Edge-based Segmentation

基于边缘检测的图像分割

ZhaoHaiwei DaiJialun WangRuchen

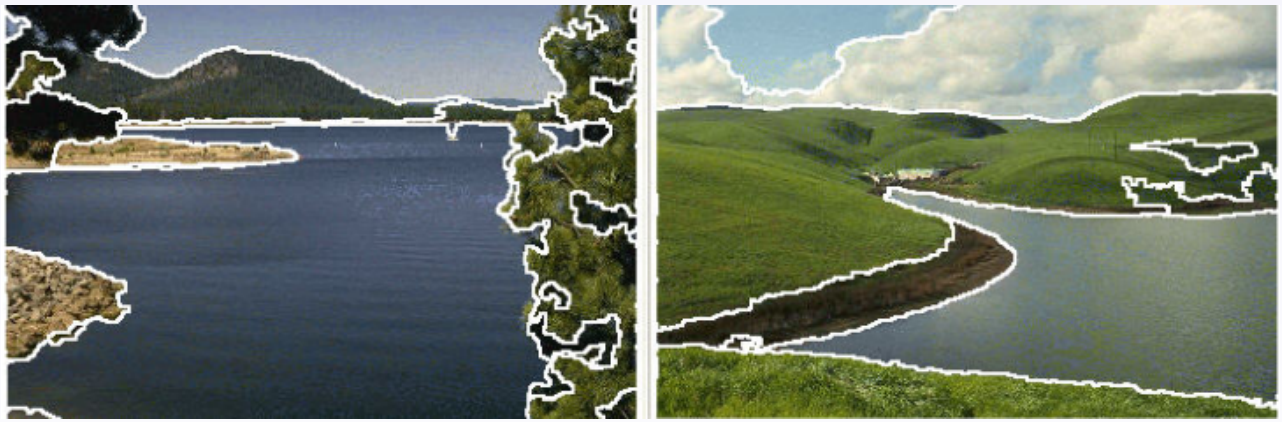
CVBIOUC

http://vision.ouc.edu.cn/~zhenghaiyong

April 24, 2015

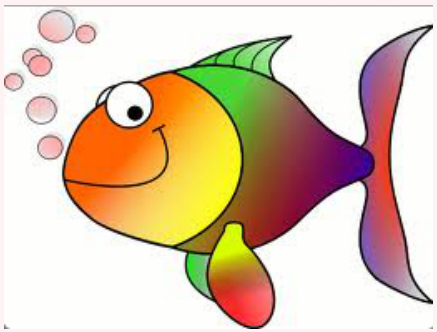
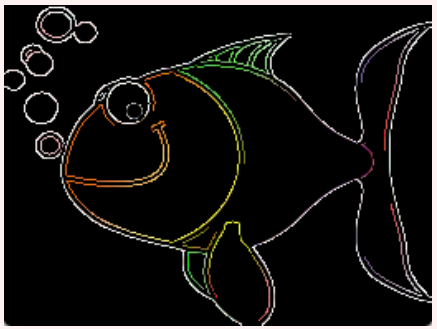
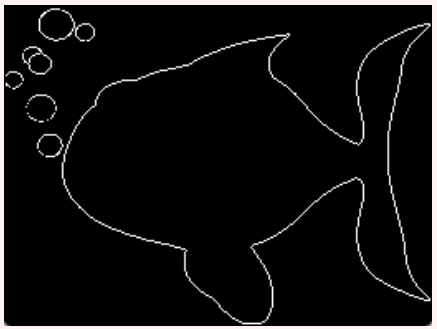
Contents

What?



Edge & Boundary

original image | edge | boundary

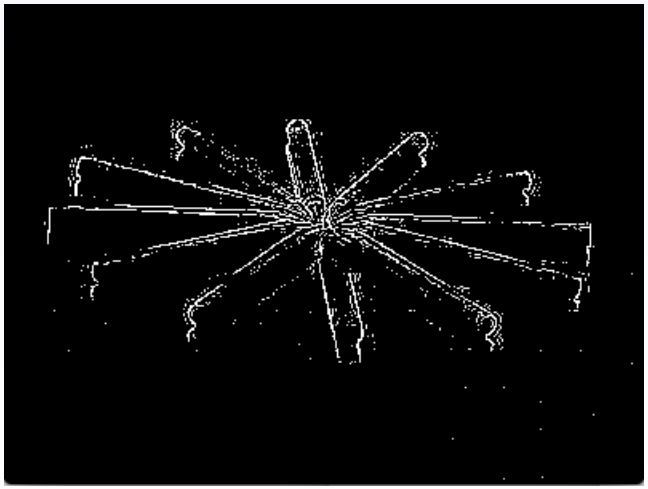




The edge of object may be not a boundary, the boundary may also be not edge.

Problems

Edge detection is a simple and fast technique used in segmentation methods. However, there also are some problems:

- Noise and background affect the accurate of edge detection.
- Edges are the sign of lack of continuity, and ending.



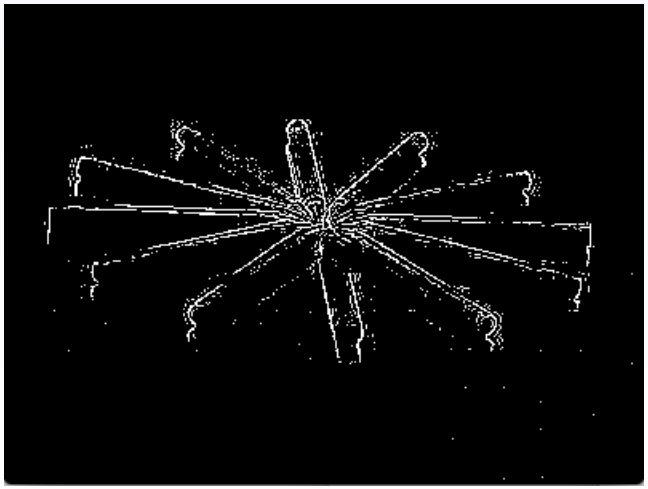
Steps in Edge Detection

1. Filtering: Filtering to reduce noise results in a loss of edge strength.
2. Enhancement: In order to facilitate the detection of edges, it is essential to determine changes in intensity in the neighborhood of a point.
3. Detection: Find the zero crossing and peak value to detect edge.

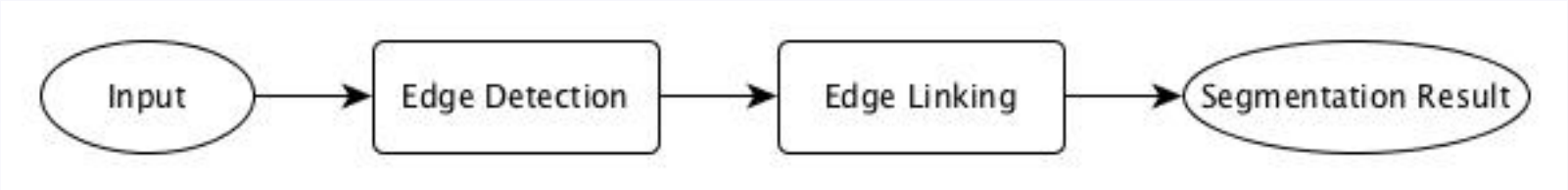
Problems

Edge detection is a simple and fast technique used in segmentation methods. However, there also are some problems:

- Noise and background affect the accurate of edge detection.
- Edges are the sign of lack of continuity, and ending.



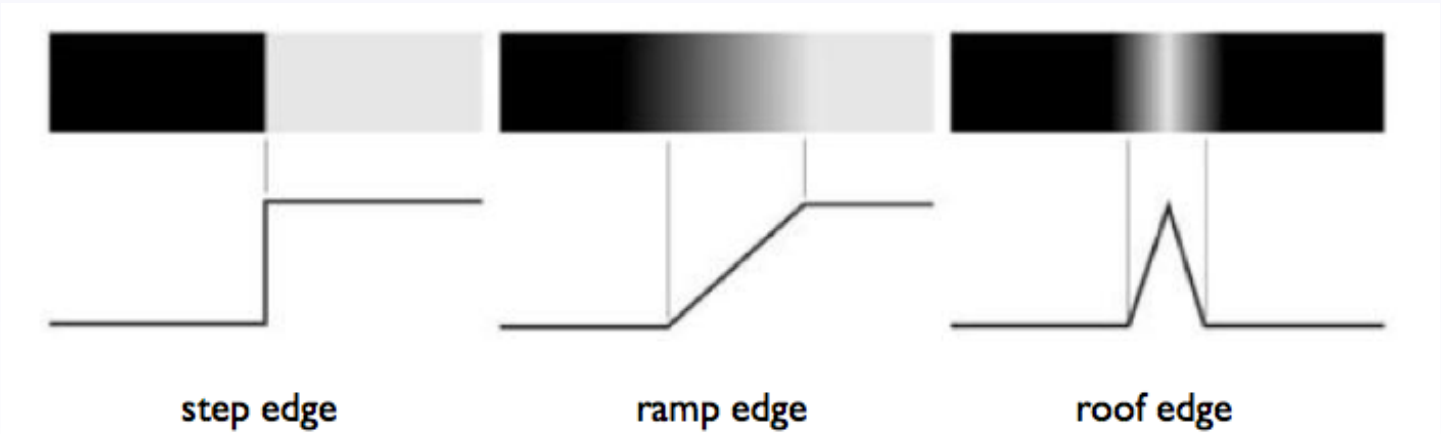
Step in Image Segmentation



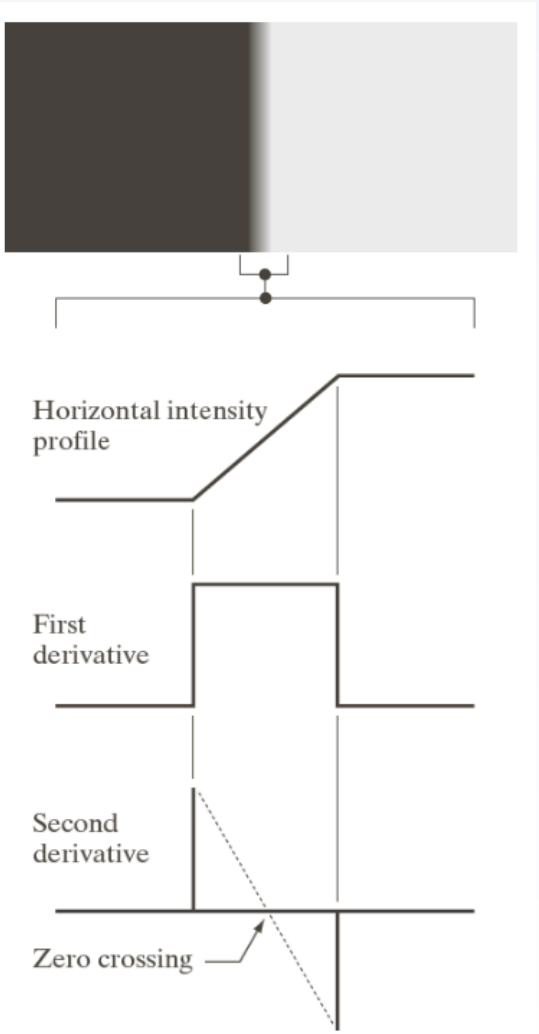
Edge Detection

Three edge types and definitions:

- Step edge
- Ramp edge
- Roof edge



Edge Detection



Edge Detection Techniques[1]

- Roberts Edge Detection
- Sobel Edge Detection
- Prewitt Edge Detection
- Marr-Hildreth Edge Detection (LoG)
- Canny Edge Detection

[1] Rafael C. Gonzalez *et al.*, "Digital Image Processing", Publishing House of Electronics Industry, 2011.

Gradient

The gradient $\bigtriangledown f$ of a function is:

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

The magnitude of the gradient is:

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

The directions of the greatest rate of change is:

$$\alpha(x, y) = arctan \left[\frac{g_x}{g_y} \right]$$

Edge Detection Techniques

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

- Roberts Edge Detection

-1	0
0	+1

0	-1
+1	0

$$g_x = \frac{\partial f}{\partial x} = z_9 - z_5$$

$$g_x = \frac{\partial f}{\partial y} = z_8 - z_6$$

Edge Detection Techniques

- Sobel Edge Detection

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

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

$$g_x = \frac{\partial f}{\partial x} = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$

$$g_x = \frac{\partial f}{\partial y} = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$$

- Prewitt Edge Detection

-1	-1	-1
0	0	0
+1	+1	+1

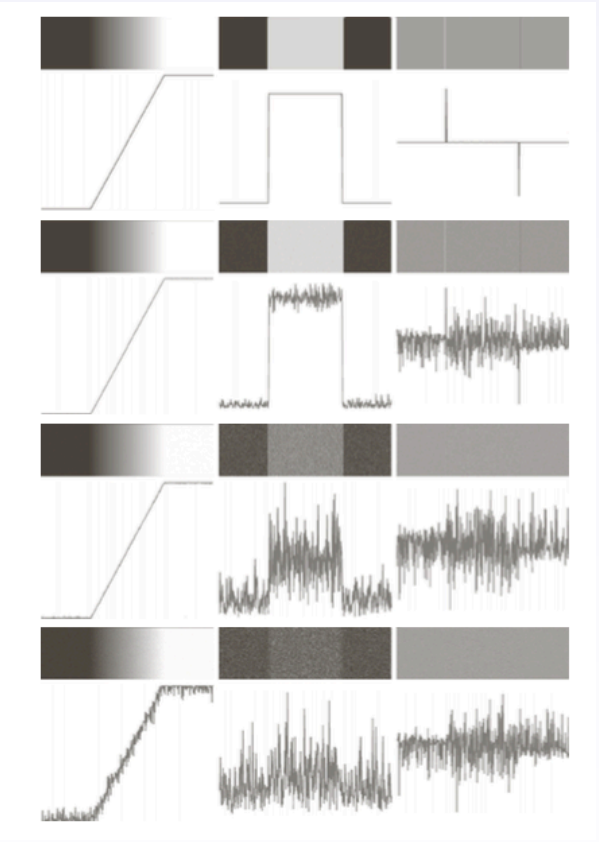
-1	0	+1
-1	0	+1
-1	0	+1

Marr-Hildreth Edge Detection (LoG)[2]

This method combines Gaussian filtering with the Laplacian for edge detection.

Idea:

1. Gaussian filer is used to swipe away noise from the image
2. Laplace operator
3. Locate zero crossings



[2]Marr D, Hildreth E, "Theory of Edge Detection", Proc. of the Royal Society of London, 1980.

Canny Edge Detection[3]

Canny is the best edge detection detector.

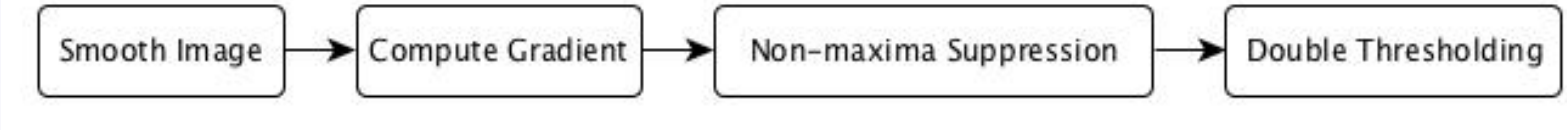
Idea:

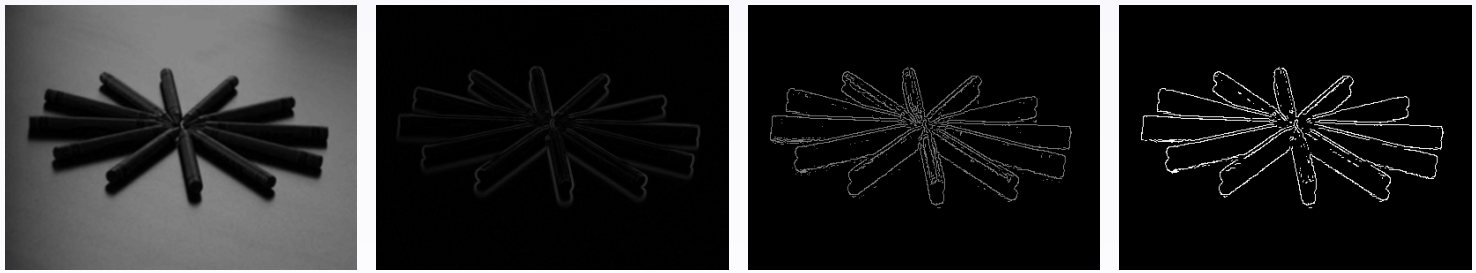
1. Low error rate
2. Edge points should be well localized

[3] Joun F. Canny, "A Computational Approach To Edge Detection", PAMI, 1986.

Canny Edge Detection

Canny algorithm:



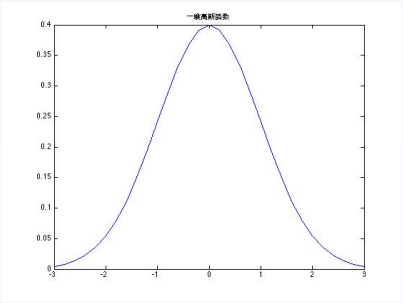


Gauss Filter:

1 Gaussian distribution

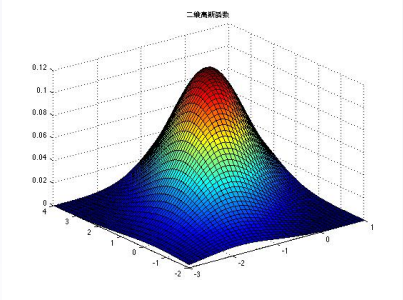
- *one dimension:*

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$



- *two dimension:*

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$



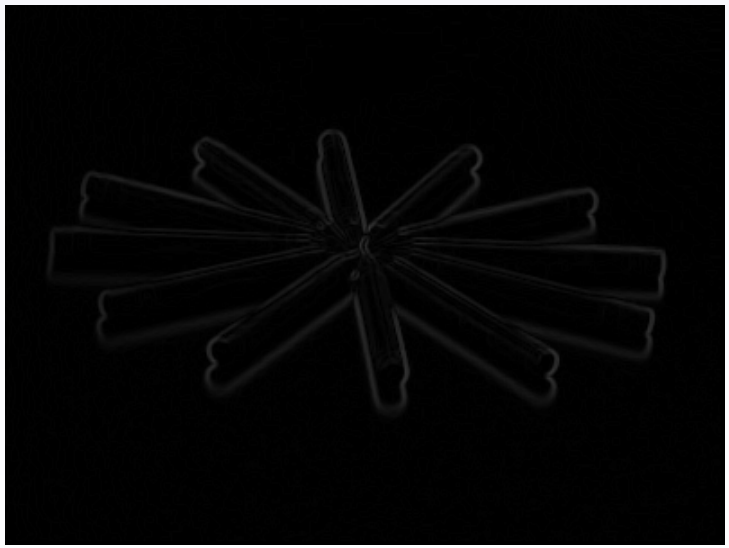
2 Gaussian kernel

$$\frac{1}{273}$$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

Compute Gradient:

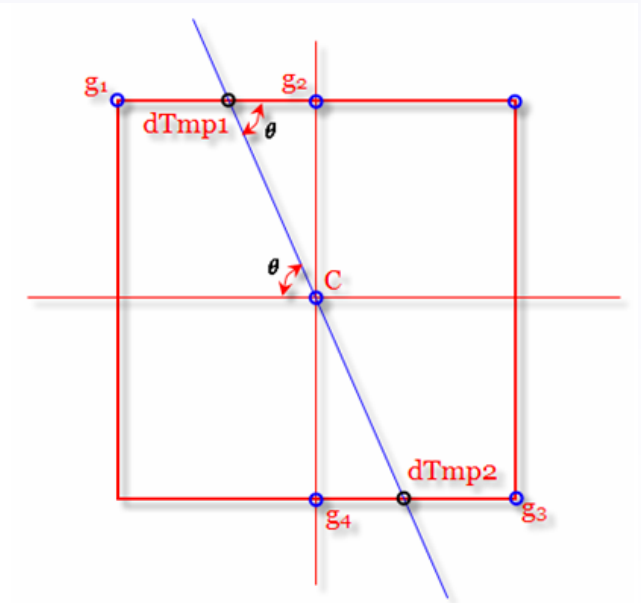
$$S_x = \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}, S_y = \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}$$



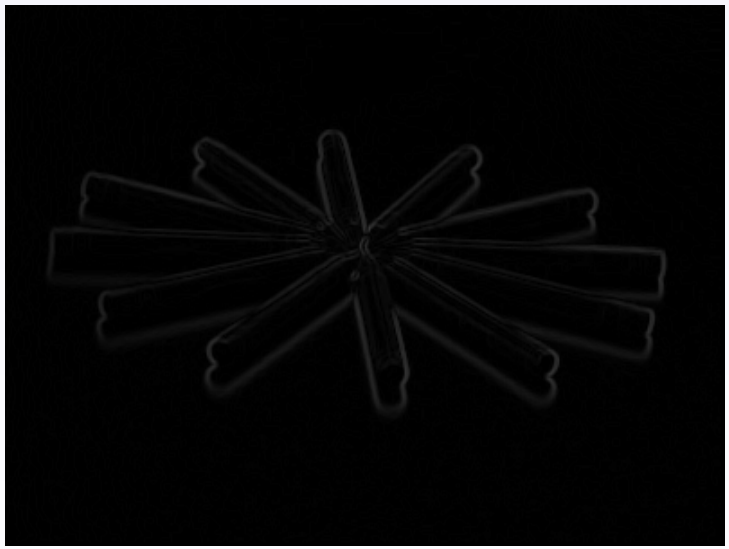
Canny Edge Detection

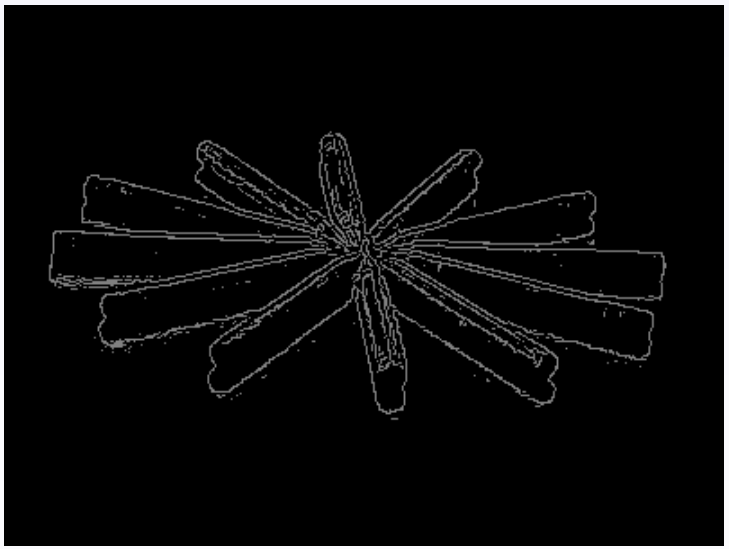
Non-maxima Suppression (NMS):

- Problem: Edges generated using gradient typically contain wide ridges around local maxima.
- Use non-maxima suppression to thin those ridges to find thin edges corresponding to local maxima.



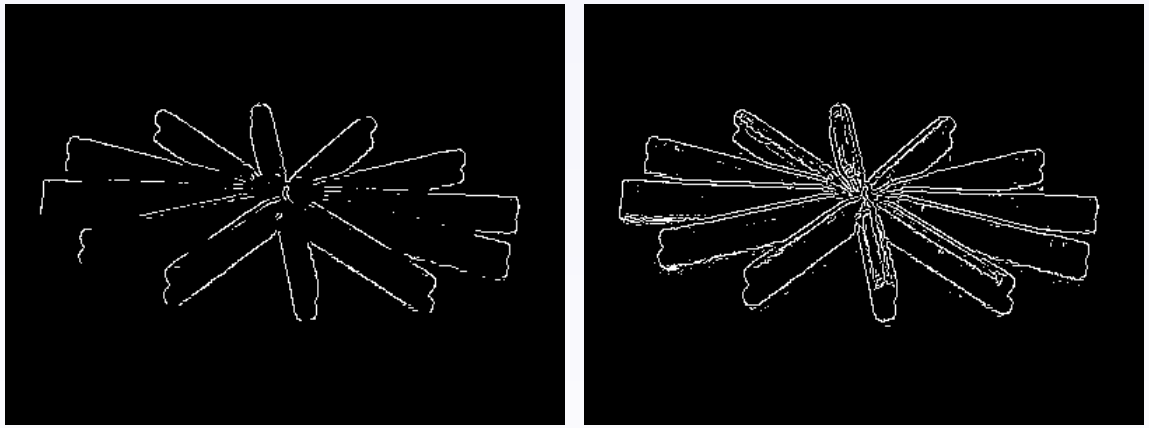
Canny Edge Detection

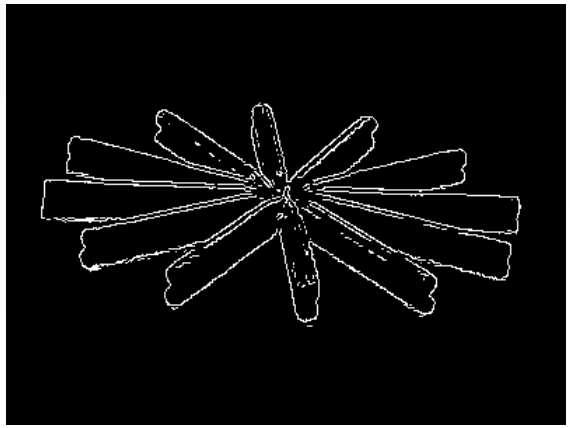




Double thresholding:

- Problem: The received image may still contain false edge points.





Edge Linking

- Edge Tracking
- Hough Transform
- Curve Fitting
- Dynamic Programming

...

Edge Tracking

Idea:

- Each point is linked to the adjacent if magnitude and direction of the gradient are similar.

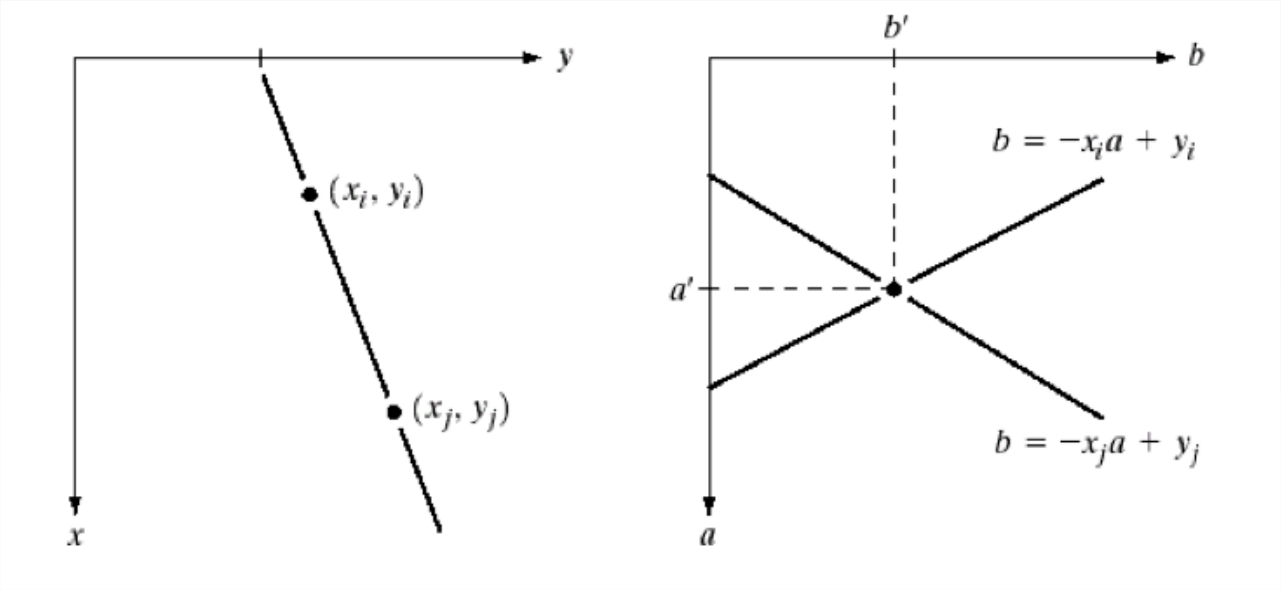
$e(x, y)$ is magnitude of the gradient, $\theta(x, y)$ is the direction of the gradient, if two each points meet the following conditions:

$$\begin{cases} \mid e(x_i, y_i) - e(x_j, y_j) \mid \leqslant E \\ \mid \theta(x_i, y_i) - \theta(x_j, y_j) \mid \leqslant A \\ \mid e(x_i, y_i) \mid, \mid e(x_j, y_j) \mid > E \end{cases}$$

Hough Transform

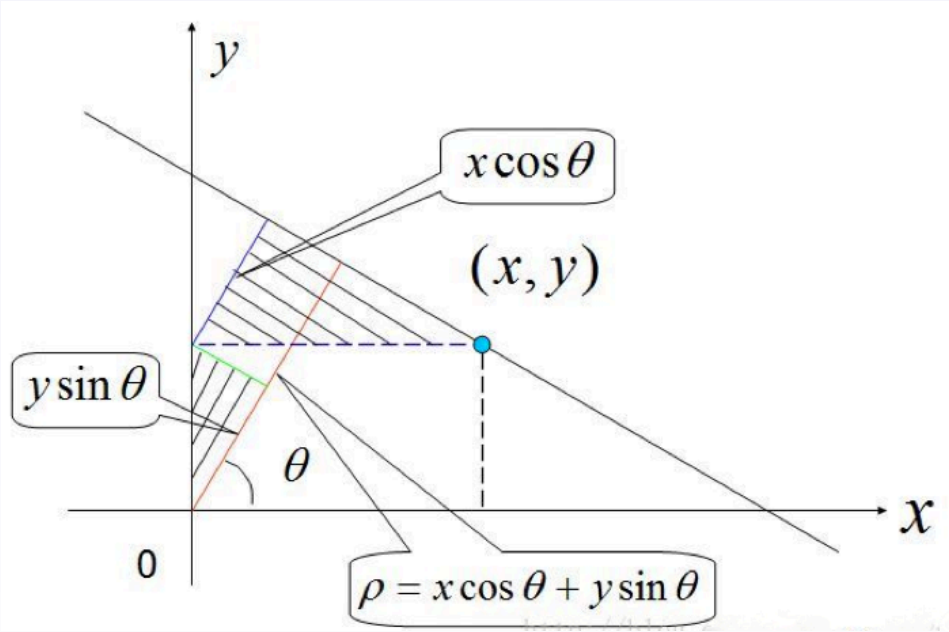
The Hough Transform can be used to detect lines, circles or other parametric curves.

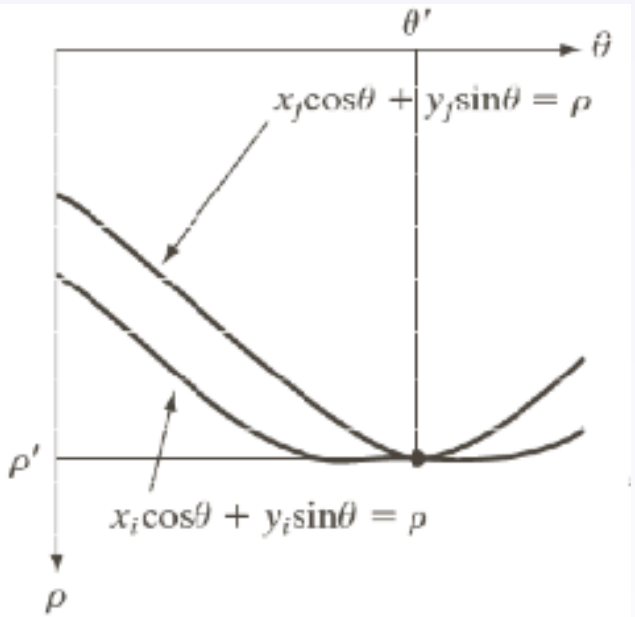
Idea:



$$y = a'x + b' \qquad \begin{cases} b = -x_i a + y_i \\ b = -x_j a + y_j \end{cases}$$

Hough Transform



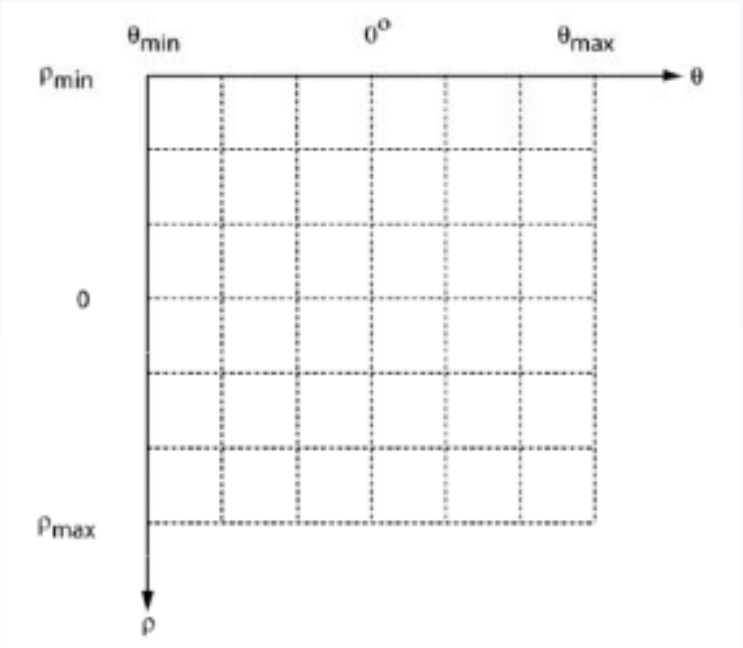


$$\begin{cases} \rho' = x_i cos\theta' + y_i sin\theta' \\ \rho' = x_j cos\theta' + y_j sin\theta' \end{cases} \Longrightarrow \begin{cases} x_i cos\theta + y_i sin\theta = \rho \\ x_j cos\theta + y_j sin\theta = \rho \end{cases}$$

Hough Transform

Hough algorithm:

1. Quantize the parameter space (ρ, θ). This quantized space is referred to as the accumulator cells.
2. Count the number of times a line intersects a given cell.
3. Lines can be found as peaks in this accumulator space.

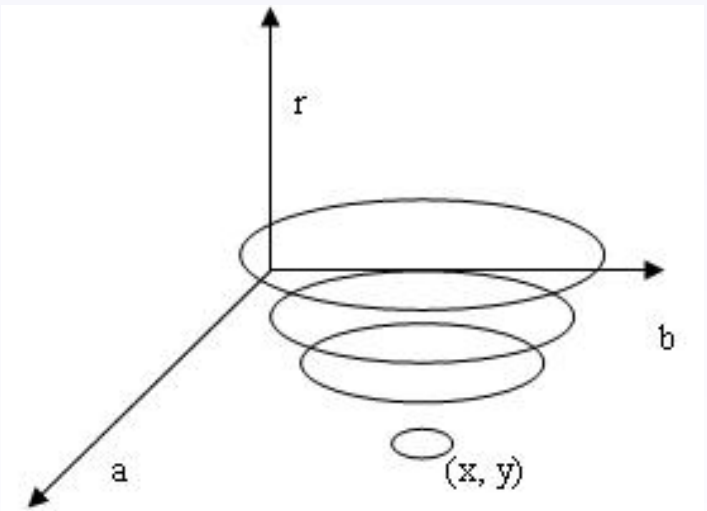


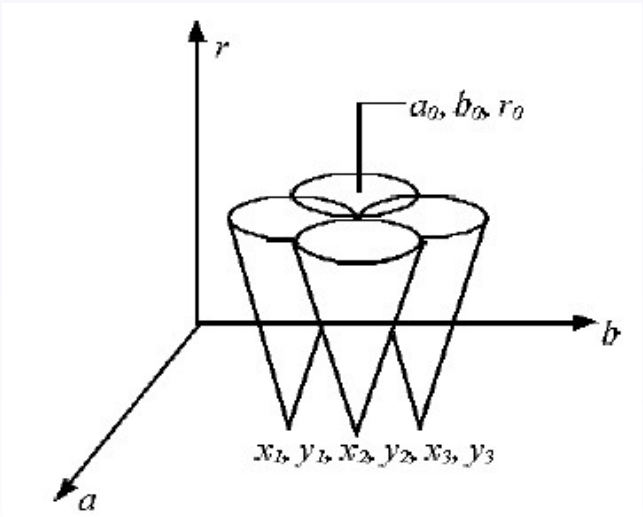
$$\rho = xcos\theta + ysin\theta$$

Hough Transform

Circle:

$$(x-a)^2+(y-b)^2=r$$





$$(a-x_i)^2+(b-y_i)^2=r$$

Thanks!