unesco
University Network
Member

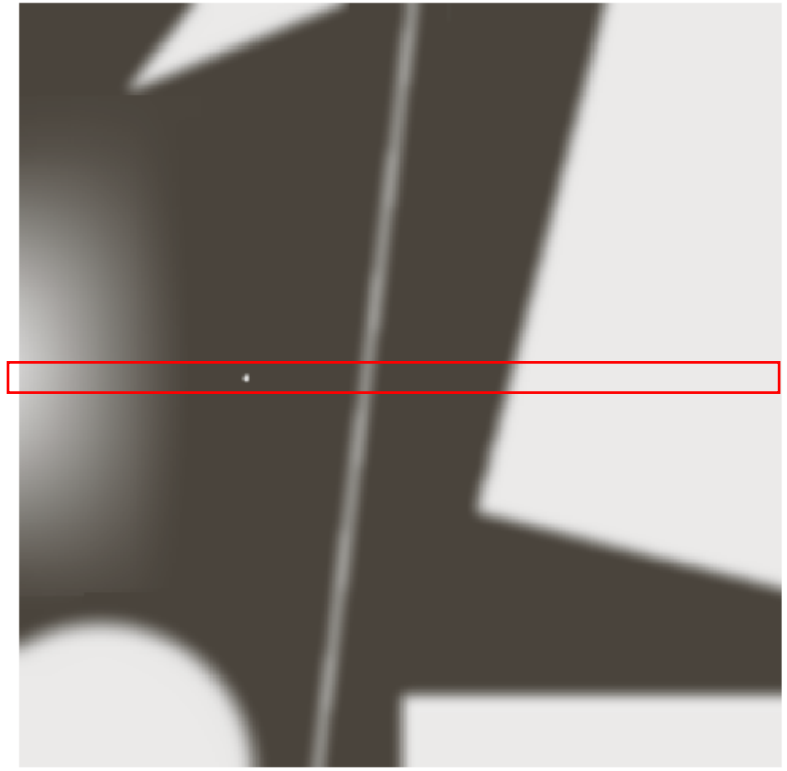
Ministry of
Education

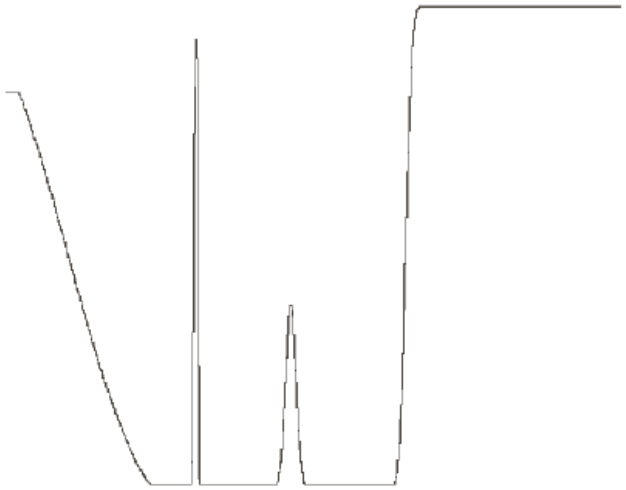
Edge Detection

Sung Soo Hwang

Introduction

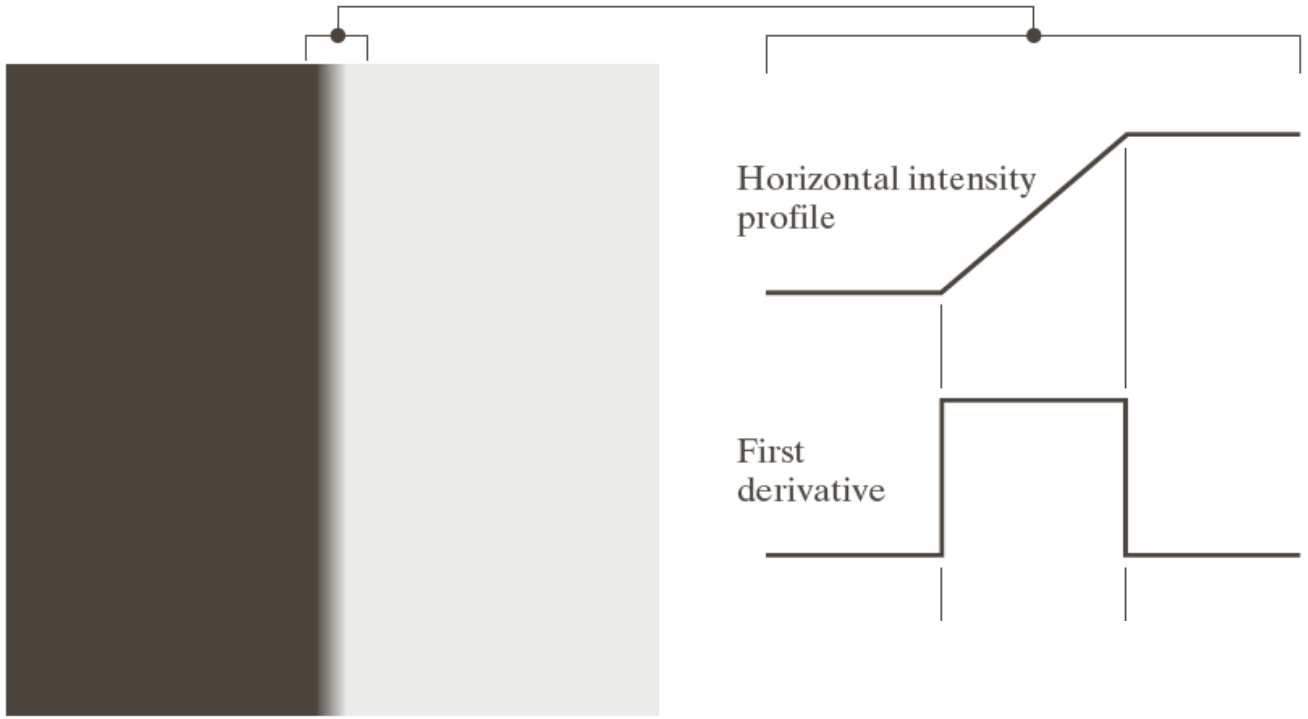
- Edge pixels: Pixels at which the intensity of an image changes abruptly
- Edges: Sets of connected edge pixels





Introduction

- How to detect edges? (in case of 1D)
  - The magnitude of the first derivative can be used to detect edges



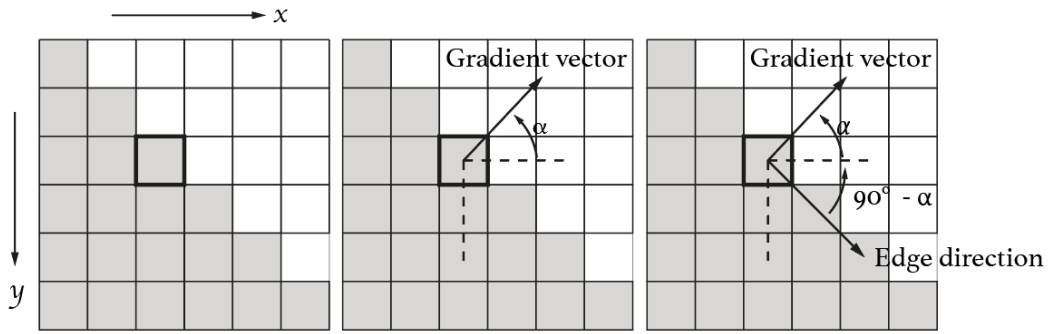
Introduction

- How to detect edges(in case of 2D)
  - By using image gradient

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

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

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



Introduction

- Effect of noise on edge detection

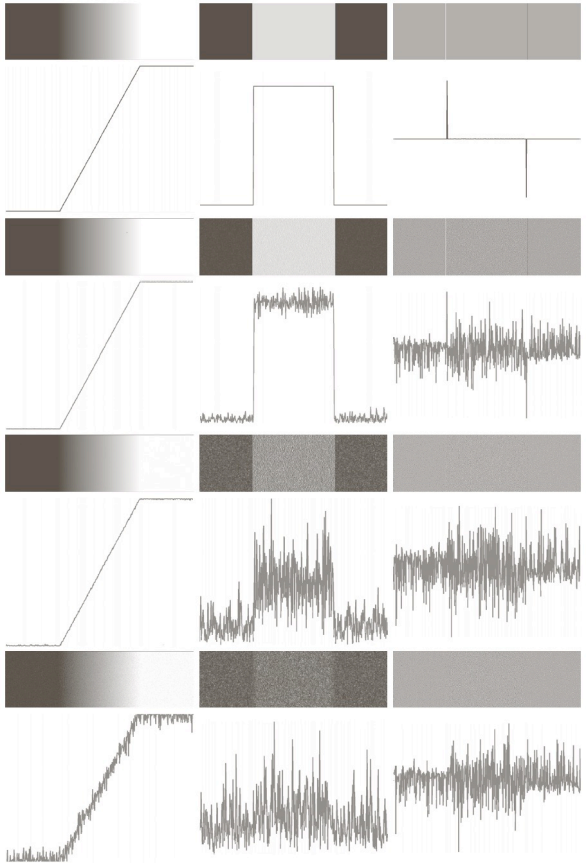
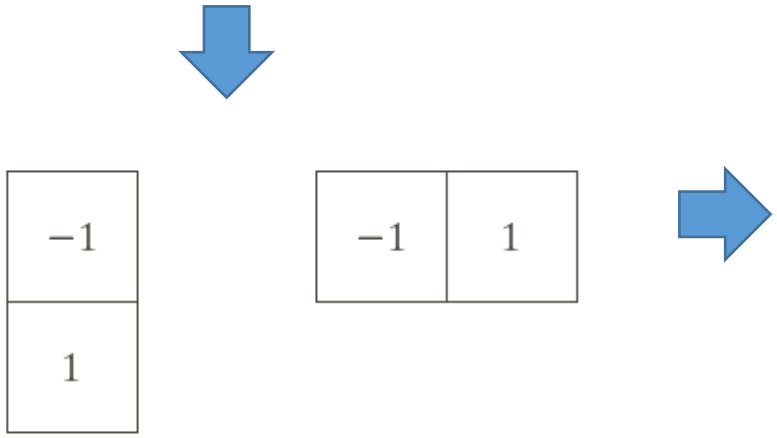


Image smoothing for noise reduction should be performed

Sobel operators

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

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



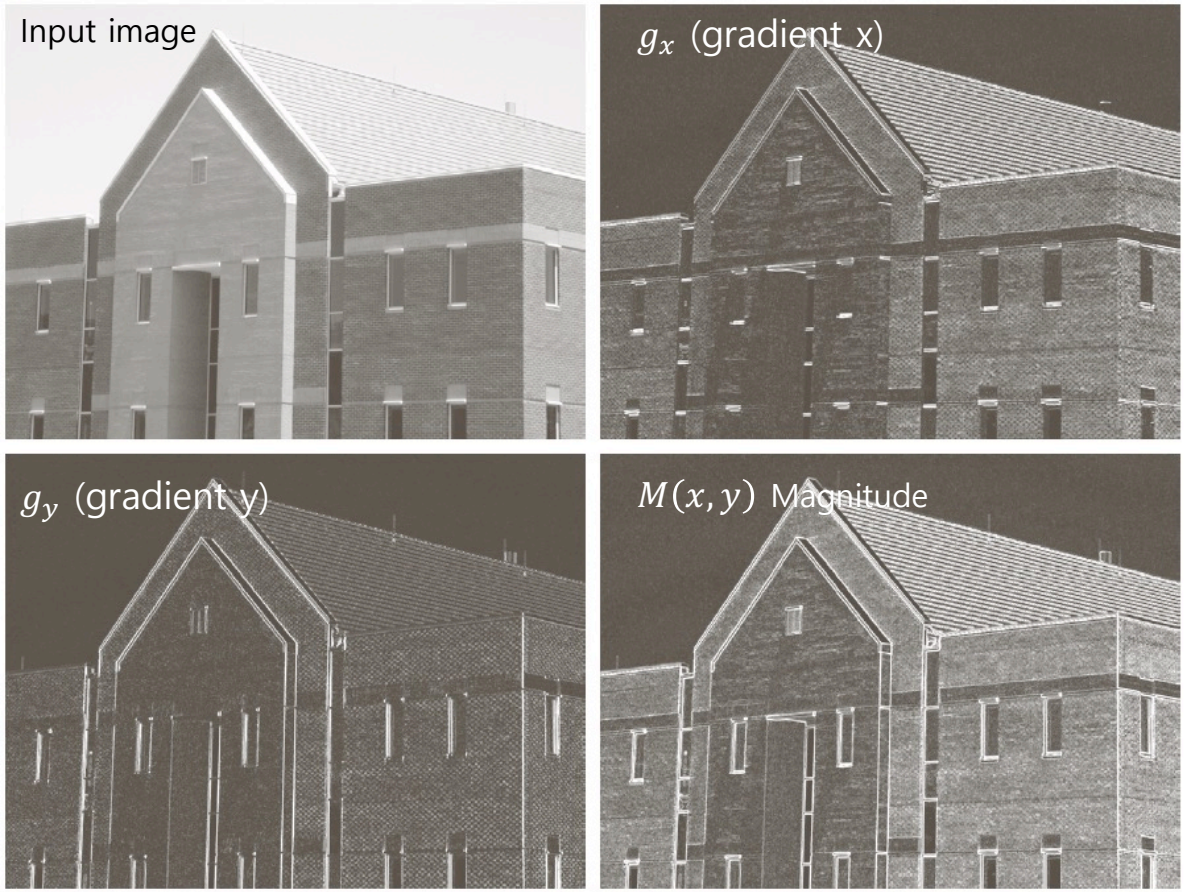
−1	−2	−1	−1	0	1
0	0	0	−2	0	2
1	2	1	−1	0	1

Sobel

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

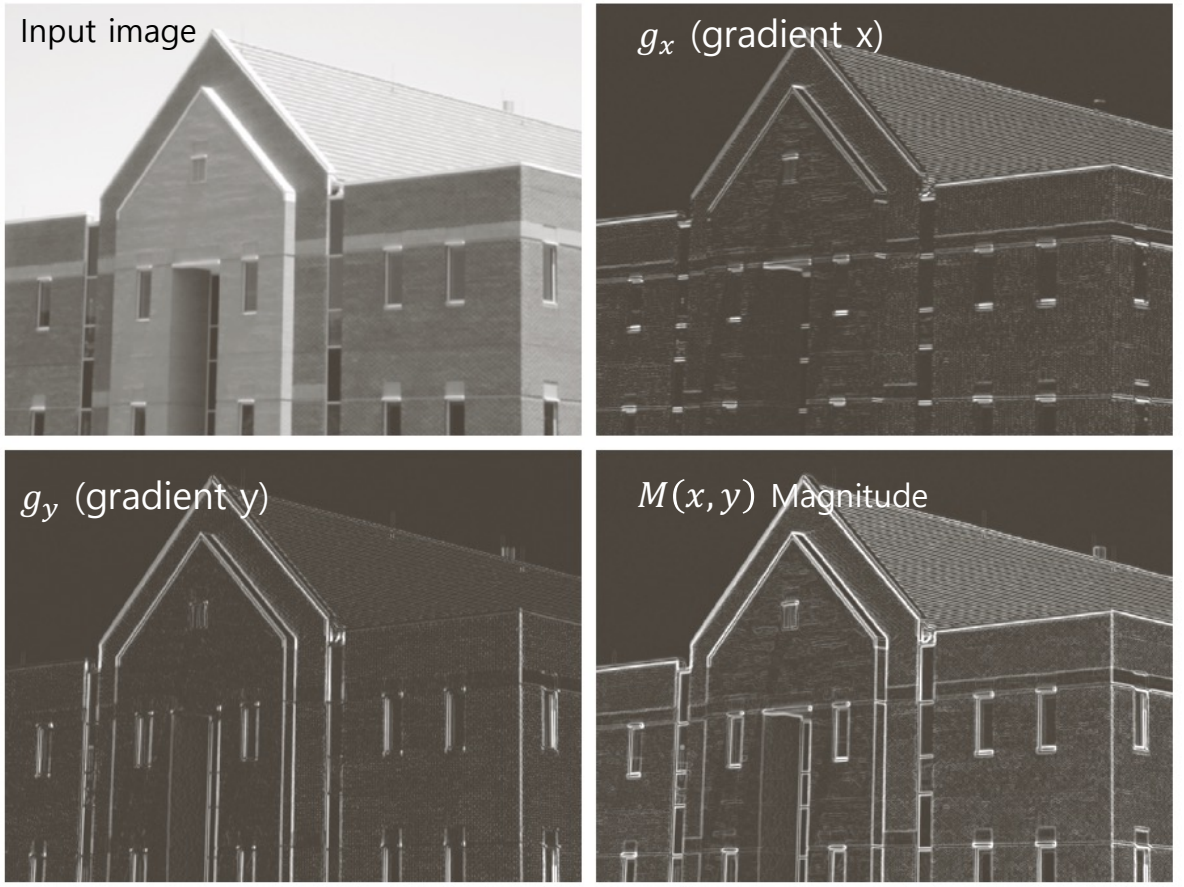
Sobel operators

- Result of applying gradient operators



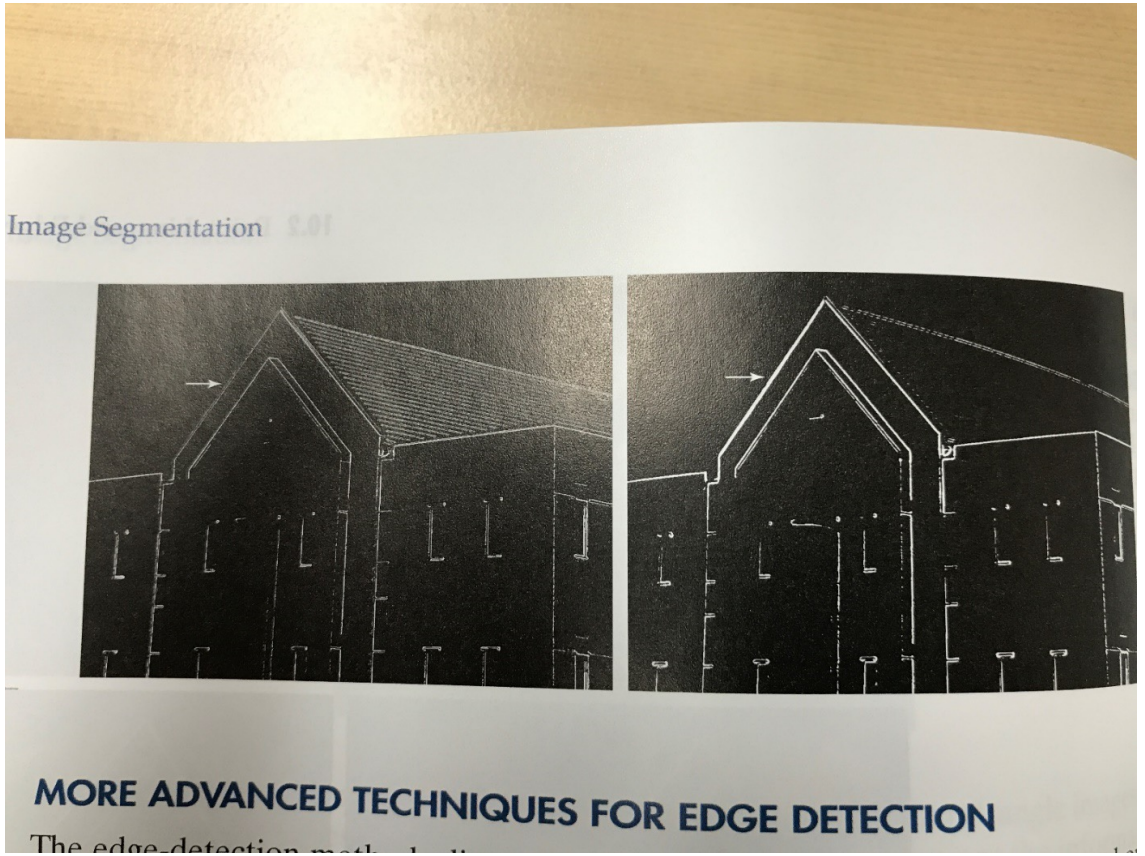
Sobel operators

- Result of applying gradient operators after 5X5 averaging filter



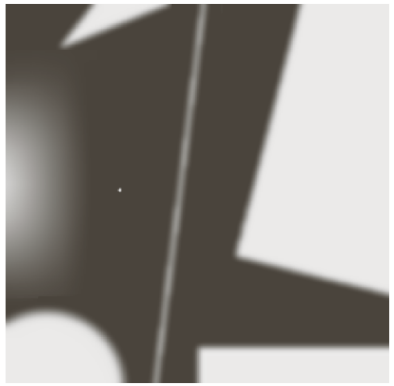
Sobel operators

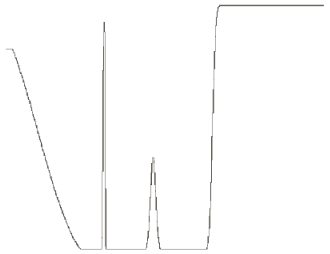
- Thresholding on magnitude of gradient

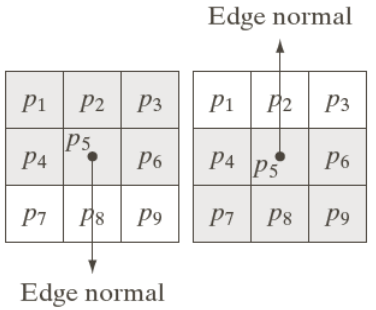


Canny Edge Detector

- Algorithm
  1. smooth the input image with a Gaussian filter
     - For noise removal
  2. Compute the gradient magnitude and angle images
     - Use Sobel edge mask
  3. Apply nonmaxima suppression to the gradient magnitude image
     - Find the direction d_k that is closest to $\alpha(x, y)$ (α:gradient direction)
     - If the value of M(x,y) is less than at least one of its two neighbors along d_k, suppress it(set to zero)

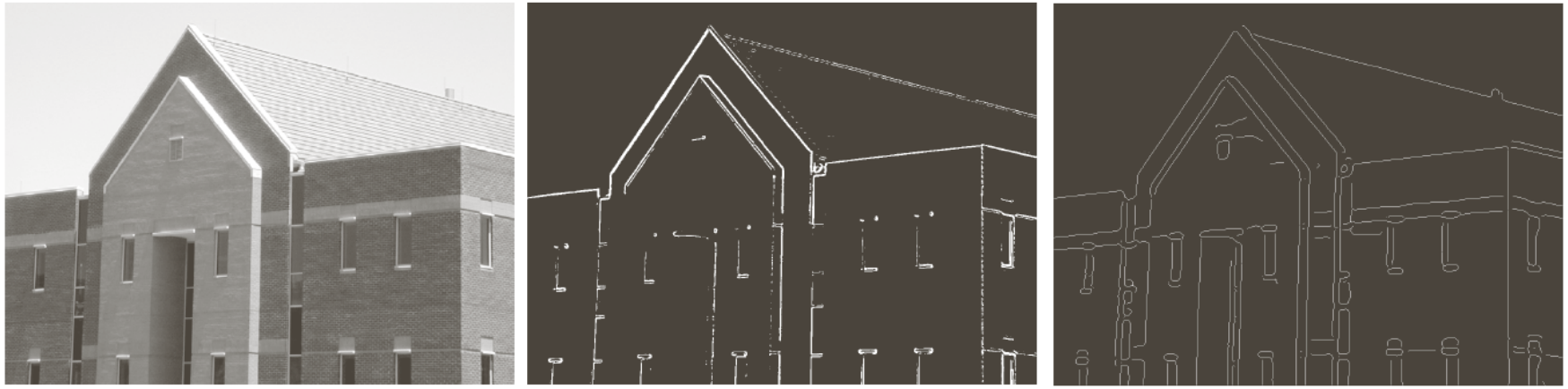






Canny Edge Detector

- Algorithm
  - 4. Use double thresholding and connectivity analysis to detect and link edges
    - $M(x, y) \geq T_H$ ← edge
    - $M(x, y) < T_L$ ← non-edge
    - Otherwise ← undetermined, use connectivity analysis

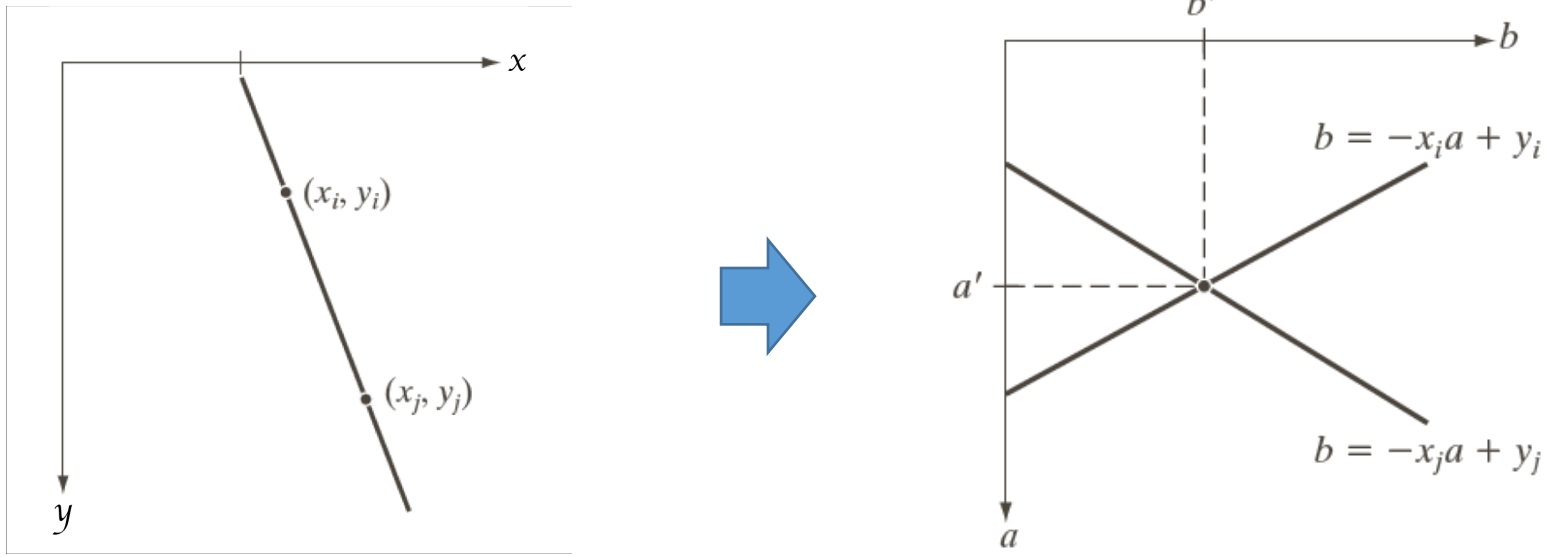


Line Detection

Sung Soo Hwang

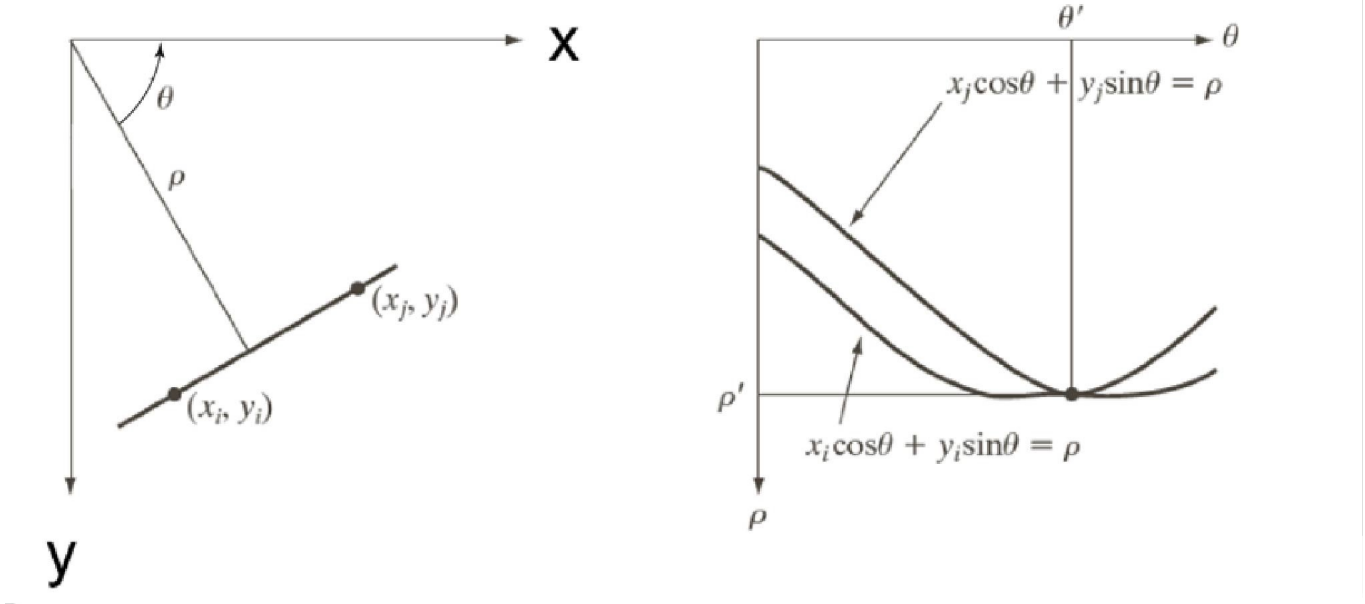
Line

- The concept of Hough transform
  - y=ax + b ➔ b= -ax + y
  - Find all combinations of (a, b) for each edge pixel
  - A combination of (a, b) which is used multiple times can be a line of an input image



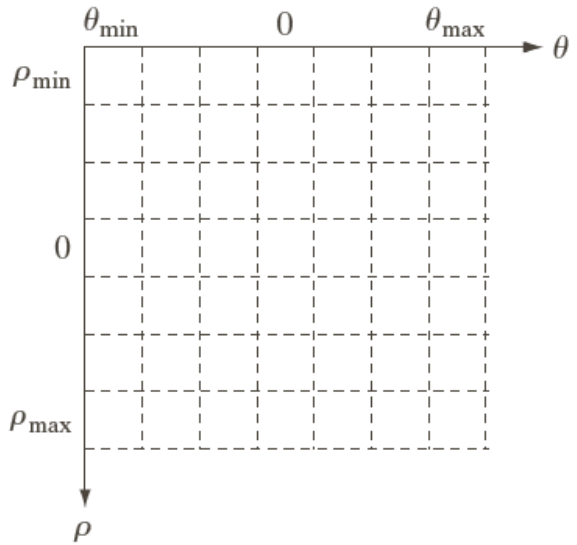
Line

- The concept of Hough transform
  - *a* becomes infinity as the line approaches the vertical direction
  - Use $\rho\theta$ representation instead
    - $xcos\theta + ysin\theta = \rho$



Hough Transform

- Algorithm
  - 1. obtain a binary edge image
  - 2. specify subdivision in the $\rho\theta$ – plane
  - 3. Examine the counts of the accumulator cells for high pixel concentrations



Hough Transform

- Circle detection
  - Hough transform is applicable to any function of the form g(v, c)=0, where v is a vector of coordinates and c is a vector of coefficients
    - Points lying on the circle
      - $(x - c_1)^2 + (y - c_2)^2 = c_3$
    - The result is a 3D parameter space