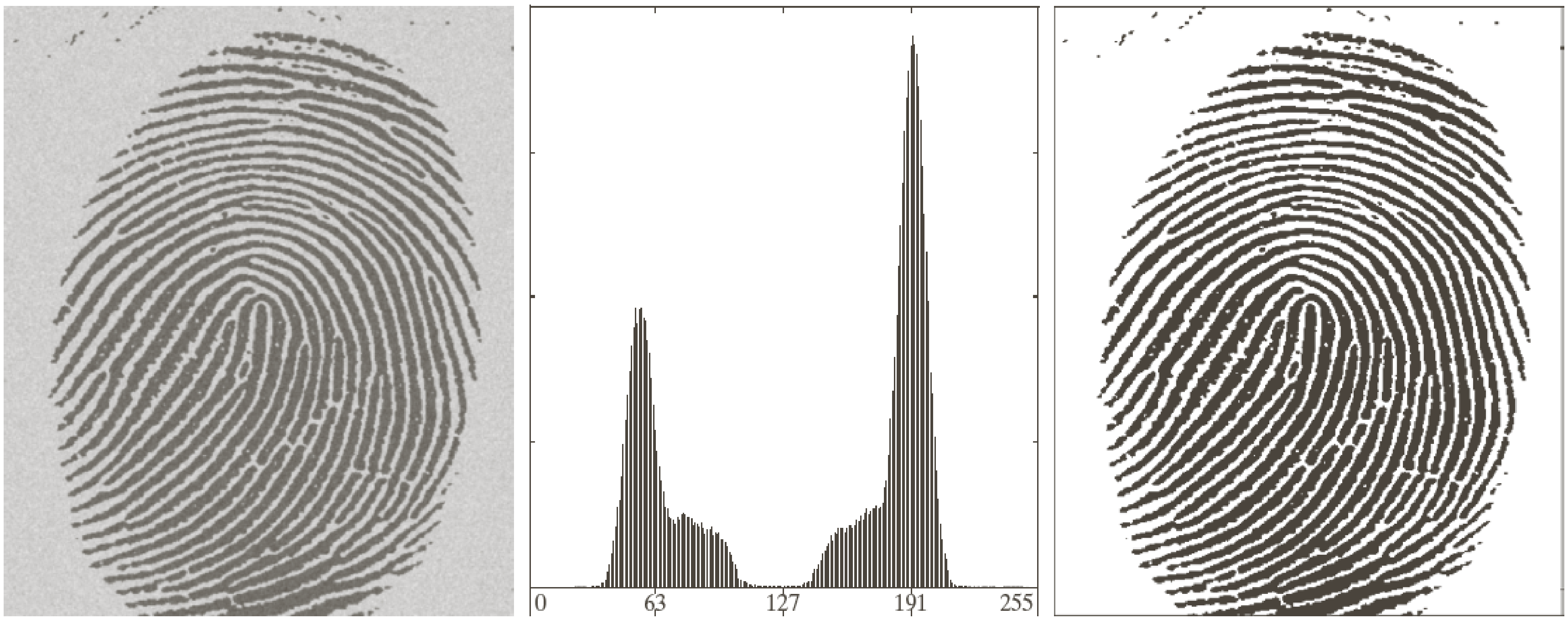
1. Select an initial estimate for the global threshold, T
2. Segment image using T to form two groups, G_1 and G_2.
3. Compute mean intensity m_1 and m_2 for pixels in each group
4. Compute new threshold

   $T_{new}=(m_1+m_2)/2$
5. Determine $T- T_{new}$
6. If $T - T_{new} > \Delta$

   then $T = T_{new}$ go to step 5

   else end
5. Repeat step 2 through 7 until $T - T_{new} < \Delta$ (predefined)
6. Predefined difference controls number of iterations
7. Less is the value of Δ, more is the number of iterations

Segmentation based on global thresholding



Optimum global thresholding: Otsu's method

- Let {0,1,2,…,L-1} be L intensity levels and size of image is MxN
- For histogram, n_i is number of pixels with intensity, i
- For normalized histogram, $p_i=n_i/MN$
- Ex: 3-bit image has n = {4,2,8,1,0, 5,2,3}
  for intensity levels, i = {0,1,2,3,4,5,6,7}
- p = {4,2,8,1,0, 5,2,3}/25

- Select threshold, T(k)=k, 0 < k < L-1 to generate classes C_1 and C_2
- Ex: T(1) = 1

Optimum global thresholding: Otsu's method

Normalized Probability, p = {4,2,8,1,0, 5,2,3}/25
For threshold, T(k), probability of C_1 is $P_1(k)$
And for C_2 probability is $P_2(k)$

$$P_1(k) = \sum_{i=0}^{k} p_i$$

$$P_2(k) = \sum_{i=k+1}^{L-1} p_i = 1 - P_1(k)$$

$P_1(1) = (4+2)/25$
$= 6/25$

$P_2(1) = (8+1+0+5+2+3)/25$
$= 19/25$

$$m_1(k) = \frac{1}{P_1(k)} \sum_{i=0}^{k} i p_i$$

$$m_2(k) = \frac{1}{P_2(k)} \sum_{i=k+1}^{L-1} i p_i$$

$m_1(1) = (25/6)\{(0 \times 4/25) + (1 \times 2/25)\}$
$= (1/3)$

$m_2(1) = (25/19)\{(2 \times 8/25) + (3 \times 1/25)$
$+ (5 \times 5/25) + (6 \times 2/25)$
$+ (7 \times 3/25)\}$
$= (77/19)$

Optimum global thresholding: Otsu's method

p = {4,2,8,1,0, 5,2,3}/25

Average intensity upto k

$$m(k) = \sum_{i=0}^{k} i p_i$$

m(1) = (0 × 4/25) +(1 × 2/25) = (2/25)

Global mean of entire image

$$m_G = \sum_{i=0}^{L-1} i p_i$$

m_G = (0 × 4/25) +(1 × 2/25)+…+ (7 × 3/25) = (79/25)

Optimum global thresholding: Otsu's method

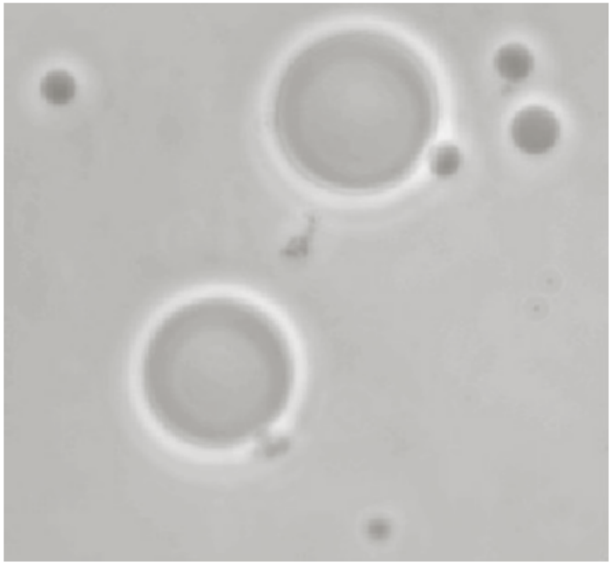
Global variance
$$\sigma_G^2 = \sum_{i=0}^{L-1} (i - m_G)^2 p_i$$

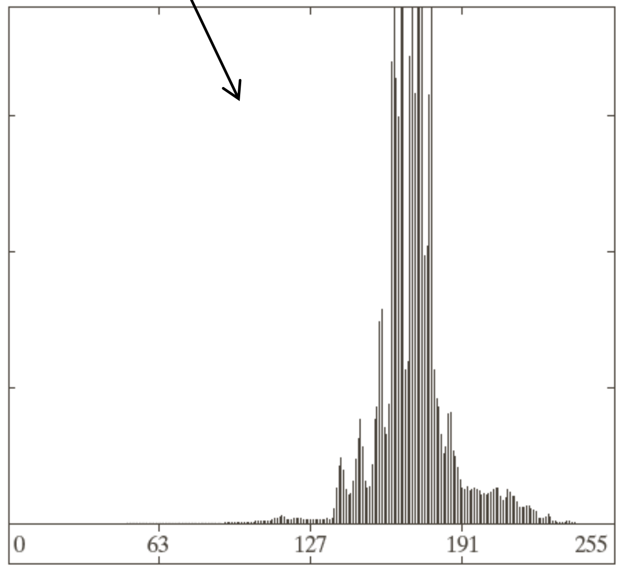
Between the class variance
$$\sigma_B^2(k) = P_1(k) P_2(k) (m_1 - m_2)^2$$

Metric for thresholding
$$\eta = \frac{\sigma_B^2}{\sigma_G^2}$$

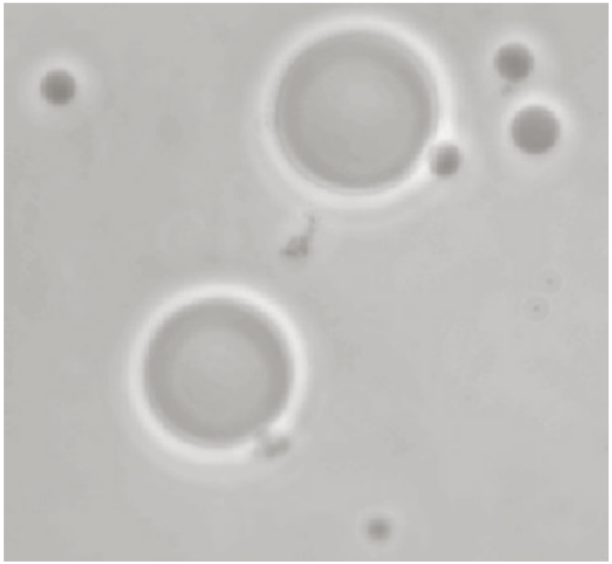
- Large value of metric represents large separability
- Choose k which provides maximum value of metric for thresholding

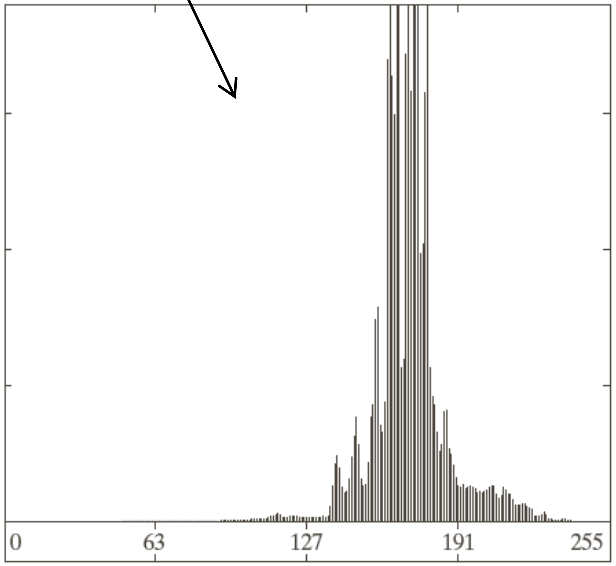
Does not have distinct valleys





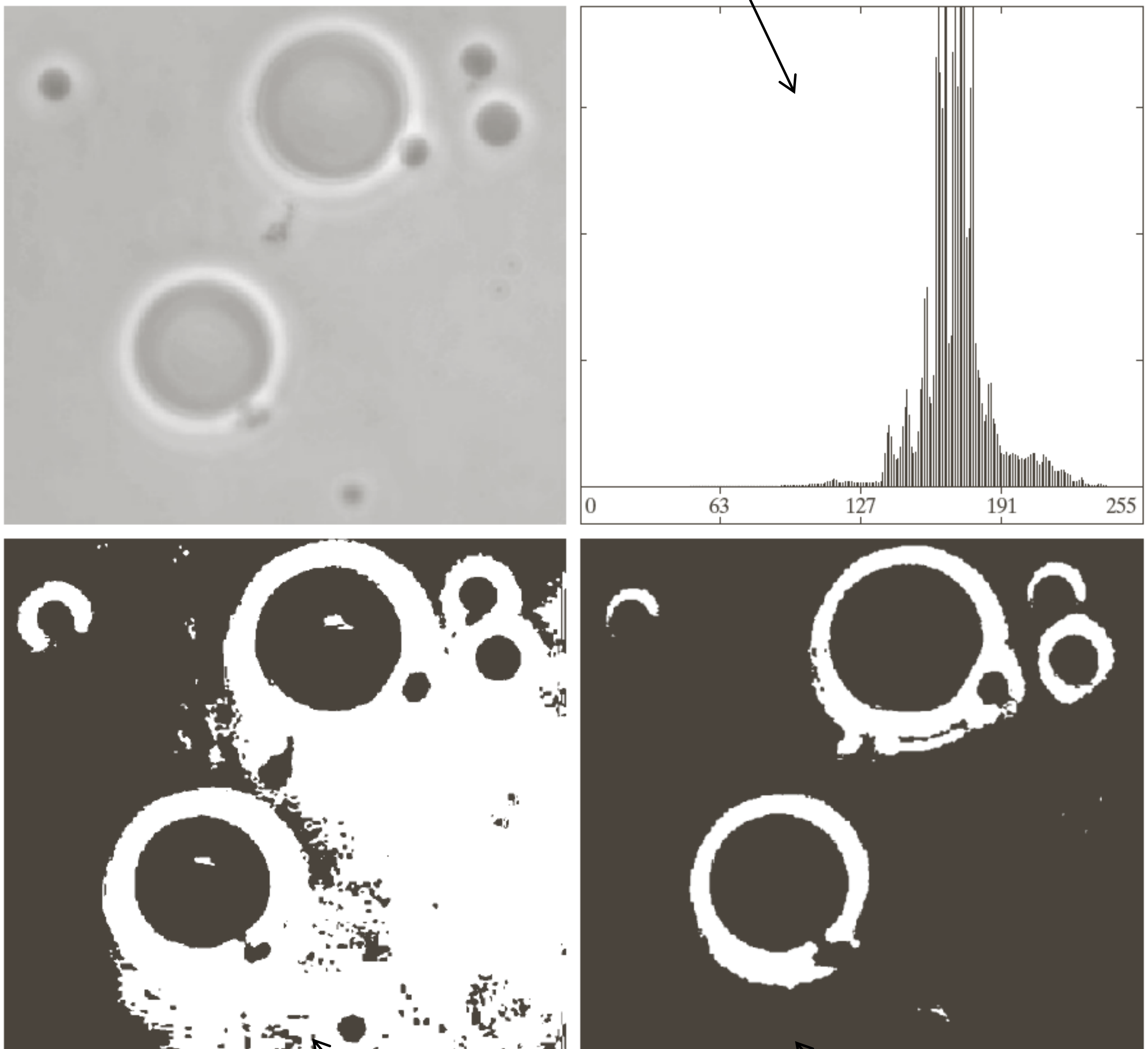
Does not have distinct valleys





Global thresholding, T=169

Does not have distinct valleys



global thresholding T=169

Otsu's method, T=181
$\eta = 0.467$

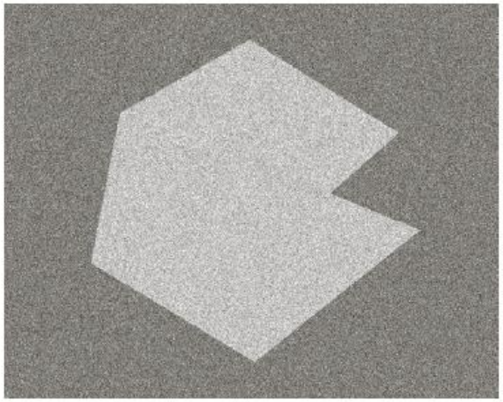
Image smoothing to before Otsu's thresholding

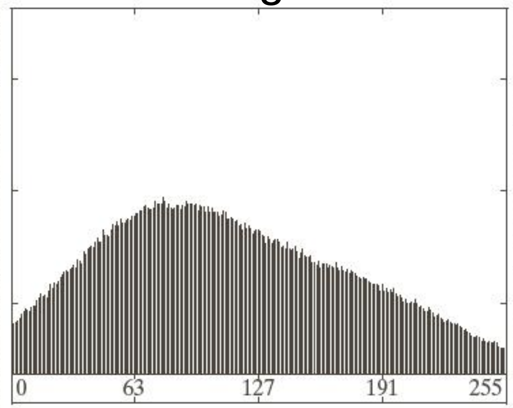
Image smoothing to before Otsu’s thresholding

Histogram

After threshoding

Noisy image





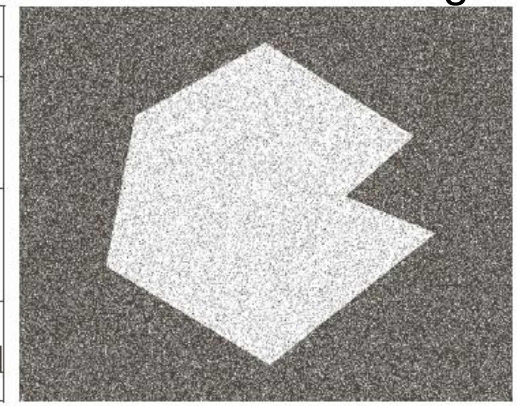


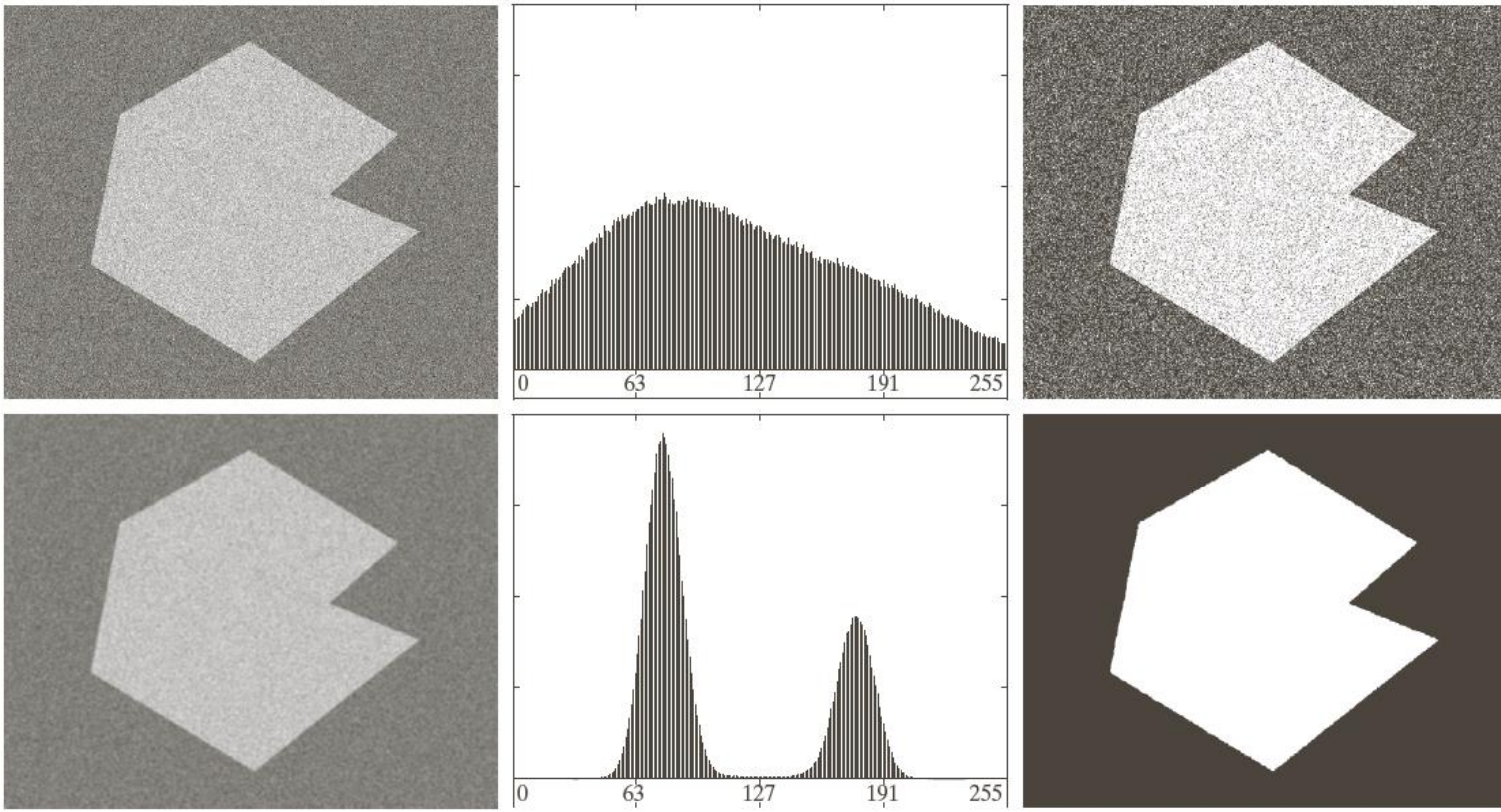
Image smoothing to improve global thresholding

Histogram

After threshoding

Noisy image

Image after averaging by 5x5 filter



Region based segmentation

What is a Region

- A group of connected pixels with similar properties
- Connected pixels correspond to objects in a scene
- Region of connected pixels correspond to objects or parts of an object

Region-based Approach

- Predicate is a condition to define region
- Pixels satisfying the condition of predicate are marked
- Group of these pixels defines an object
- Predicates are based on

  1. Value similarity

     -Gray value differences

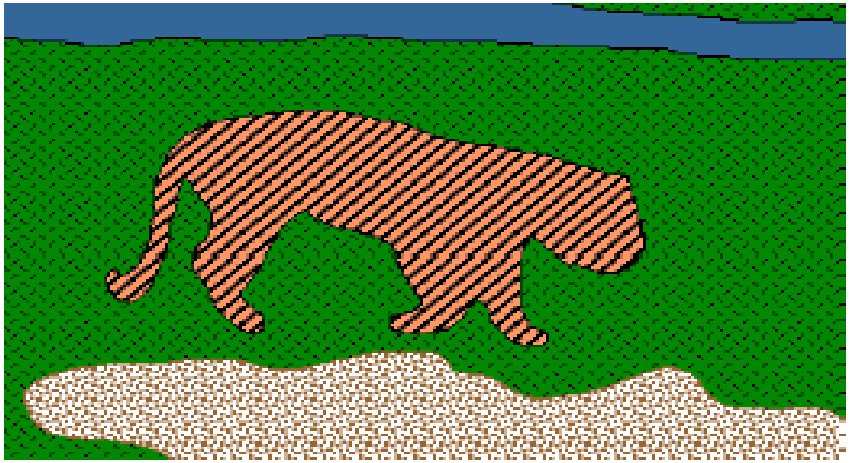
     -Gray value variance

  2. Spatial Proximity

     Euclidean distance

Example: Region Based Segmentation

Original Image



Segmented image

Methods Region Based Segmentation

- Region
  - Growing
  - Split
  - Merge
  - Split and Merge

Region Growing

- Region growing requires a seed to begin with
- Ideally, the seed would be a region, but it could be a single pixel
- A new segment is grown from the seed
- Resultant segment is then removed from the process
- A new seed is chosen from the remaining pixels
- This continues until all pixels have been allocated to a segment
- The resulting segmentation could
  - depend on the initial seed chosen
  - order in which neighboring pixels are examined

Region Growing

- Region growing offers several advantages over conventional segmentation techniques
- The borders of regions found by region growing are perfectly thin and connected
- The algorithm is also very stable with respect to noise
- based on predicate
  - threshold
  - constant
  - variance
  - color etc…
- first step is to choose a seed

Region growing Predicate: Example

Predicate: Difference between two consequtive pixels <= 5

10	10	10	10	10	10	10
10	10	10	69	70	10	10
59	10	60	64	59	56	60
10	59	10	<u>60</u>	70	10	62
10	60	59	65	67	10	65
10	10	10	10	10	10	10
10	10	10	10	10	10	10

10	10	10	10	10	10	10
10	10	10	69	70	10	10
59	10	60	64	59	56	60
10	59	10	<u>60</u>	70	10	62
10	60	59	65	67	10	65
10	10	10	10	10	10	10
10	10	10	10	10	10	10

If location of seed changes then regions change

Region growing

Multiple seeds are used

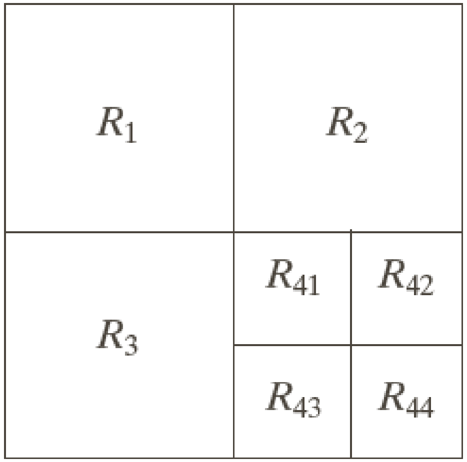
Region Splitting

- Region growing starts from a set of seed points
- Opposite approach to region growing is region splitting
- It is a top-down approach and it starts with the assumption that the entire image is homogeneous
- If this is not true, the image is split into four sub images
- Splitting procedure is repeated recursively until we split the image into homogeneous regions

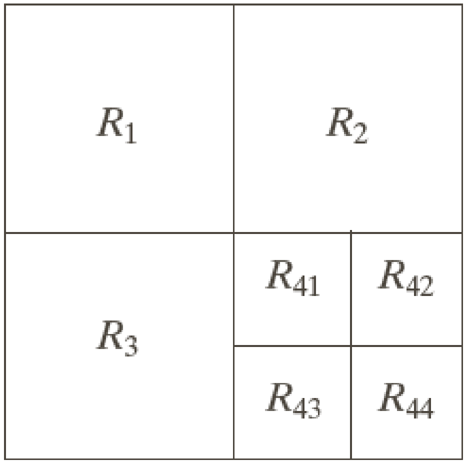
Region Splitting

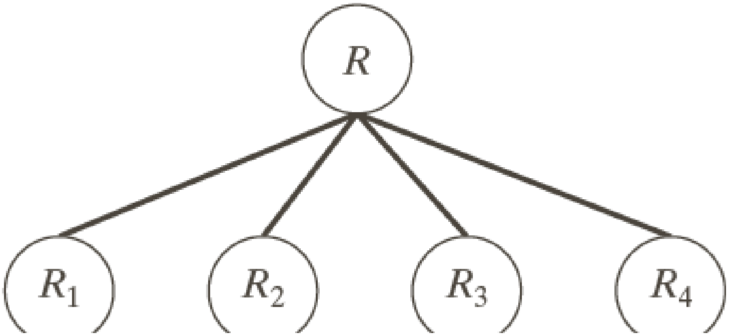
- If the original image has size M x N,
- Then $M = 2^n$ and $N = 2^n$
- Procedure of splitting is recursive
- Produces an image representation that can be described by a tree/subtree with four nodes
- Tree is called a Quadtree.

Region Splitting

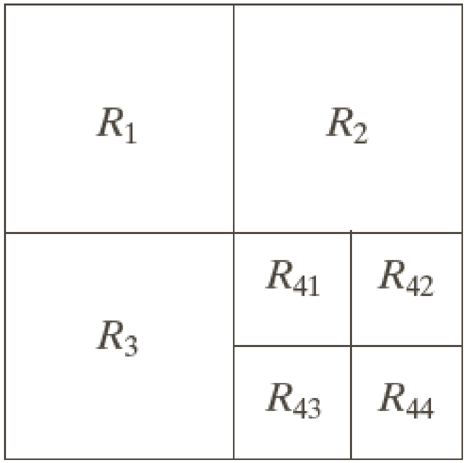


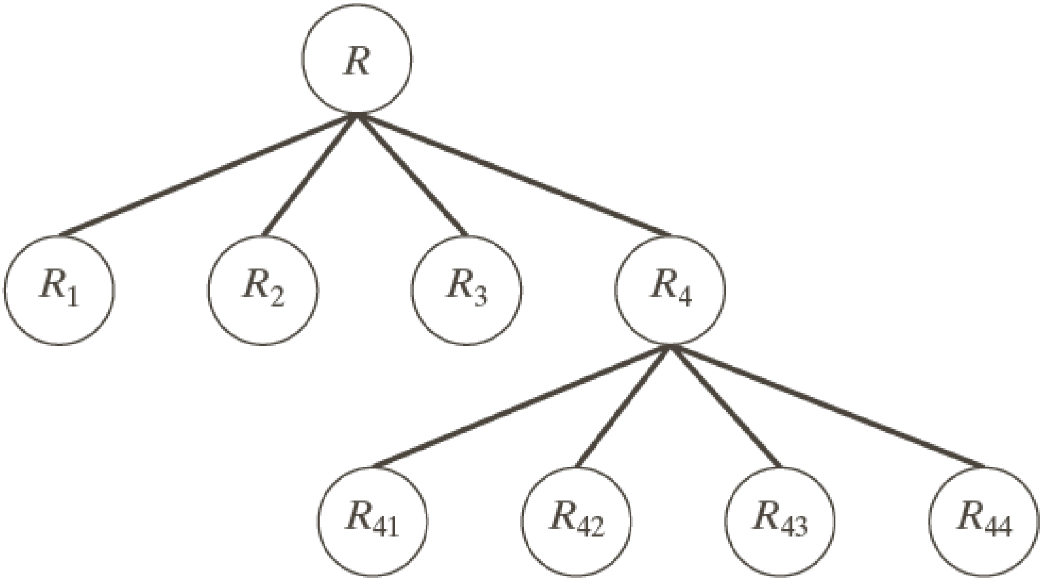
Region Splitting





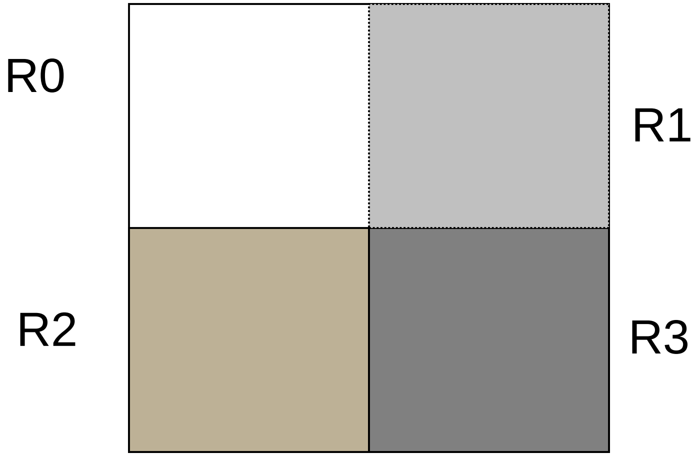
Region Splitting



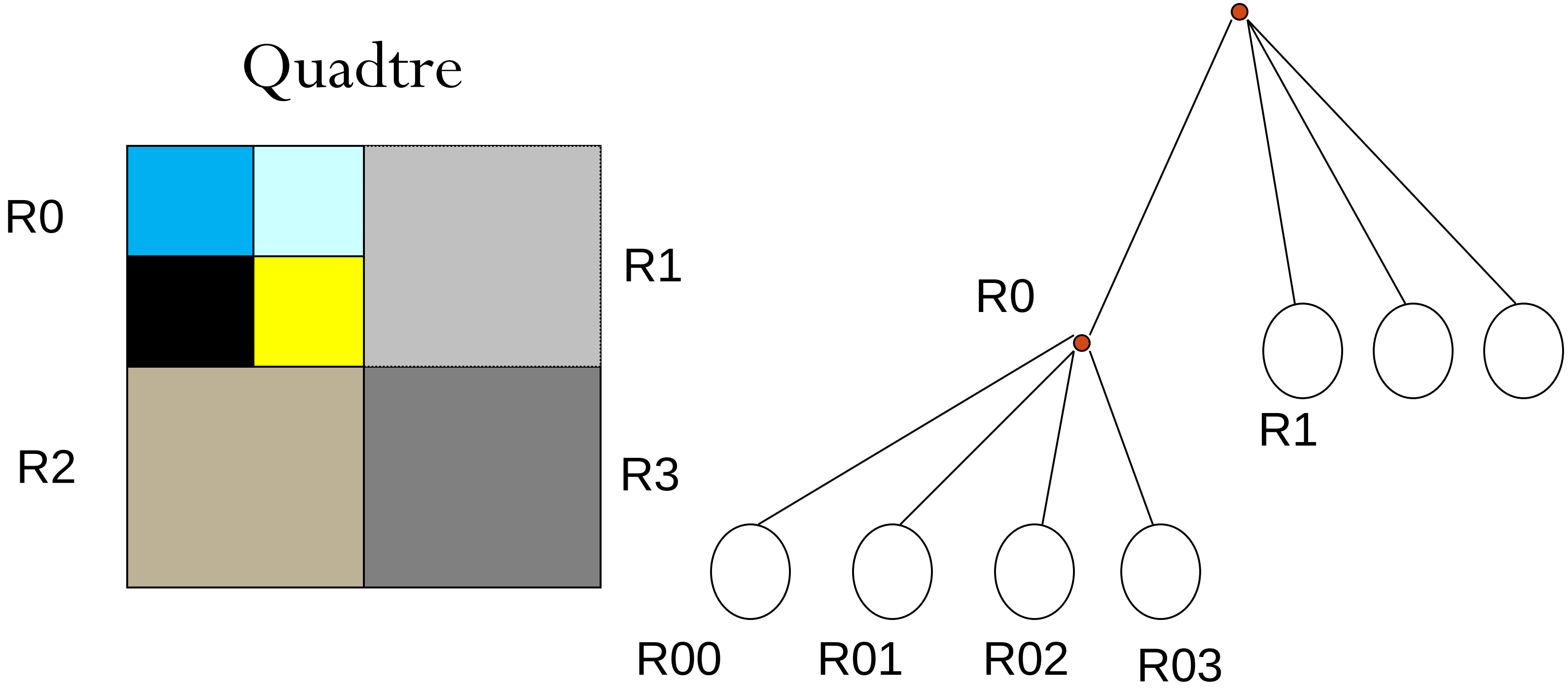


Region Splitting: Example

Quadtre



Region Splitting: Example



Region Splitting Example

Minimum size after splitting 2x2

10	**11**	**10**	**11**	**15**	**16**	**15**	**15**
10	10	10	11	15	15	16	15
11	11	10	11	16	15	15	15
11	10	11	10	16	15	16	15
17	18	17	18	9	8	3	4
17	18	18	17	9	8	3	4
18	17	18	17	12	11	21	21
17	18	17	18	11	11	20	20

Condition:

Pixel difference between two consecutive pixels (Separation Threshold) <=2

All the pixels do not satisfy the condition

Region Splitting Example

Minimum size after splitting 2x2

R_0 R_1 R_2 R_3

10	**11**	**10**	**11**	**15**	**16**	**15**	**15**
10	10	10	11	15	15	16	15
11	11	10	11	16	15	15	15
11	10	11	10	16	15	16	15
17	18	17	18	9	8	3	4
17	18	18	17	9	8	3	4
18	17	18	17	12	11	21	21
17	18	17	18	11	11	20	20

Condition:

Pixel difference between two consecutive pixels (Separation Threshold) <=2

Pixels of R_3 do not satisfy the condition

Region Splitting Example

Minimum size after splitting 2x2

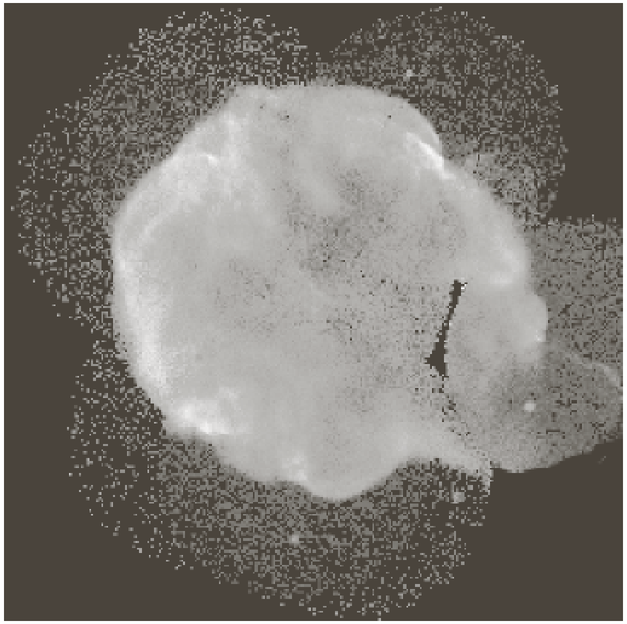
10	**11**	**10**	**11**	**15**	**16**	**15**	**15**
10	10	10	11	15	15	16	15
11	11	10	11	16	15	15	15
11	10	11	10	16	15	16	15
17	18	17	18	9	8	3	4
17	18	18	17	9	8	3	4
18	17	18	17	12	11	21	41
17	18	17	18	11	11	10	20

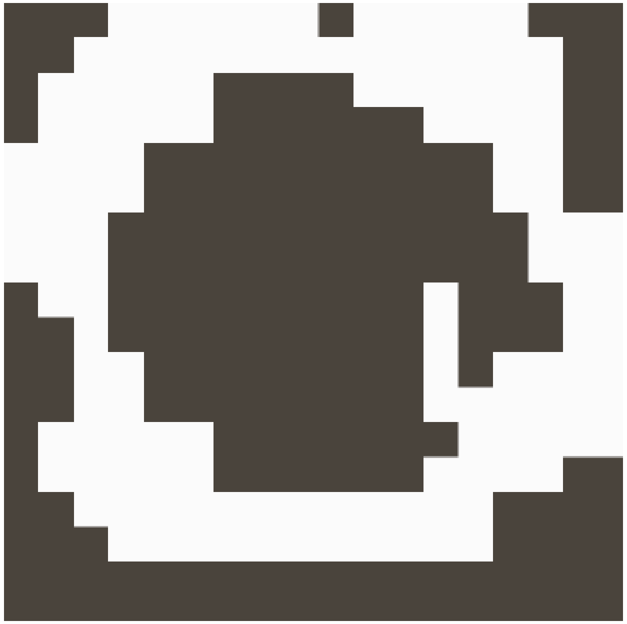
Condition:
Pixel difference between two consecutive pixels (Separation Threshold) <=2

Results – Region Split

Splitting Ex: Condition: $\sigma > 10$ & $0 < m < 125$

σ is variance and m is mean

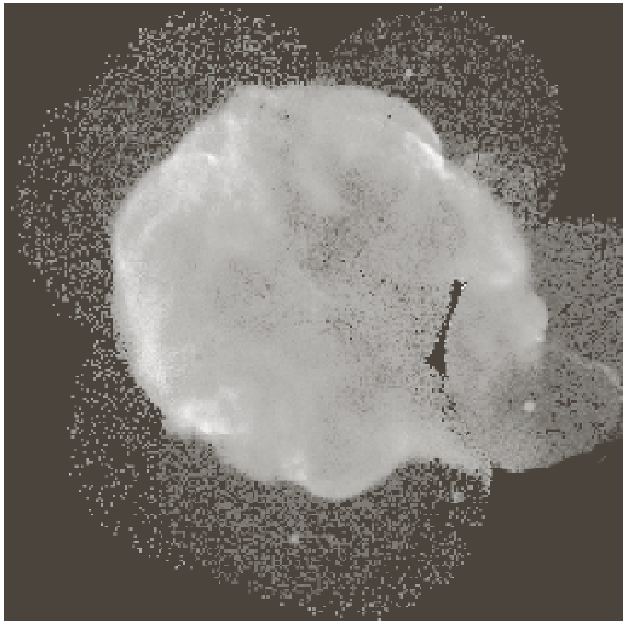


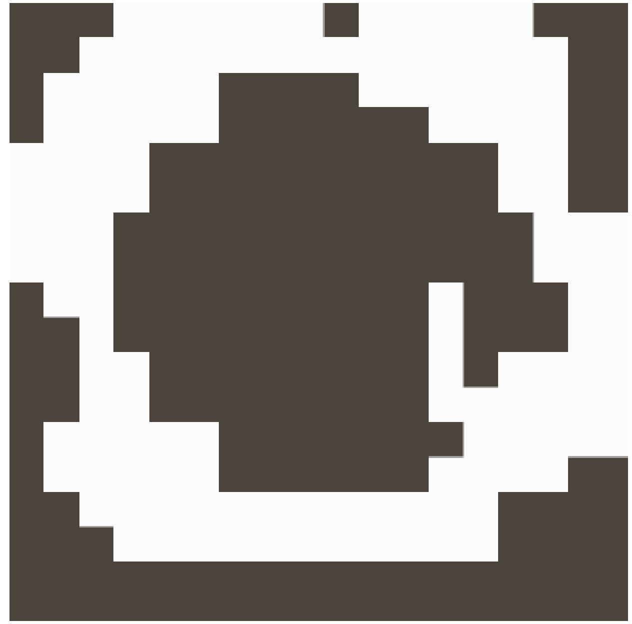


Min size 32x32

Splitting Ex: Condition: $\sigma > 10 \;\&\; 0 < m < 125$

σ is variance and m is mean



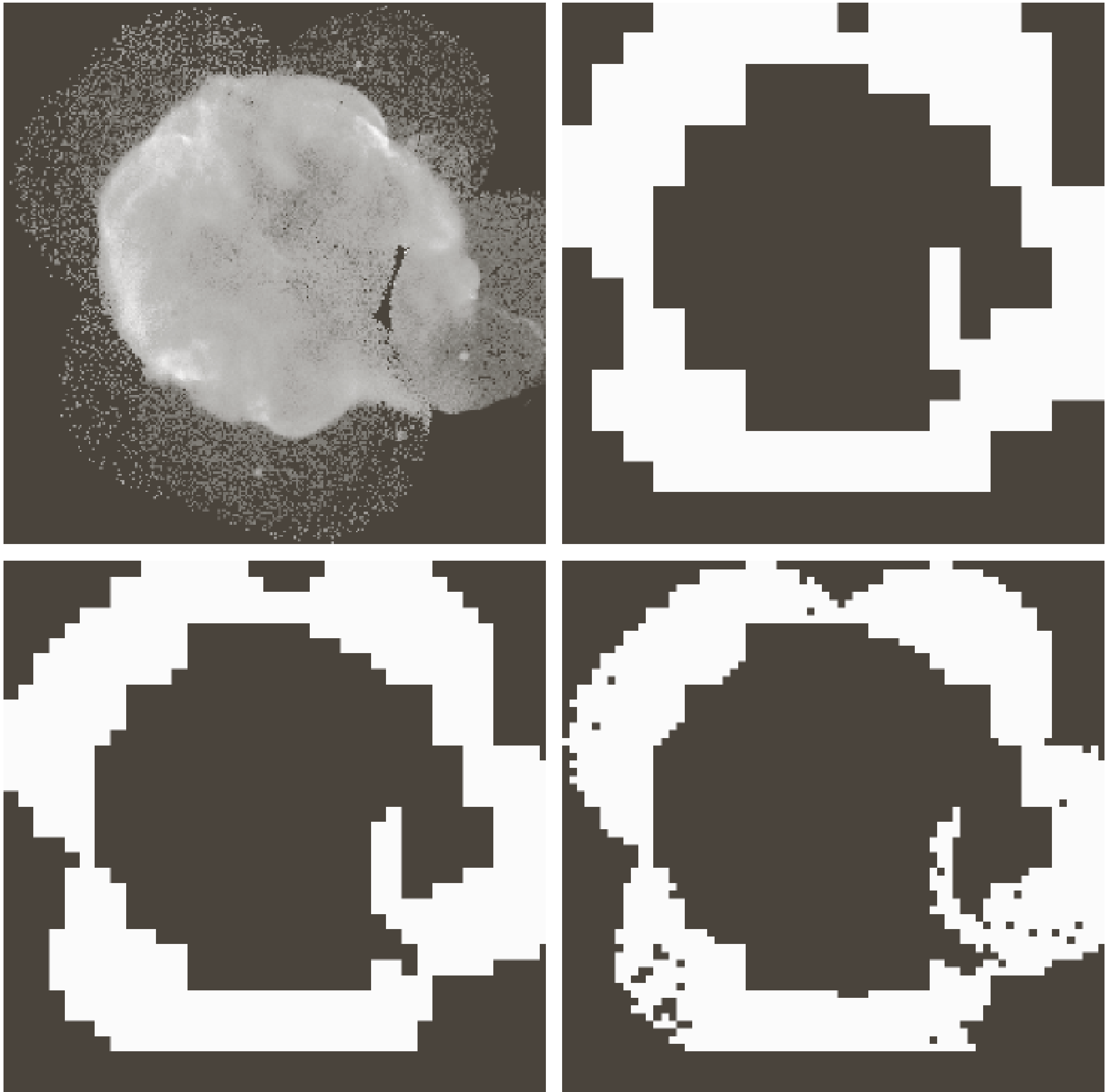


Min size 32x32

Min size 16x16

Splitting Ex: Condition: $\sigma > 10 \,\&\, 0 < m < 125$

σ is variance and m is mean



Min size 32x32

Min size 16x16

Min size 8x8

Region Merge Example

10	**11**	**10**	**11**	**15**	**16**	**15**	**15**
10	10	10	11	15	15	16	15
11	11	10	11	16	15	15	15
11	10	11	10	16	15	16	15
17	18	17	18	9	8	3	4
17	18	18	17	9	8	3	4
18	17	18	17	12	11	21	21
17	18	17	18	11	11	20	20

Minimum size 2x2

Condition:

Pixel difference between two consecutive pixels (Separation Threshold) <=2

- Divide image into subimage of minimum size

Region Merge Example

- Divide image into subimage of minimum size

10	**11**	**10**	**11**	**15**	**16**	**15**	**15**
10	10	10	11	15	15	16	15
11	11	10	11	16	15	15	15
11	10	11	10	16	15	16	15
17	18	17	18	9	8	3	4
17	18	18	17	9	8	3	4
18	17	18	17	12	11	21	21
17	18	17	18	11	11	20	20

Minimum size 2x2

Condition:
Pixel difference between two consecutive pixels (Separation Threshold) <=2

- Merge subimages if neighboring images satisfy the condition

Region Merge Example

- Divide image into subimage of minimum size
- Merge subimages if neighboring images satisfy the condition

10	**11**	**10**	**11**	**15**	**16**	**15**	**15**
10	10	10	11	15	15	16	15
11	11	10	11	16	15	15	15
11	10	11	10	16	15	16	15
17	18	17	18	9	8	3	4
17	18	18	17	9	8	3	4
18	17	18	17	12	11	21	21
17	18	17	18	11	11	20	20

Condition:

Pixel difference between two consecutive pixels (Separation Threshold) <=2

Split and Merge

- Split image
- Merge segments with similar predicate

Split and Merge: Example

Image after splitting

10	**11**	**10**	**11**	**15**	**16**	**15**	**15**
10	10	10	11	15	15	16	15
11	11	10	11	16	15	15	15
11	10	11	10	16	15	16	15
17	18	17	18	9	8	3	4
17	18	18	17	9	8	3	4
18	17	18	17	10	9	21	21
17	18	17	18	8	10	20	20

Condition:

Pixel difference between two consecutive pixels (Separation Threshold) <=2

Split and Merge: Example

Image after splitting followed by merging

10	**11**	**10**	**11**	**15**	**16**	**15**	**15**
10	10	10	11	15	15	16	15
11	11	10	11	16	15	15	15
11	10	11	10	16	15	16	15
17	18	17	18	9	8	3	4
17	18	18	17	9	8	3	4
18	17	18	17	10	9	21	21
17	18	17	18	8	10	20	20

Condition:

Pixel difference between two consecutive pixels (Separation Threshold) <=2

Split and Merge

Split

Original image

Split and Merge