Kỹ thuật xử lý ảnh số

Giảng viên: TS. Nguyễn Thúy Bình

Email: thuybinh_ktdt@utc.edu.vn

Nội dung môn học

- 1. Thời lượng: 2 tín chỉ (9 buổi LT + 4 buổi BT)
- 2. Đánh giá học phần:
 - ➤ Điểm danh: 10%
 - Diểm bài tập: 15%
 - Bài kiểm tra: 15%
 - Thi kết thúc học phần: 60%

Nội dung môn học

1. Lý thuyết

- Tổng quan về xử lý ảnh
- Cải thiện và nâng cao chất lượng ảnh
- Phát hiện và tách biên ảnh
- Phân vùng ảnh
- Xử lý ảnh nhị phân

2. Bài tập+Thực hành:

- Biểu diễn ảnh và biến đổi không gian màu
- Cải thiện và nâng cao chất lượng ảnh
- Phát hiện và tách biên ảnh
- Phân vùng ảnh
- Xử lý ảnh nhị phân

Phân vùng ảnh

Hình ảnh được phân chia thành các vùng con, gọi là segment



Phân vùng ảnh

- Làm giảm độ phức tạp của hình ảnh
- Giúp quá trình phân tích, xử lý hình ảnh trở nên đơn giản



Gán nhãn cho từng điểm ảnh, các điểm ảnh thuộc cùng một vùng có nhãn giống nhau



- Tiếp cận tương đồng (Similarity approach)
- Tiếp cận gián đoạn (Discontinuty approach)

Phân vùng ảnh

- Hướng tiếp cận tương đồng: phát hiện sự tương đồng giữa các điểm ảnh → phân đoạn dựa trên ngưỡng
- Hướng tiếp cận gián đoạn: phát hiện sự thay đổi cường độ sáng đột ngột → biên, đường, cạnh → phân vùng

■ Phân loại:

- Phân vùng dựa trên biên (Edge Based Segmentation)
- Phân vùng dựa trên ngưỡng (Threshold Based Segmentation)
- Phân vùng dựa trên khu vực (Region-Based Segmentation)
- Phân vùng dựa trên kỹ thuật phân cụm (Clustering Based Segmentation)
- Phân vùng dựa trên mạng nơron nhân tạo (Artificial Neural Network Based Segmentation)

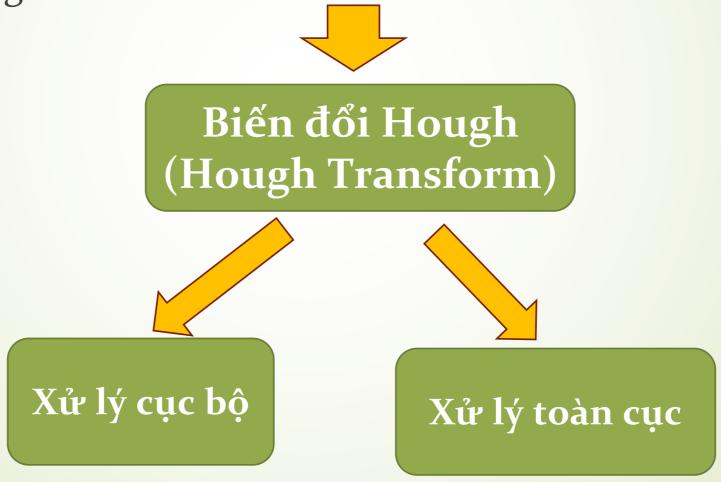
- Biên ảnh: tập hợp các điểm ảnh nằm trên ranh giới giữa hai vùng ảnh.
- Phát hiện biên ảnh dựa trên đạo hàm bậc nhất (Gradient) và đạo hàm bậc hai (Laplace)
- Phân vùng ảnh dựa trên biên:
 - Biển thường không khép kín
 - Nhiễu gây nên biên giả



Bộ lọc Sobel



- * Kết nối biên dựa trên 2 tính chất:
 - Cường độ sáng (biên độ) của các điểm ảnh nằm trên biên
 - Hướng đi của biên



Xử lý cục bộ

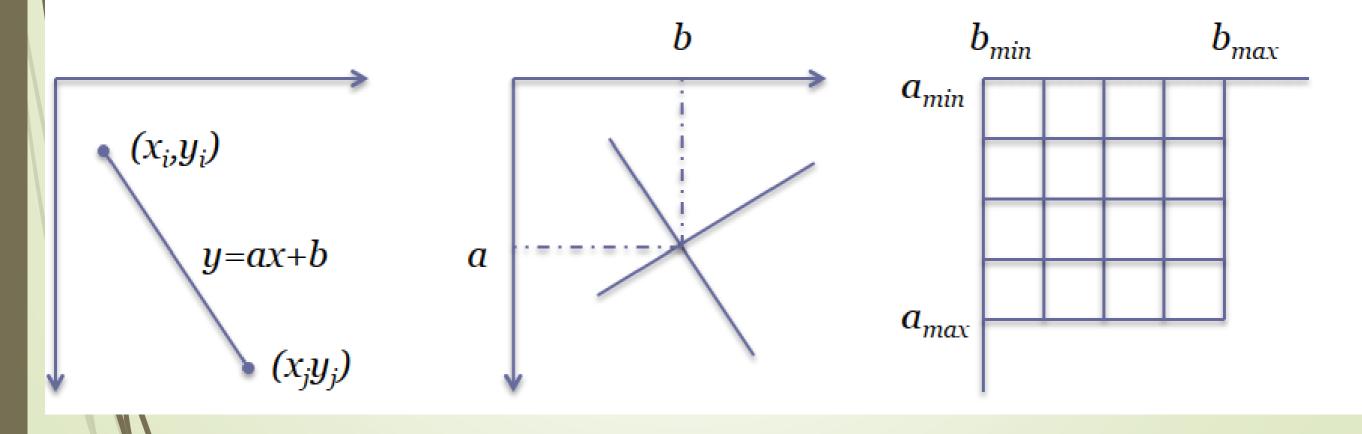
• Cho điểm biên (x_o, y_o) , điểm ảnh (x,y) thuộc lân cận của (x_o, y_o) được kết nối với (x_o, y_o) nếu

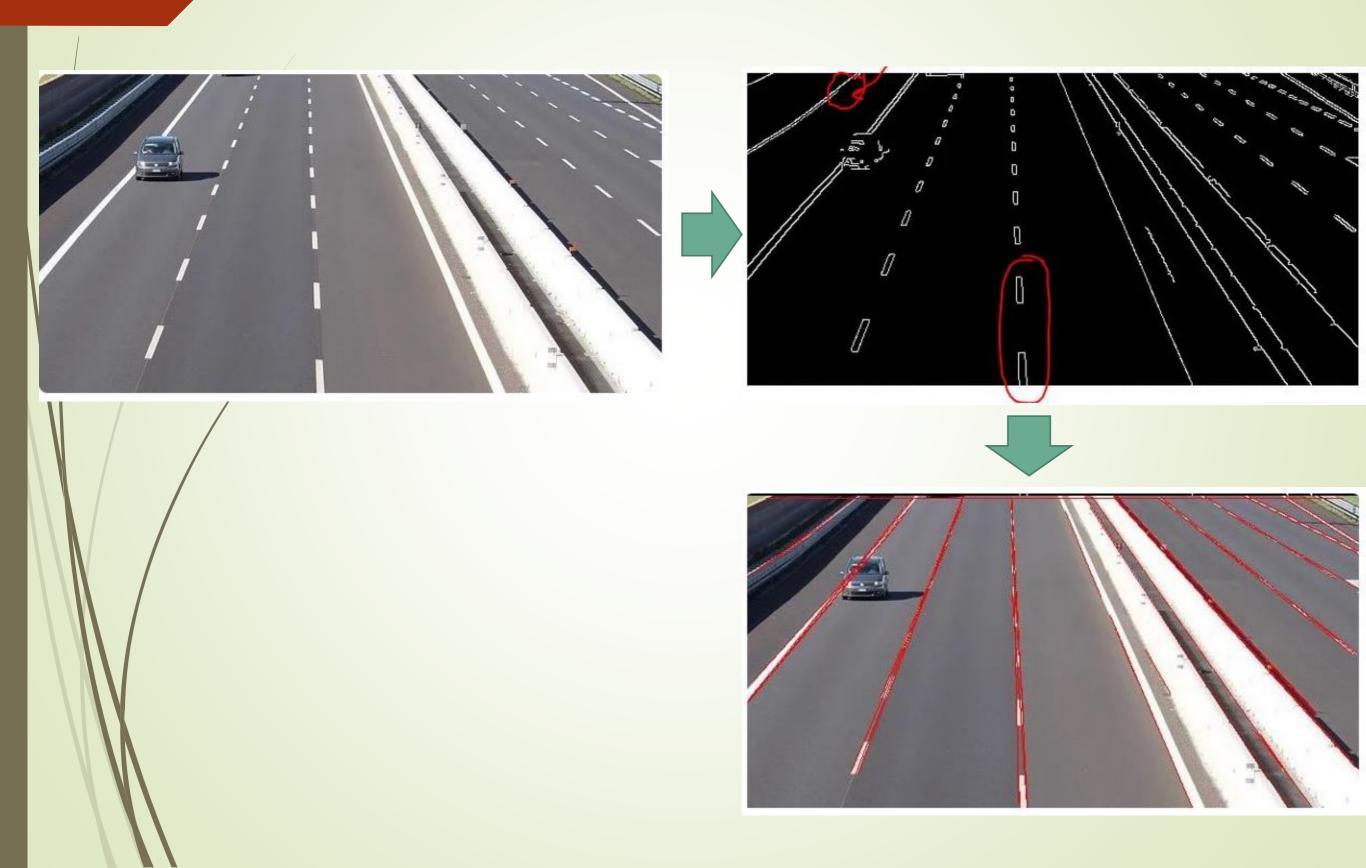
$$\left|\nabla f(x,y) - \nabla f(x_0,y_0)\right| \le E$$
 $\left|\alpha(x,y) - \alpha(x_0,y_0)\right| \le A$

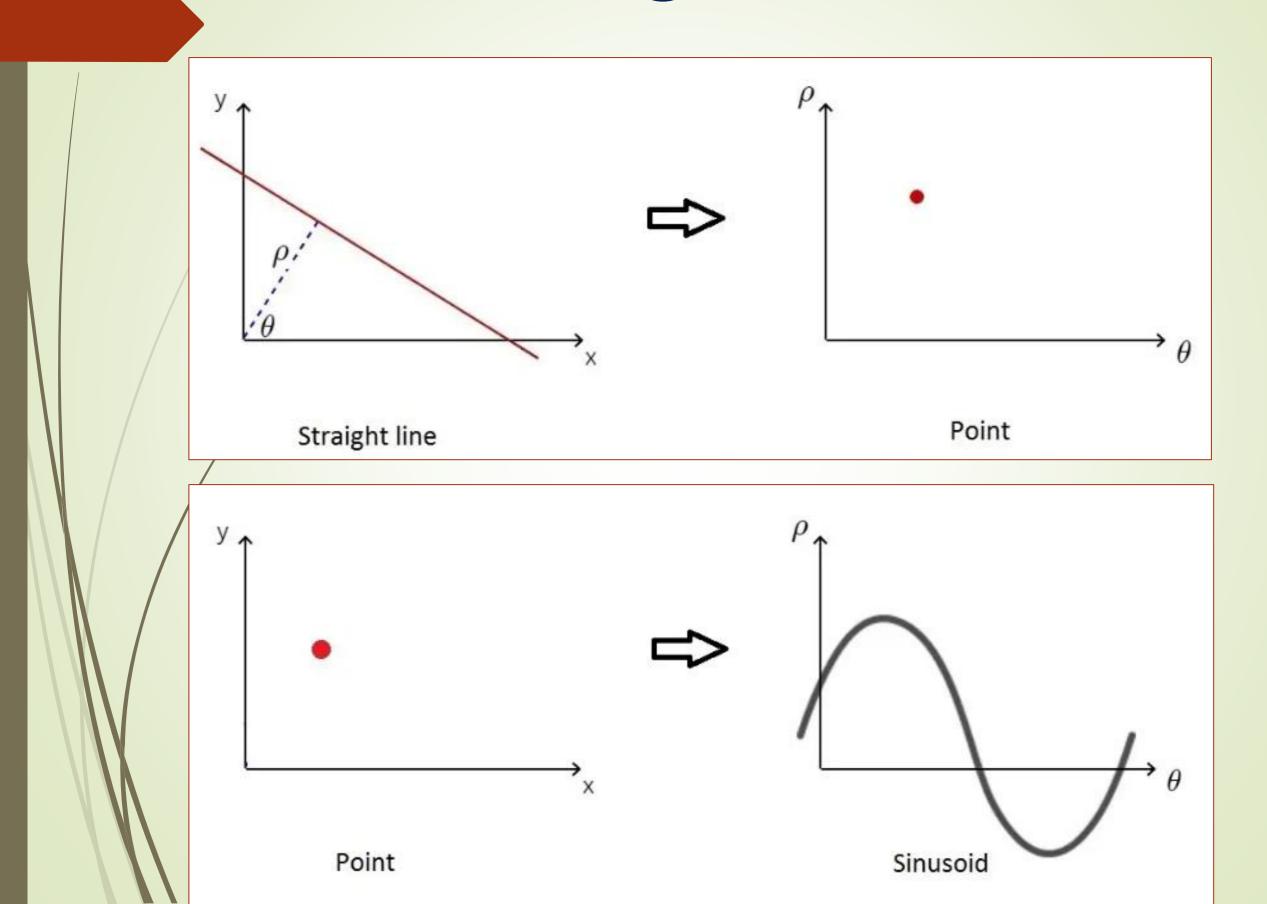
- Với E và A là các giá trị ngưỡng. Quá trình trên được lặp lại với tất cả các điểm ảnh. Các lân cận được xem xét thuộc mặt na kích thước 3x3 hoặc 5x5.
- Kỹ thuật xử lý cục bộ không tính đến lịch sử tìm kiếm nên có thể tạo ra các điểm biên giả.

Xử lý toàn cục

• Kỹ thuật xử lý toàn cục tìm đường thẳng xấp xỉ đi qua n điểm ảnh. Giả sử đường thẳng đi qua điểm ảnh (x_i, y_i) có dạng $y_i = ax_i + b$, ta có thể viết $b = y_i - ax_i$. Từ đó, đường thẳng được tìm kiếm trên mặt phẳng tham số ab.

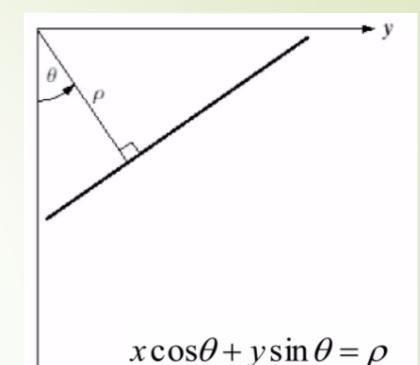




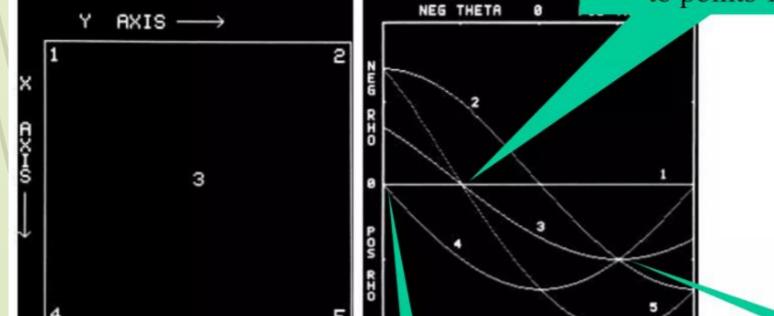


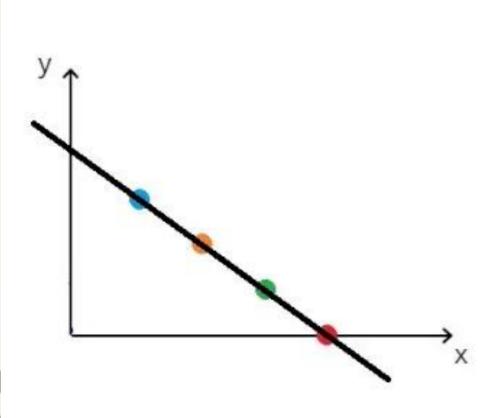
Biến đổi Hough:

- Tìm các cặp θ và ρ thỏa mãn phương trình đi qua cặp điểm (x,y)
- Các cặp giá trị θ và ρ được biểu
 diễn bởi mảng 2 chiều
- Biểu diễn các cặp giá trị θ và ρ →
 có dạng hình sin

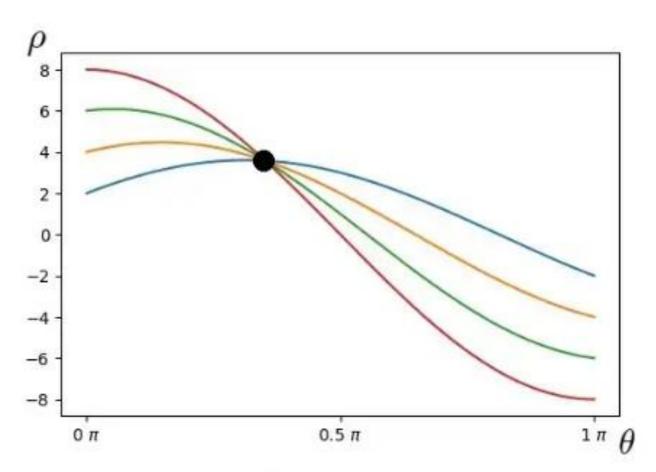


The intersection of the curves corresponding to points 1,3,5

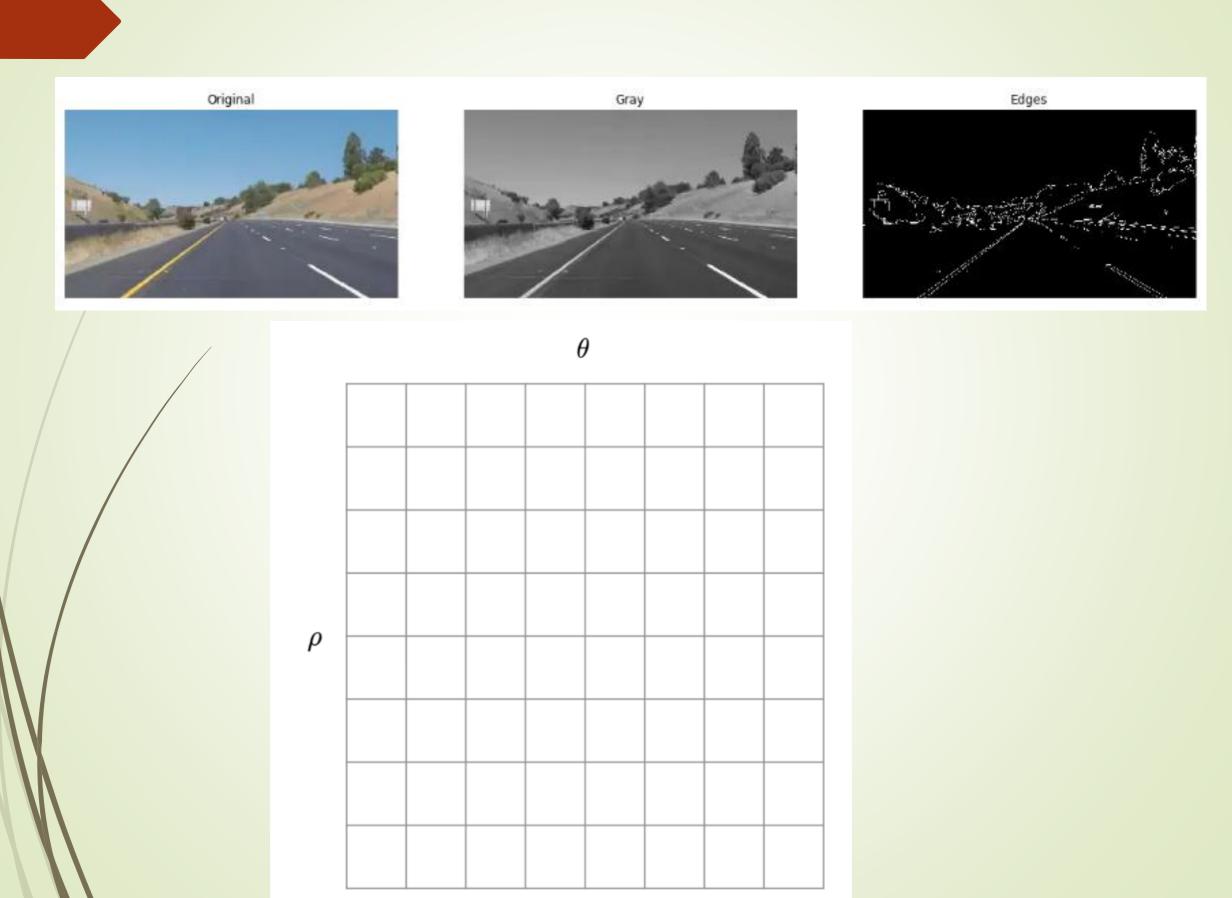




Points which form a line



Bunch of sinusoids intersecting at one point

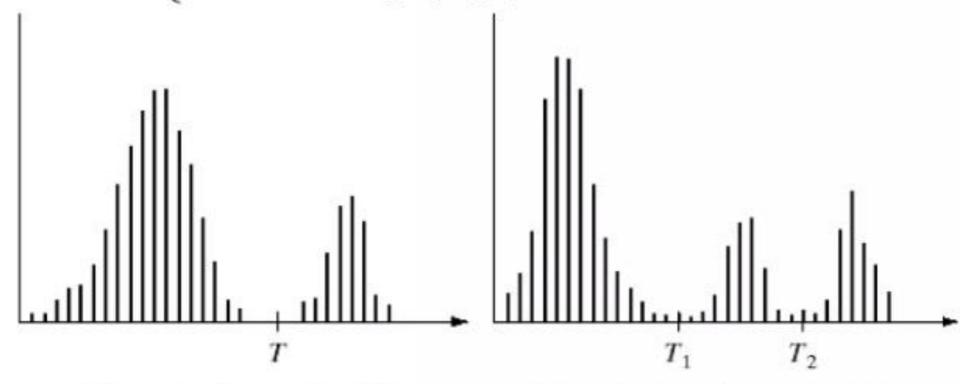


```
import numpy as np
def hough line(img):
 # Rho and Theta ranges
  thetas = np.deg2rad(np.arange(-90.0, 90.0))
 width, height = img.shape
  diag_len = np.ceil(np.sqrt(width * width + height * height)) # max dist
  rhos = np.linspace(-diag len, diag len, diag len * 2.0)
 # Cache some resuable values
  cos_t = np.cos(thetas)
  sin t = np.sin(thetas)
  num thetas = len(thetas)
```

```
# Hough accumulator array of theta vs rho
accumulator = np.zeros((2 * diag_len, num_thetas), dtype=np.uint64)
y_idxs, x_idxs = np.nonzero(img) # (row, col) indexes to edges
# Vote in the hough accumulator
for i in range(len(x_idxs)):
  x = x_idxs[i]
  y = y idxs[i]
  for t_idx in range(num_thetas):
    # Calculate rho. diag_len is added for a positive index
    rho = round(x * cos_t[t_idx] + y * sin_t[t_idx]) + diag_len
    accumulator[rho, t idx] += 1
return accumulator, thetas, rhos
```

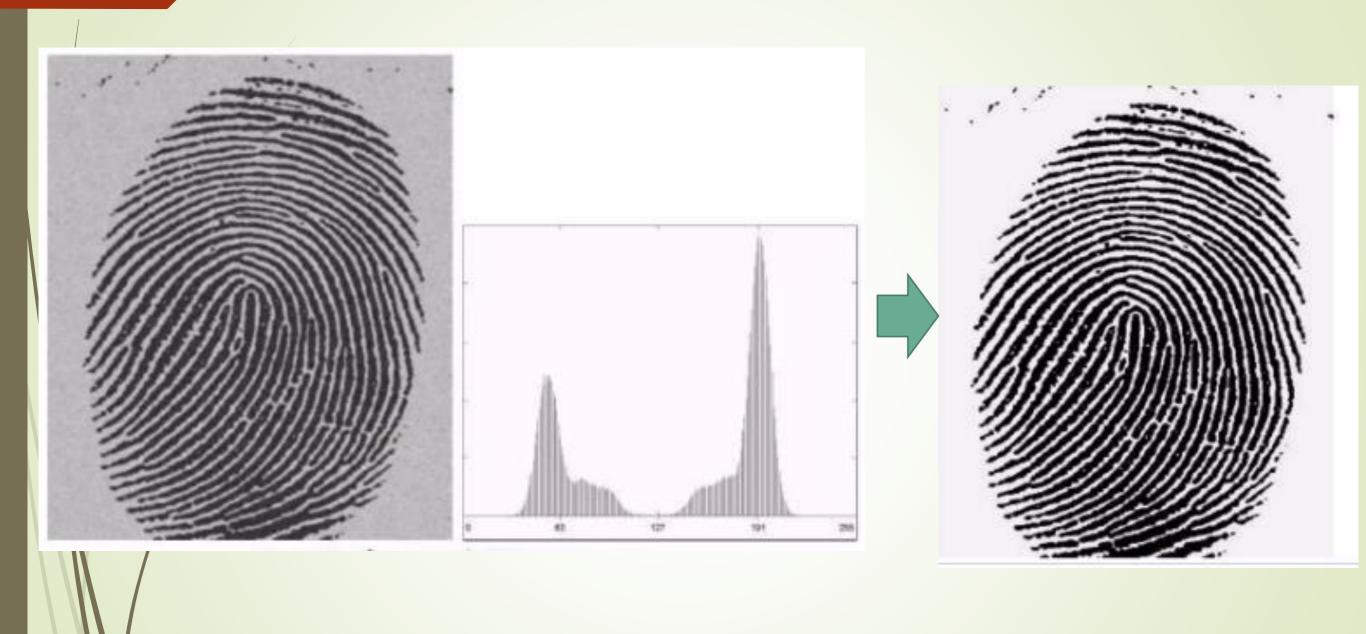
Assumption: the range of intensity levels covered by objects of interest is different from the background.

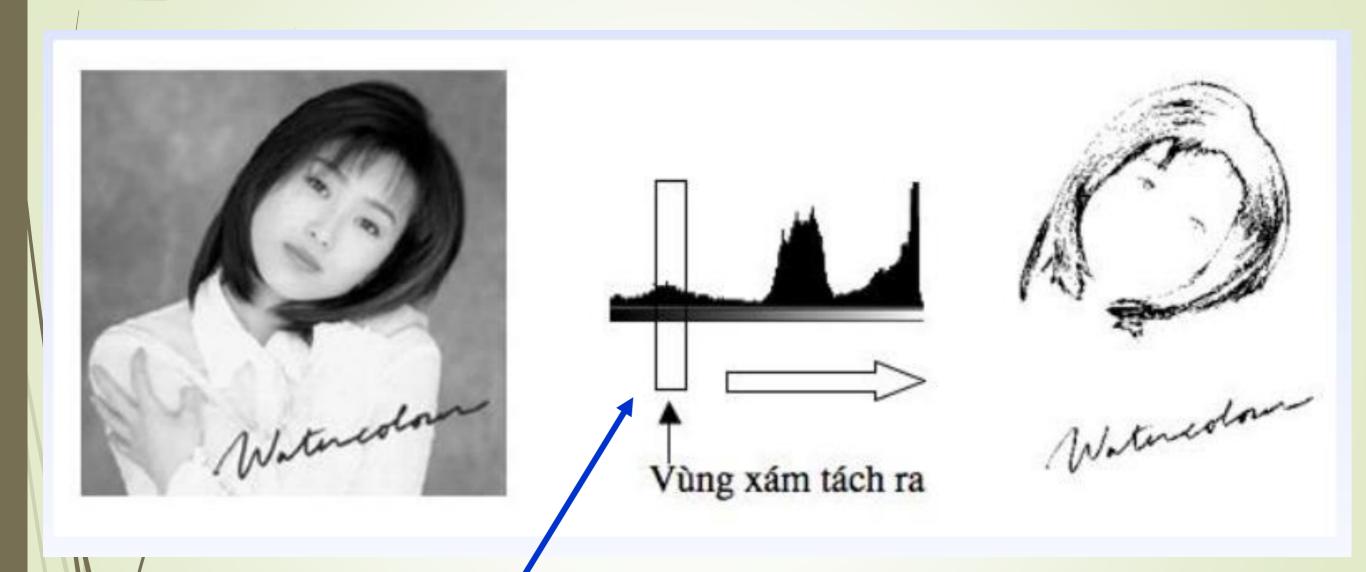
$$g(x,y) = \begin{cases} 1 & \text{if } f(x,y) > T \\ 0 & \text{if } f(x,y) \le T \end{cases}$$



Single threshold

Multiple threshold

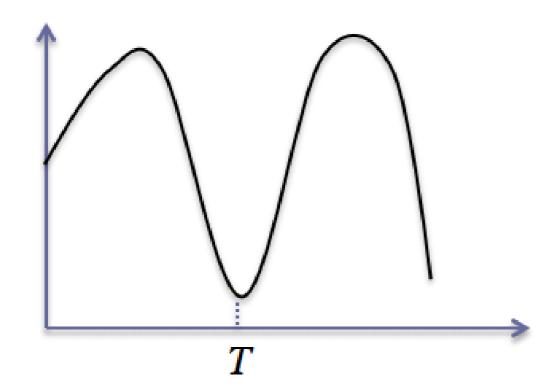




Histogram

Ngưỡng thích nghi

- 1) Chọn giá trị khởi tạo T_o
- 2) Chia vùng ảnh dựa theo $T_o(G_1 và G_2)$
- 3) Tính μ_1 và μ_2 tương ứng của G_1 và G_2
- 4) Tính ngưỡng mới $T_1 = (\mu_1 + \mu_2)/2$
- 5) Lặp lại các bước 2-4 đến khi T_i - T_{i-1} <Th



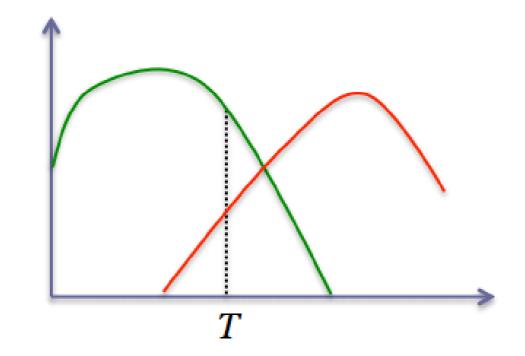
$$g(x,y) = \begin{cases} 1 & \text{If } f(x,y) > T \\ 0 & \text{If } f(x,y) \le T \end{cases}$$

Phân vùng dựa trên hàm xác suất

$$p(z) = P_1 p_1(z) + P_2 p_2(z)$$

$$P_1 + P_2 = 1$$

$$E_1(T) = \int_{-\infty}^{T} p_2(z) dz$$
 $E_2(T) = \int_{-\infty}^{T} p_2(z) dz$



$$E(T) = P_2 E_1(T) + P_1 E_2(T)$$

$$p_i(z) = \frac{1}{\sqrt{2\pi\sigma_i}} e^{-\frac{(z-\mu_i)^2}{2\sigma_i^2}}$$

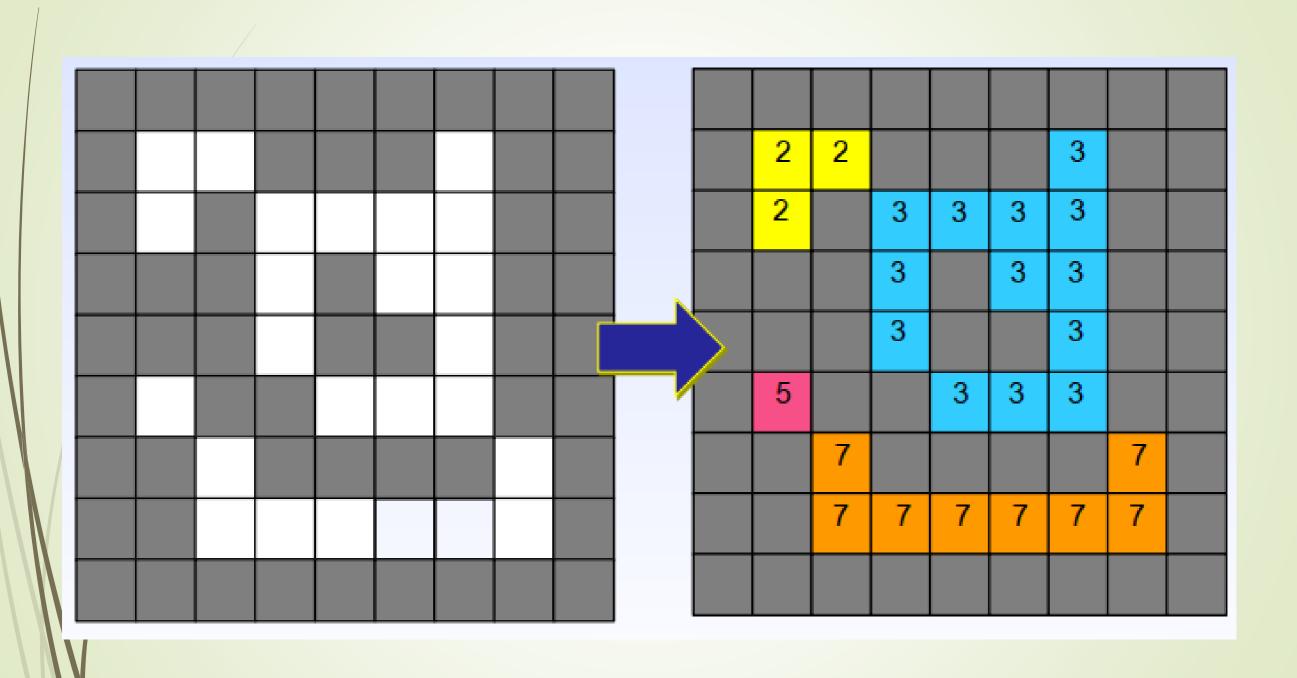
$$\frac{\partial E}{\partial T} = 0 \Rightarrow P_2 p_1(T) = P_1 p_2(T)$$

$$T = \frac{\mu_1 + \mu_2}{2} + \frac{\sigma^2}{\mu_1 - \mu_2} \ln \left(\frac{P_2}{P_1} \right)$$

Phân vùng dựa trên cải thiện histogram

$$s(x,y) = \begin{cases} 0 & \text{If } \nabla f < T \\ + & \text{If } \nabla f \ge T \text{ and } \nabla^2 f \ge 0 \\ - & \text{If } \nabla f \ge T \text{ and } \nabla^2 f < 0 \end{cases}$$

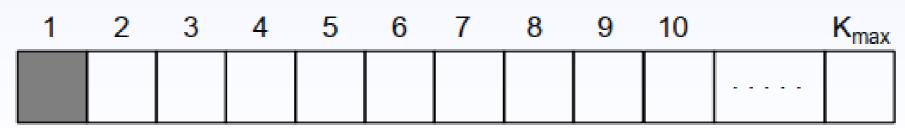
- Điểm ảnh không thuộc biên ảnh hoặc lân cận của biên ảnh được ký hiệu o. Điểm ảnh lân cận của biên ảnh thuộc vùng tối được ký hiệu +. Điểm ảnh lân cận của biên ảnh thuộc vùng sáng được ký hiệu -.
- Biên ảnh tại những điểm có bước chuyển (-,+) hoặc (+,-).



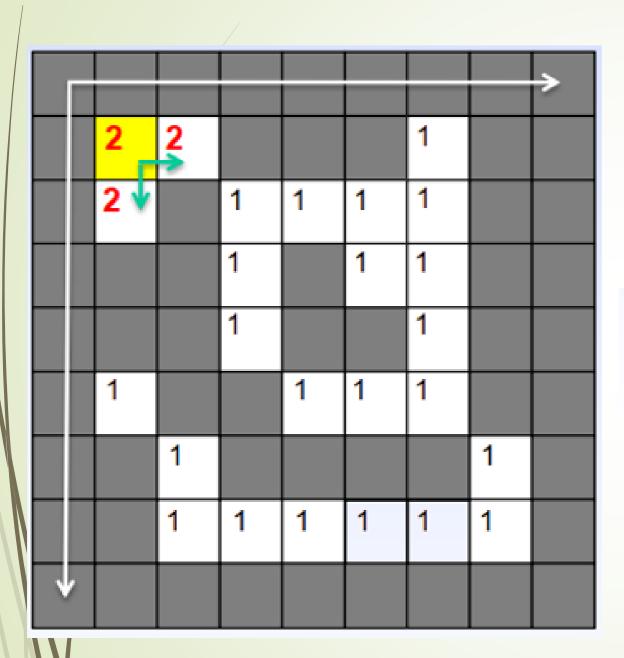
Các điểm chưa dán nhãn có giá trị bằng 1 (điểm nền giá trị bằng 0)

1	1				1		
1		1	1	1	1		
		1		1	1		
		1			1		
1			1	1	1		
	1					1	
	1	1	1	1	1	1	

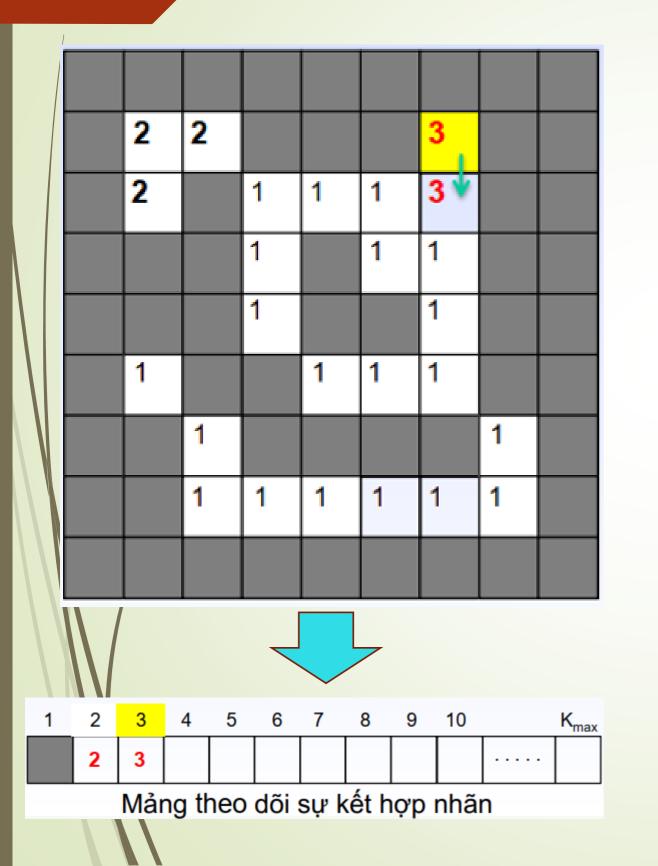
Ånh cần phân vùng bằng thuật toán CCL

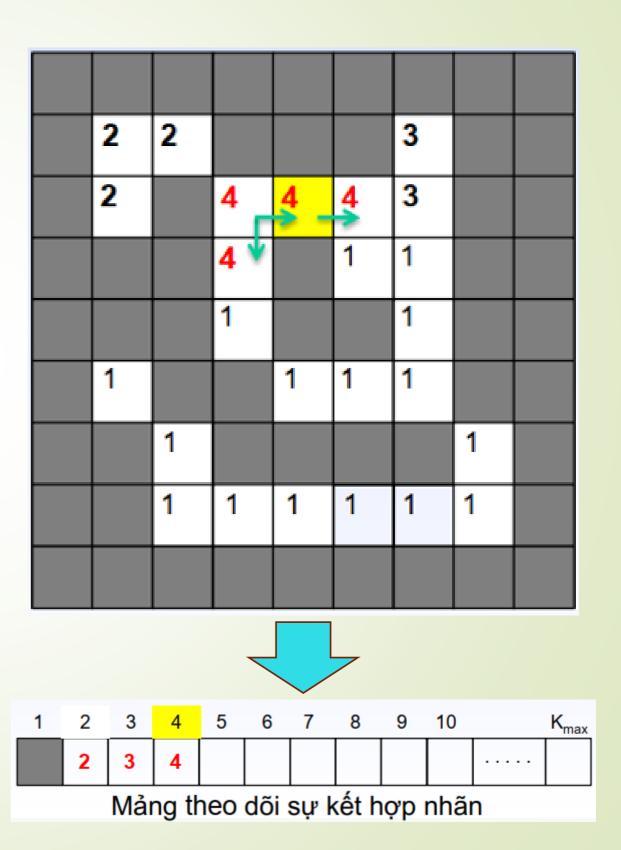


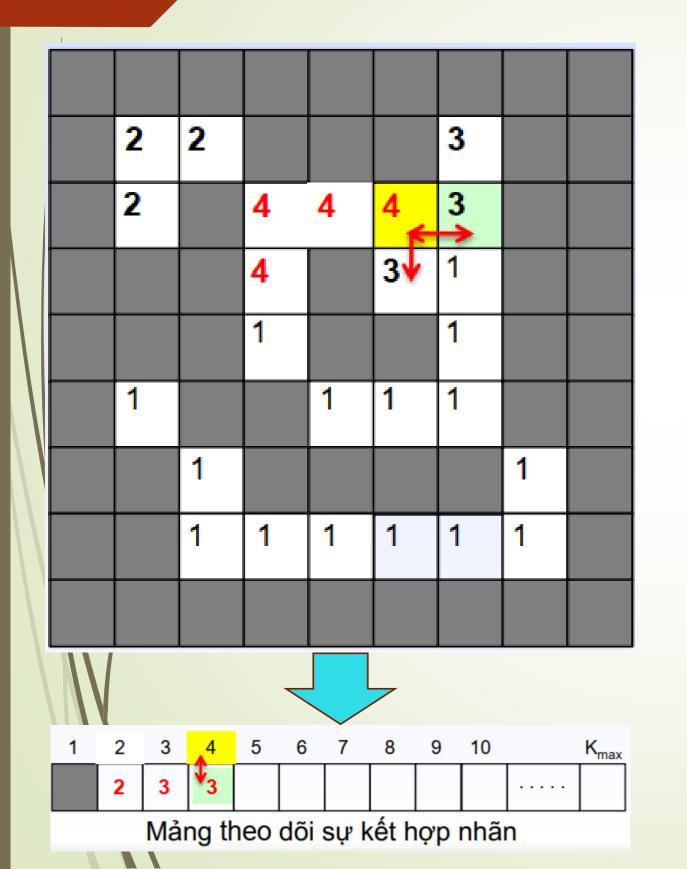
Mảng theo dõi sự kết hợp nhãn

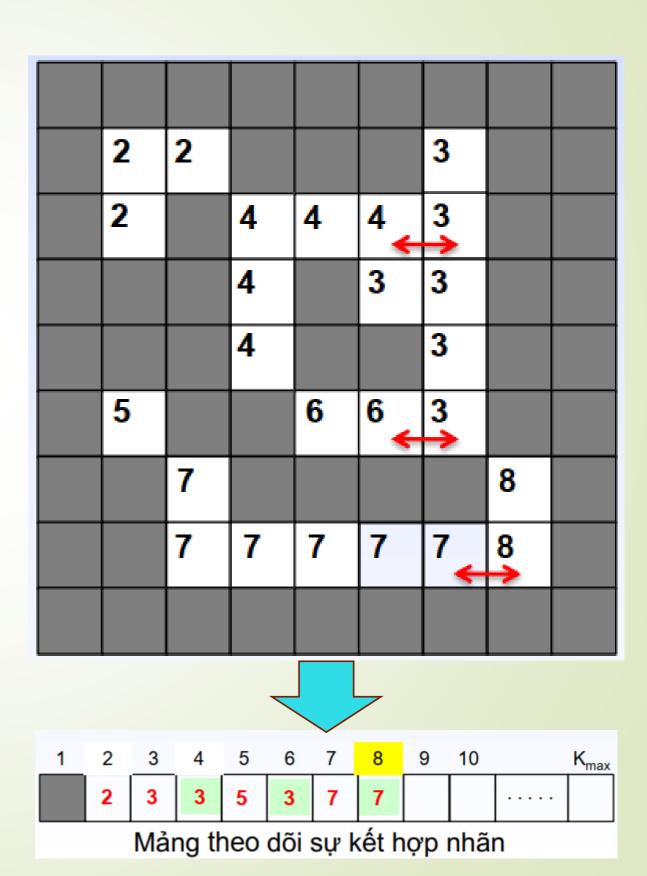


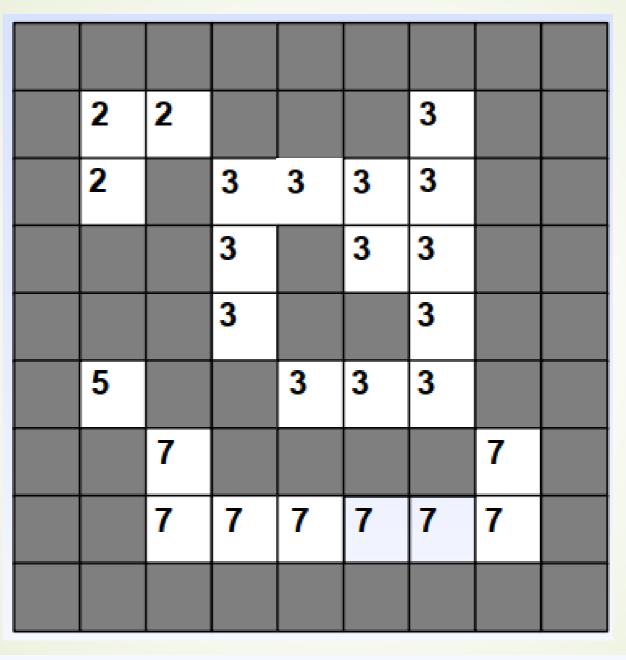
1	2	3	4	5	6	7	8	9	10		K _{max}
	2										
Mảng theo dõi sư kết hợp nhãn											













Mảng theo dõi sự kết hợp nhãn

Ứng dụng

