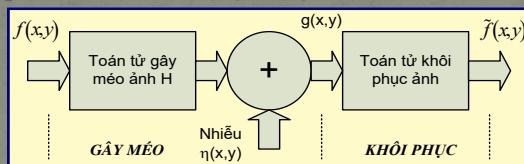


❖ Khôi phục ảnh

Khôi phục ảnh là quá trình loại bỏ hay tối thiểu hóa các ảnh hưởng của môi trường bên ngoài đến ảnh thu nhận được

Mô hình gây méo/khôi phục ảnh



Nếu H là *toán tử tuyến tính và bất biến*, thì ảnh bị nhiễu $g(x,y)$ có dạng như sau:

$$g(x,y) = H[f(x,y)] + \eta(x,y) = h(x,y) * f(x,y) + \eta(x,y)$$

$h(x,y)$ - hàm đáp ứng đặc trưng cho toán tử H trong không gian.

Trong miền tần số, dựa trên tính chất của biến đổi Fourier ta có:

$$G(u,v) = H(u,v) \cdot F(u,v) + N(u,v)$$

với $G(u,v), H(u,v), F(u,v), N(u,v)$ là kết quả biến đổi Fourier của các hàm tương ứng.

• Kỹ thuật lọc ngược (inverse filter)

$$g(x,y) = H[f(x,y)] + \eta(x,y)$$

Xét trường hợp tín hiệu bị làm méo bởi toán tử H tuyến tính, bất biến:

$$g(x,y) = h(x,y) * f(x,y) + \eta(x,y)$$

$$G(u,v) = H(u,v) \cdot F(u,v) + N(u,v)$$

Khi hàm $H(u,v)$ được cho trước, nhiễu bằng không, kỹ thuật khôi phục chính xác nhất là **lọc ngược**. Hàm đáp ứng tần số của bộ lọc ngược là $1/H(u,v)$.

Phổ của ảnh kết quả $\hat{F}(u,v)$ sẽ bằng:

$$\hat{F}(u,v) = F(u,v) / H(u,v)$$

Nếu ảnh bị tác động bởi nhiễu cộng, ta có:

$$\hat{F}(u,v) = F(u,v) + N(u,v) / H(u,v)$$

Có thể thấy rằng, ngay cả khi biết trước $H(u,v)$, chúng ta vẫn không thể khôi phục chính xác được ảnh gốc, vì nhiễu cộng là tín hiệu ngẫu nhiên và hàm mật độ phổ $N(u,v)$ không xác định chính xác được.

• Lọc Wiener

Bộ lọc Wiener cho phép khôi phục ảnh dựa trên hàm biến đổi $H(u,v)$ và đặc điểm của nguồn nhiễu.

Trong kỹ thuật lọc Wiener, ảnh số $f(u,v)$ và nhiễu được mô tả như các tín hiệu ngẫu nhiên; Tín hiệu ở đầu ra của bộ lọc là $\hat{f}(u,v)$ phải đáp ứng được điều kiện sao cho **lỗi trung bình bình phương** (MSE - mean square error) của $\hat{f}(u,v)$ và $f(u,v)$ phải là nhỏ nhất. Giá trị MSE tính theo công thức sau:

$$e^2 = E\left\{\left(f - \hat{f}\right)^2\right\}$$

Xét trường hợp thỏa mãn các điều kiện sau:

- Nhiều và ảnh gốc là hai tín hiệu ngẫu nhiên độc lập tương hỗ (không tương quan);
- Hoặc ảnh gốc, hoặc nhiễu có giá trị trung bình bằng 0;
- Hàm biến đổi $H(u,v)$ là tuyến tính

khi đó, có thể chứng minh được MSE sẽ là nhỏ nhất khi đáp ứng của bộ lọc được xác định theo biểu thức sau:

$$\begin{aligned}\hat{F}(u,v) &= \left(\frac{H^*(u,v) S_f(u,v)}{S_f(u,v)|H(u,v)|^2 + S_\eta(u,v)} \right) G(u,v) = \\ &= \left(\frac{1}{H(u,v)} \cdot \frac{|H(u,v)|^2}{|H(u,v)|^2 + S_\eta(u,v) / S_f(u,v)} \right) G(u,v)\end{aligned}$$

$S_\eta(u,v) = |N(u,v)|^2$ - Hàm mật độ phổ công suất của nhiễu;

$S_f(u,v) = |F(u,v)|^2$ - Hàm mật độ phổ công suất của tín hiệu ảnh gốc;

$S_\eta(u,v) / S_f(u,v)$ - Tỷ lệ công suất nhiễu/tín hiệu

$G(u,v)$ - Phổ của ảnh đã bị biến đổi và chịu tác động của nhiễu.

Nếu nhiễu không tồn tại trong hệ thống, mật độ phổ công suất $S_\eta(u,v) = 0$, bộ lọc Wiener sẽ biến thành bộ lọc ngược.

Hai đại lượng quan trọng là **công suất trung bình** của ảnh gốc và của nhiễu là:

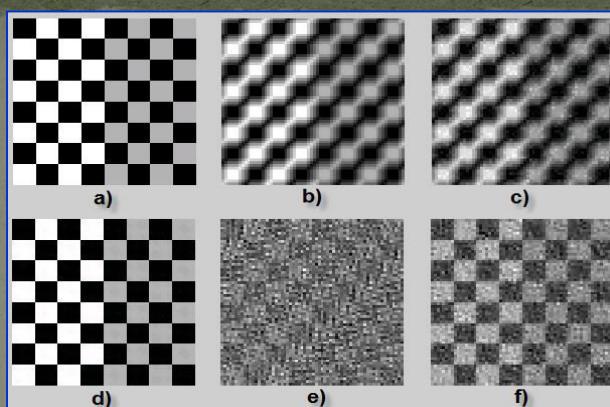
$$f_A = \frac{1}{MN} \sum_u \sum_v S_f(u,v)$$

$$\eta_A = \frac{1}{MN} \sum_u \sum_v S_\eta(u,v)$$

MxN là kích thước ma trận ảnh và nhiễu.

Khi không thể tìm được hàm mô tả **mật độ phổ công suất** của nhiễu cũng như của tín hiệu hình ảnh ta có thể dùng tỷ lệ $R = \eta_A / f_A$ (là hằng số) thay cho tỷ lệ NSPR. Phổ ảnh kết quả được mô tả bằng biểu thức sau:

$$\hat{F}(u,v) = \left(\frac{1}{H(u,v)} \cdot \frac{|H(u,v)|^2}{|H(u,v)|^2 + R} \right) G(u,v)$$



`h = fspecial('motion', len, theta)`

returns a filter to approximate, once convolved with an image, the linear motion of a camera.

- a) Ảnh gốc b) Ảnh nhòe khi camera dịch chuyển
- c) ảnh nhòe và nhiễu Gaussian d) Kết quả lọc ngược từ ảnh (b)
- e) Ảnh kết quả lọc ngược ảnh có nhiễu cộng
- f) là kết quả thực hiện lọc Wiener

$$R = \eta_A / f_A = 0,07$$

Chương 5. Phân đoạn ảnh

5.1. Khái niệm cơ bản

5.2. Phát hiện cạnh, đường, điểm trong ảnh

5.2.1. Phát hiện điểm

5.2.2. Phát hiện đường

5.2.3. Mô hình cạnh

5.2.4. Phát hiện cạnh cơ bản

5.2.5. Những kỹ thuật nâng cao trong phát hiện cạnh

5.2.6. Liên kết những điểm cạnh

Chương 5. Phân đoạn ảnh

5.1. Khái niệm cơ bản

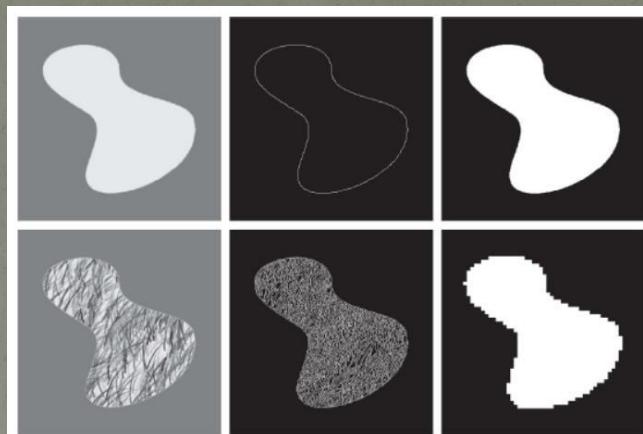
Phân đoạn ảnh là phương pháp XLA mức trung và mức cao, tín hiệu ở đầu ra ko phải là ảnh mà là một số thông tin về hình ảnh.

Sau khi phân đoạn, các đối tượng cần quan tâm sẽ được tách rời và **mô tả độc lập**.

Các giải thuật phân đoạn thường dựa trên tính chất của sự thay đổi độ chói màu sắc của các chi tiết trong ảnh hoặc dựa trên mức độ đồng nhất của vùng ảnh.

1- Phân đoạn dựa trên **tốc độ biến thiên** của độ chói hay màu sắc của ảnh

2- Phân đoạn dựa trên **tính chất không đổi** được lựa chọn trước của vùng ảnh cần quan tâm



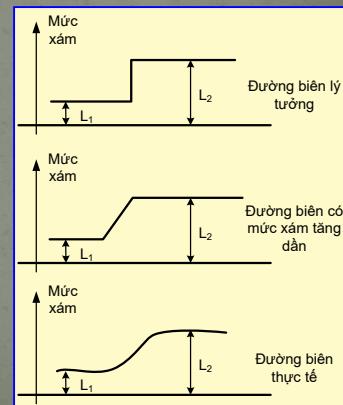
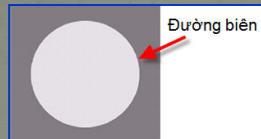
Phân đoạn dựa trên xác định đường biên và dựa trên tính chất vùng ảnh
(vùng ảnh có độ chói không đồng đều)

5.2. Phát hiện điểm, đường biên trong ảnh

Đường biên của một vùng ảnh R được tạo ra bởi các điểm ảnh có một hoặc nhiều điểm lân cận không nằm trong tập liên kết R

Đường biên là đường nối các điểm ảnh nằm trong khu vực ảnh có thay đổi đột ngột về độ chói

Đường biên thường ngăn cách hai vùng ảnh, khi mỗi vùng có các mức xám gần như không đổi.

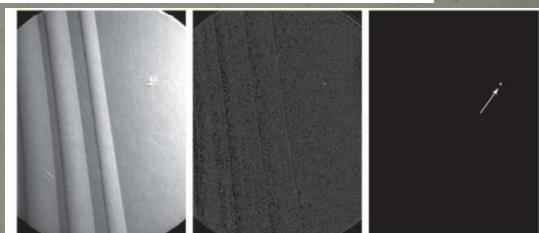


5.2.1 Phát hiện điểm ảnh

$$\nabla^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

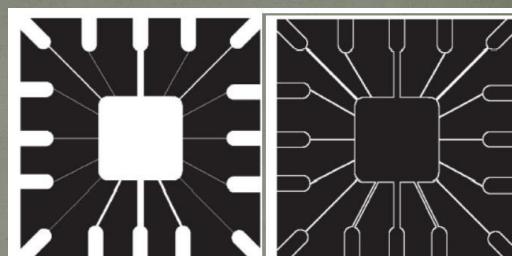
$$\nabla^2 f(x, y) = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y)$$

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



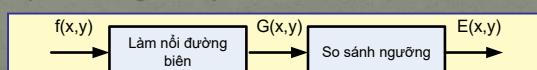
5.2.2 Phát hiện đường biên

$$\nabla^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

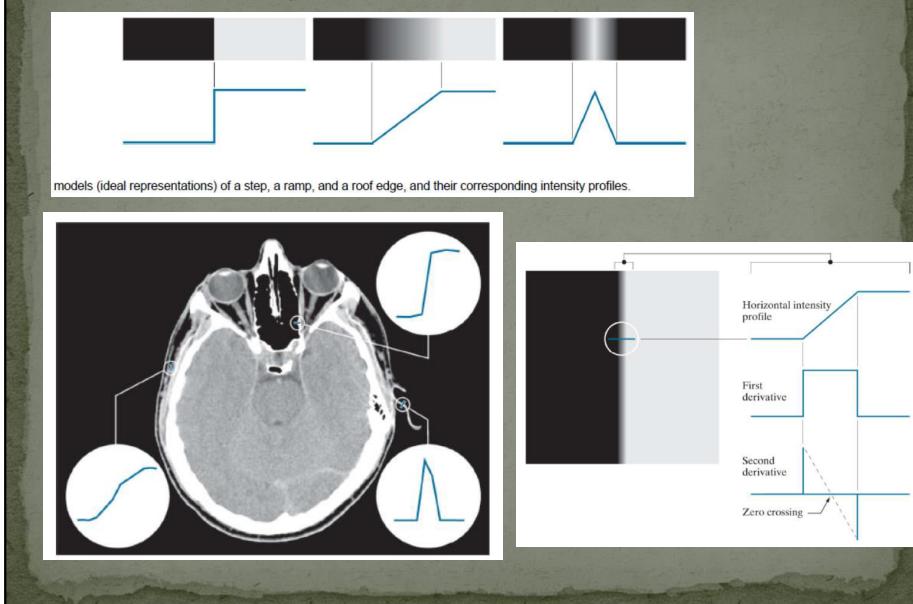


1- Phương pháp phát hiện đường biên **trực tiếp** dựa trên các phân tích về sự thay đổi độ chói của ảnh. Kỹ thuật chủ yếu dùng để phát hiện biên là dùng đạo hàm. Khi lấy đạo hàm bậc nhất của ảnh ta có phương pháp gradient, khi lấy đạo hàm bậc hai ta có kỹ thuật Laplace.

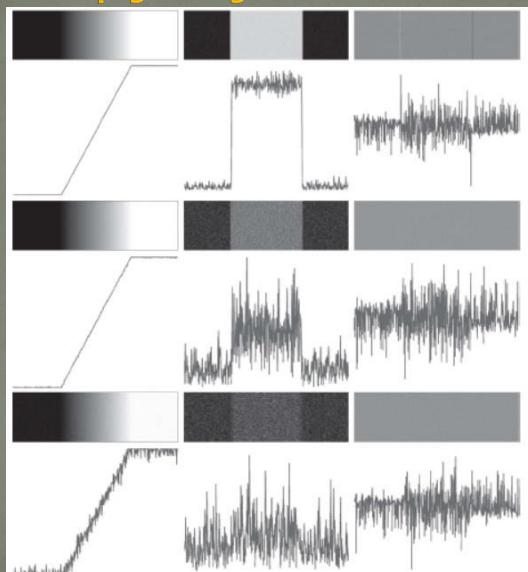
2-Mô tả (xấp xỉ đường biên)



5.2.3. Mô hình cạnh



Các dạng đường biên



Đường biên trên ảnh có nhiều

5.2.4. Phát hiện cạnh cơ bản

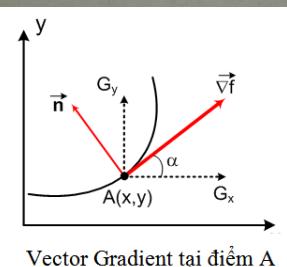
➤ Phương pháp Gradient

Phương pháp gradient là phương pháp dò biên cục bộ dựa vào giá trị cực đại của đạo hàm bậc một. **Gradient** của ảnh $f(x,y)$ tại điểm A là **vector** ∇f . Hướng của vector gradient trùng với hướng có tốc độ thay đổi lớn nhất của hàm $f(x,y)$.

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

$$G_x = \frac{\partial f(x,y)}{\partial x} = \frac{f(x+dx,y) - f(x,y)}{dx}$$

$$G_y = \frac{\partial f(x,y)}{\partial y} = \frac{f(x,y+dy) - f(x,y)}{dy}$$



$$\text{Modul gradient: } |\nabla f| = \sqrt{G_x^2 + G_y^2}$$

$$\text{hoặc } |\nabla f| \approx |G_x| + |G_y|$$

Hướng của vector gradient

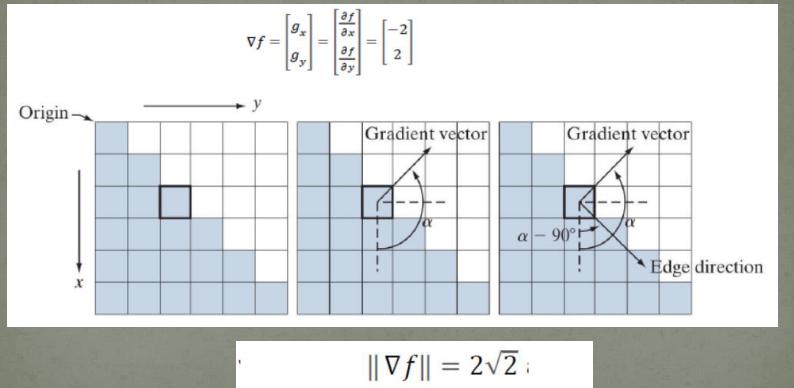
là góc $\alpha(x,y)$ giữa vector ∇f và trục x:

$$\alpha(x,y) = \arctg \left(\frac{G_y}{G_x} \right)$$

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

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

-1	-1	1
1		



Ma trận 9 điểm ảnh và các
mặt nạ dùng để tìm
gradient tại điểm z_5

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

a- Ma trận các điểm ảnh

-1	0
0	-1
1	0
0	1

b- Mặt nạ Robert

-1	-1	-1
0	0	0
1	1	1

c - Mặt nạ Prewitt

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

d- mặt nạ Sobel

Toán tử Robert

$$\begin{aligned} G_1 &= z_9 - z_5 \\ G_2 &= z_8 - z_6 \end{aligned}$$

Toán tử Prewitt

$$\begin{aligned} G_x &= (z_7 + z_8 + z_9) - (z_1 + z_2 + z_5) \\ G_y &= (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7) \end{aligned}$$

Toán tử Sobel

$$\begin{aligned} G_x &= (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3) \\ G_y &= (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7) \end{aligned}$$



Kết hợp gradient và lấy ngưỡng

Toán tử Sobel

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

mặt nạ Sobel

$G_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$

$G_y = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$

(6) Нарисуйте гистограмму направлений перепадов, вычисленных с помощью соотношения (10.1-5). Укажите точные значения высот всех пиков гистограммы.

Xác định gradient tại các điểm trên hình. Gradient nhận được dựa trên toán tử Sobel

$\nabla f \approx |G_x| + |G_y|$.

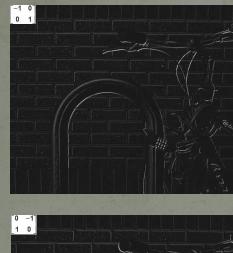
$\alpha(x, y) = \arctg\left(\frac{G_y}{G_x}\right)$

0	0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0	0	[1,1] [3,1] [4,0] [4,0] 2 4 4 4 . . . 4 4 4 4 2 0 0	[4,0] [4,0] [3,-1] [1,-1] 4 4 4 4 2 0 0	[4,0] [3,-3] [1,-3]
0	0	[1,3] [3,3] [4,0] 4 6 4 4 . . . 4 4 6 4 0 0	[0,0] [0,-4] [0,-4]	[0,0] [0,-4] [0,-4]
0	0	[0,4] [0,4] [0,0] 4 4 0 0 . . . 0 0 4 4 0 0	[0,0] [0,-4] [0,-4]	[0,-4] [0,-4]
0	0	[0,4] [0,4] 4 4 0 0 . . . 0 0 4 4 0 0	[0,0] [0,-4] [0,-4]	[0,-4] [0,-4]
⋮	⋮	⋮	⋮	⋮
0	0	[0,4] [0,4] 4 4 0 0 . . . 0 0 4 4 0 0	[0,0] [0,-4] [0,-4]	[0,-4] [0,-4]
0	0	[0,4] [0,4] [0,0] 4 4 0 0 . . . 0 0 4 4 0 0	[0,0] [0,-4] [0,-4]	[0,-4] [0,-4]
0	0	[1,-3] [-3,-3] [-4,0] 6 6 4 4 . . . 4 4 6 4 0 0	[-4,0] [-3,-3] [-1,-3]	[-4,0] [-3,-3] [-1,-3]
0	0	[-1,1] [-3,1] [-4,0] 2 4 4 4 . . . 4 4 4 4 2 0 0	[-4,0] [-4,0] [-3,-1] [-1,-1]	[-4,0] [-4,0] [-3,-1] [-1,-1]
0	0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

												n pixels												
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	-1	-1	-1	.	.	.	-1	-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	-1	2	1	1	.	.	.	1	1	2	-1	0	0	0	0	0	0	0	0	0	0	0	
0	0	-1	1	0	0	.	.	.	0	0	1	-1	0	0	0	0	0	0	0	0	0	0	0	
0	0	-1	1	0	0	.	.	.	0	0	1	-1	0	0	0	0	0	0	0	0	0	0	0	
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
0	0	-1	1	0	0	.	.	.	0	0	1	-1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	-1	1	0	0	.	.	.	0	0	1	-1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	-1	2	1	1	.	.	.	1	1	2	-1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	-1	-1	-1	.	.	.	-1	-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

(c)

Toán tử Robert



Toán tử Prewitt



Toán tử Sobel



Tổng các hệ số trong các mặt nạ nói trên bằng 0, như vậy tại những vùng ảnh có độ chói không đổi, đáp ứng của bộ lọc sẽ bằng 0, điều này hoàn toàn phù hợp với tính chất của toán tử tìm đạo hàm.

Toán tử Prewitt và Sobel được sử dụng rộng rãi hơn so với Robert vì chúng ít nhạy cảm với nhiễu nhờ tháo tác lấy trung bình cục bộ trong vùng rộng hơn (9 điểm ảnh).

Hai cặp mặt nạ bổ xung Prewitt và Sobel
dùng để làm nổi biên theo đường chéo

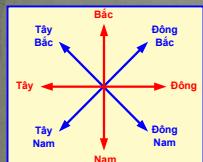
0	1	1	-1	-1	0
-1	0	1	-1	0	1
-1	-1	0	0	1	1

Cặp mặt nạ Prewitt (bổ xung)

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

Cặp mặt nạ Sobel (bổ xung)

➤Làm nổi biên bằng toán tử la bàn



Bắc	Đông Bắc	Đông	Đông Nam
1 1 1	1 1 1	-1 1 1	-1 -1 1
1 -2 1	-1 -2 1	-1 -2 1	-1 -2 1
-1 -1 -1	-1 -1 1	-1 1 1	1 1 1
Nam	Tây Nam	Tây	Tây Bắc
-1 -1 -1	1 -1 -1	1 1 -1	-1 1 1
1 -2 1	1 -2 -1	1 1 -1	1 -2 -1
1 1 1	1 1 1	1 1 -1	1 -1 -1

Mặt nạ dùng trong toán tử la bàn



Mặt nạ Đông Bắc

Mặt nạ Đông



Toán tử la bàn Kirsh cũng cho phép làm nổi biên theo tám hướng. 8 mặt nạ Kirsh có dạng như sau:

Bắc	Đông Bắc	Đông	Đông Nam
5 5 5	-3 5 5	-3 -3 5	-3 -3 -3
-3 0 -3	-3 0 5	-3 0 5	-3 0 5
-3 -3 -3	-3 -3 -3	-3 -3 5	-3 5 5
Nam	Tây Nam	Tây	Tây Bắc
-3 -3 -3	-3 -3 -3	5 -3 -3	5 5 -3
-3 0 -3	5 0 -3	5 0 -3	5 0 -3
5 5 5	5 5 -3	5 -3 -3	-3 -3 -3

>Kỹ thuật Laplace

Khi mức xám thay đổi chậm, các đường biên không rõ nét, miền chuyển tiếp tương đối rộng, phương pháp làm nổi biên hiệu quả hơn là dùng **đạo hàm bậc hai**.

Phương pháp này gọi là **phương pháp Laplace**.

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

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

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

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

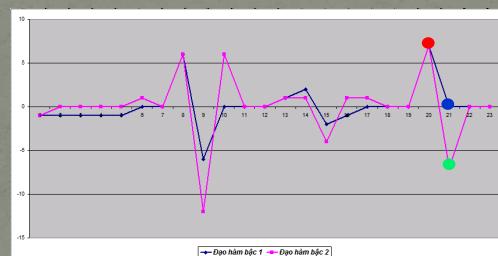
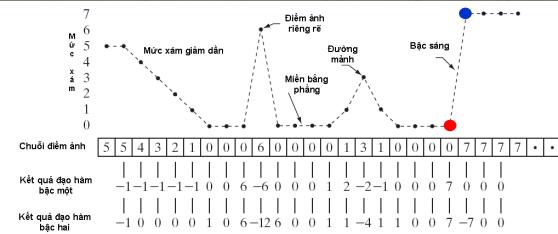
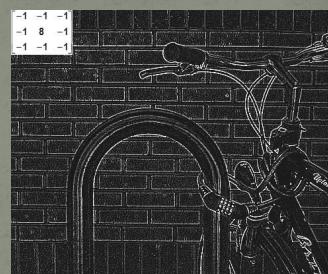
z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

0	-1	0
-1	4	-1
0	-1	0

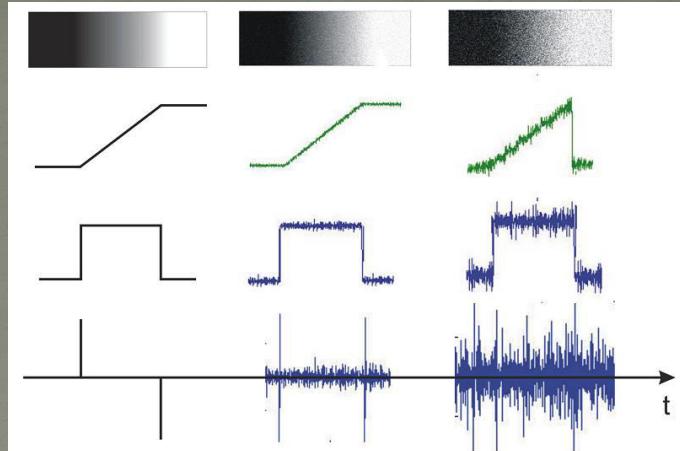
-1	-1	-1
-1	8	-1
-1	-1	-1

$$\nabla^2 f = 4z_5 - [z_2 + z_4 + z_6 + z_8]$$

$$\nabla^2 f = 8z_5 - [z_1 + z_2 + z_3 + z_4 + z_6 + z_7 + z_8 + z_9]$$



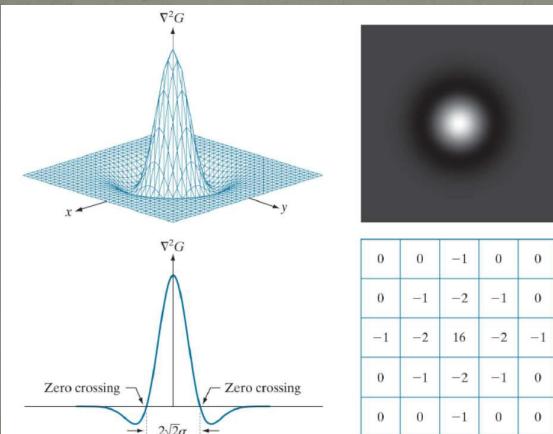
Dạng tín hiệu nhận được sau khi lấy đạo hàm bậc 1 và bậc 2



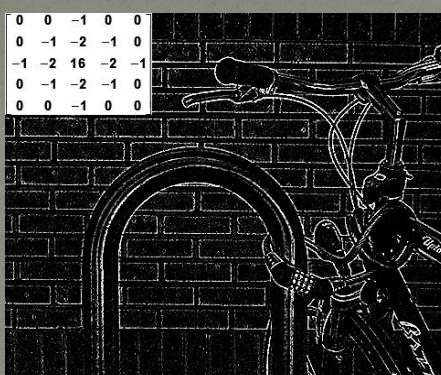
Hiệu ứng đường biên đôi và khả năng chống nhiễu kém của phương pháp tìm biên bằng đạo hàm bậc hai so với đạo hàm bậc một.

Để làm giảm ảnh hưởng của nhiễu, trước khi thực hiện toán tử Laplace người ta thường làm mờ ảnh bằng bộ lọc Gaussian

$$G(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \quad \nabla^2 G(x, y) = \left(\frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} \right) e^{-\frac{x^2+y^2}{2\sigma^2}}$$



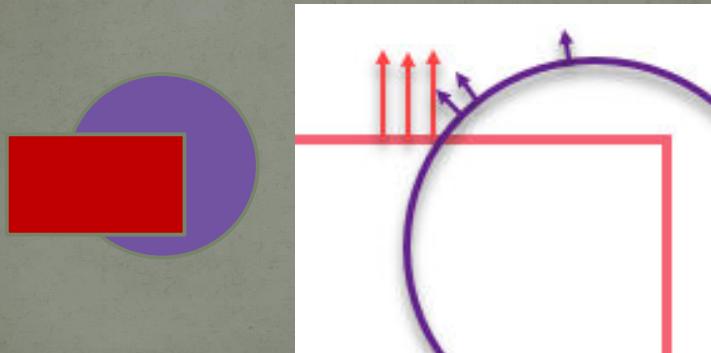
Hàm đáp ứng của bộ lọc Laplacian-Gaussian



5.3 Kết nối các điểm đường biên và xấp xỉ đường biên

- ✓ Mô tả đường biên bằng giải pháp xử lý cục bộ

Phân tích các đặc trưng của điểm ảnh trong vùng lân cận nhỏ (3x3 hoặc 5x5) của mỗi điểm (x,y) trong ảnh cần tách biên. Tất cả các điểm ảnh có chung một đặc điểm lựa chọn trước sẽ được liên kết lại thành **đường biên**.



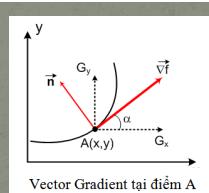
5.3 Kết nối các điểm đường biên và xấp xỉ đường biên

- ✓ Mô tả đường biên bằng giải pháp xử lý cục bộ

Phân tích các đặc trưng của điểm ảnh trong vùng lân cận nhỏ (3x3 hoặc 5x5) của mỗi điểm (x,y) trong ảnh cần tách biên. Tất cả các điểm ảnh có chung một đặc điểm lựa chọn trước sẽ được liên kết lại thành **đường biên**.



$$\nabla \vec{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

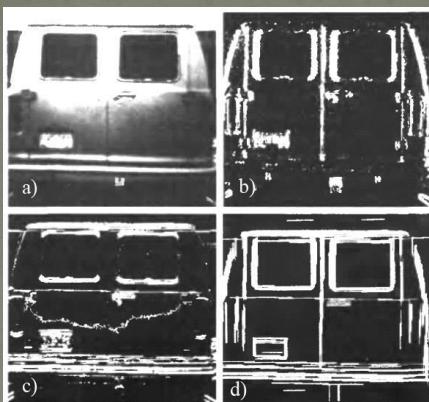


Điểm ảnh biên (x_0, y_0) , nằm trong vùng ảnh xung quanh điểm (x, y) sẽ được coi là “giống” với điểm (x, y) nếu:

$$|\nabla f(x, y) - \nabla f(x_0, y_0)| \leq E \quad E \text{- là } \textcolor{red}{ngưỡng biên độ} \text{ cho trước.}$$

Hướng của vector gradient tại điểm (x_0, y_0) được cho là “trùng” với hướng vector gradient tại điểm (x, y) nếu:

$$|\alpha(x, y) - \alpha(x_0, y_0)| \leq A \quad A \text{- góc lệch được cho trước.}$$

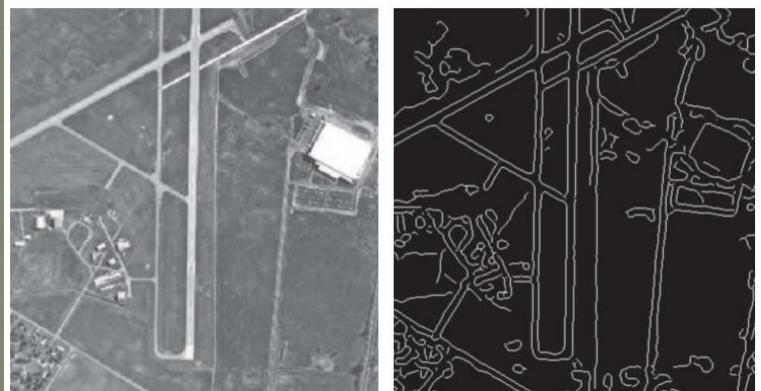


- a) Ảnh gốc.
- b) Thành phần Gradient G_y ,
- c) Thành phần Gradient G_x .
- d) Kết quả liên kết đường biên

Kết quả liên kết đường
biên theo phương pháp
xử lý cục bộ với các giá trị:
 $E=25$; $A=15^\circ$

5.3 Kết nối các điểm đường biên và xấp xi đường biên

✓ Mô tả đường biên bằng giải pháp xử lý cục bộ



Các điểm nằm trên đường biên được xác định không hoàn toàn chính xác

✓ Mô tả đường biên bằng phương pháp xử lý toàn cục (global processing)

Quá trình mô tả đường biên được thực hiện dựa trên cở sở xác định các điểm ảnh có **nằm trên cùng đường cong** (hoặc đường thẳng) cho trước hay không. Chúng ta phải xét đến **quan hệ giữa các điểm ảnh** nằm trong toàn ảnh.

▪ Xấp xi đường biên bằng đa thức

Xét tập hợp các điểm ảnh có tọa độ (x_i, y_i) với $i = 0, 1, 2, \dots, M$ nằm trên ranh giới giữa hai vùng ảnh. Các điểm ảnh được đánh số theo thứ tự sao cho điểm (x_i, y_i) nằm cạnh điểm (x_{i+1}, y_{i+1}) . Quá trình **xấp xi đường cong** trên tập điểm ảnh là quá trình xác định hàm $\hat{y} = g(x)$ sao cho sai số xấp xi, tức độ khác biệt giữa điểm (x_i, y_i) và $[x_i, g(x_i)]$, là nhỏ nhất.

Trong trường hợp tổng quát, ta có thể dùng **đa thức bậc N** để xấp xi đường biên:

$$\hat{y} = a_0 + a_1 x + a_2 x^2 + \dots + a_N x^N$$

a_i - hệ số của đa thức.

Sai số xấp xi có thể được đánh giá theo:

➤ Sai số tuyệt đối: $\varepsilon = \sum_{i=1}^M |y_i - g(x_i)|$

➤ Sai số trung bình bình phương: $\varepsilon = \sum_{i=1}^M [y_i - g(x_i)]^2$

➤ Sai số tối đa: $\varepsilon = \max_i |y_i - g(x_i)|$

$$\hat{y} = a_0 + a_1 x + a_2 x^2 + \dots + a_N x^N$$

$$\begin{bmatrix} 1 & x_0 & x_0^2 & \dots & x_0^N \\ 1 & x_1 & x_1^2 & \dots & x_1^N \\ 1 & x_2 & x_2^2 & \dots & x_2^N \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_M & x_M^2 & \dots & x_M^N \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix} = \begin{bmatrix} \hat{y}_0 \\ \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_N \end{bmatrix}$$

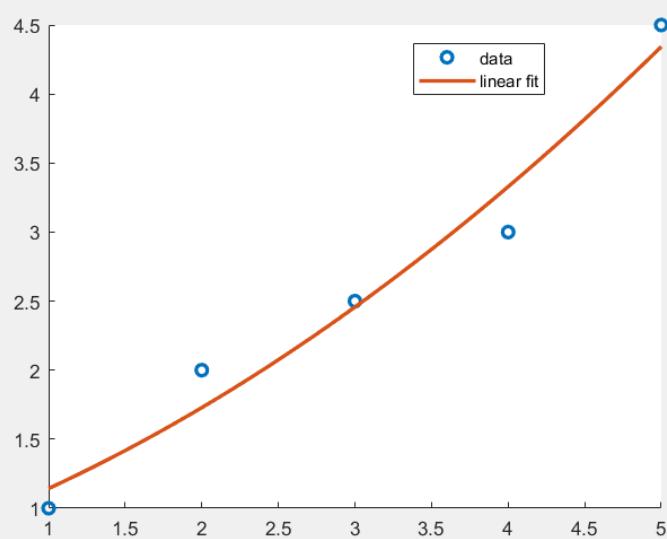
Giải hệ phương trình trên, ta có thể tìm ra được tập hệ số a_i của đa thức \hat{y} .

Đường cong xác định bởi \hat{y} phải cho **sai số xấp xi**, tức độ khác biệt giữa điểm (x_i, y_i) và $[x_i, g(x_i)]$, là nhỏ nhất.

`p = polyfit(x,y,n)` returns the coefficients for a polynomial $p(x)$ of degree n that is a best fit (in a least-squares sense) for the data in y . The coefficients in p are in descending powers, and the length of p is $n+1$

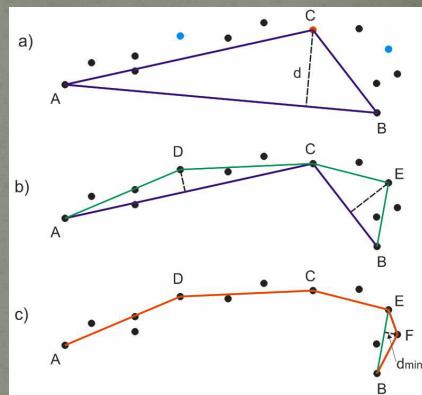
$$p(x) = p_1x^n + p_2x^{n-1} + \dots + p_nx + p_{n+1}.$$

```
x = 1:1:5;
y = [1 2 2.5 3 4.5 ];
[p, mu] = polyfit(x,y,2);
f = polyval(p,x);
```



• Xấp xỉ đường biên bằng các đoạn thẳng

- a) Nối hai điểm ngoài cùng của tập điểm đang xét là A và B bằng một đường thẳng. Tiến hành dò tìm điểm có độ lệch lớn nhất theo hướng trực giao với đường AB (điểm C). Nối điểm A,B và C với nhau.
- b) Lặp lại thao tác trên cho hai đoạn thẳng AC và CB với các điểm ảnh lân cận của chúng.
- c) Quá trình xấp xỉ biên sẽ được kết thúc khi đường thẳng gấp khúc sẽ "gắn giöng" với đường biên thực: cho đến lúc khoảng cách giữa các điểm biên và đường thẳng gấp khúc luôn nhỏ hơn ngưỡng quy định là d_{min} (điểm F).

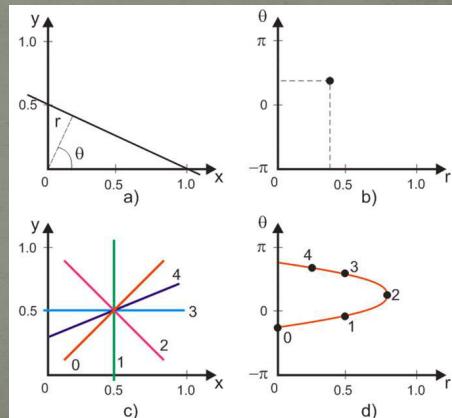


Xấp xỉ đường biên bằng biến đổi Hough (Hough transform)

Một đường thẳng trong mặt phẳng (x,y) có thể được biểu diễn bằng hệ tọa độ góc thông qua **khoảng cách** r từ đường thẳng tới điểm 0 và **góc** θ như trên hình :

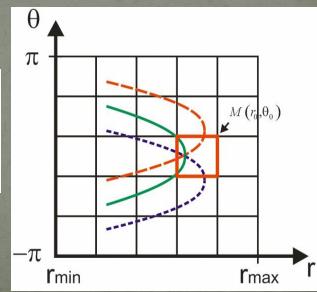
$$r = x \cos \theta + y \sin \theta$$

- a) Đường thẳng trong không gian (x,y) .
- b) Đường thẳng được biểu diễn bằng 1 điểm trong không gian (r,θ) .
- c) Tập các đường thẳng đi qua một điểm
- d) Biến đổi Hough của tập đường thẳng



- e) Các điểm A,B,C nằm trên một đường thẳng có tọa độ góc (r_0, θ_0)
- f) Biến đổi Hough cho tập các đường thẳng chạy qua A,B và C

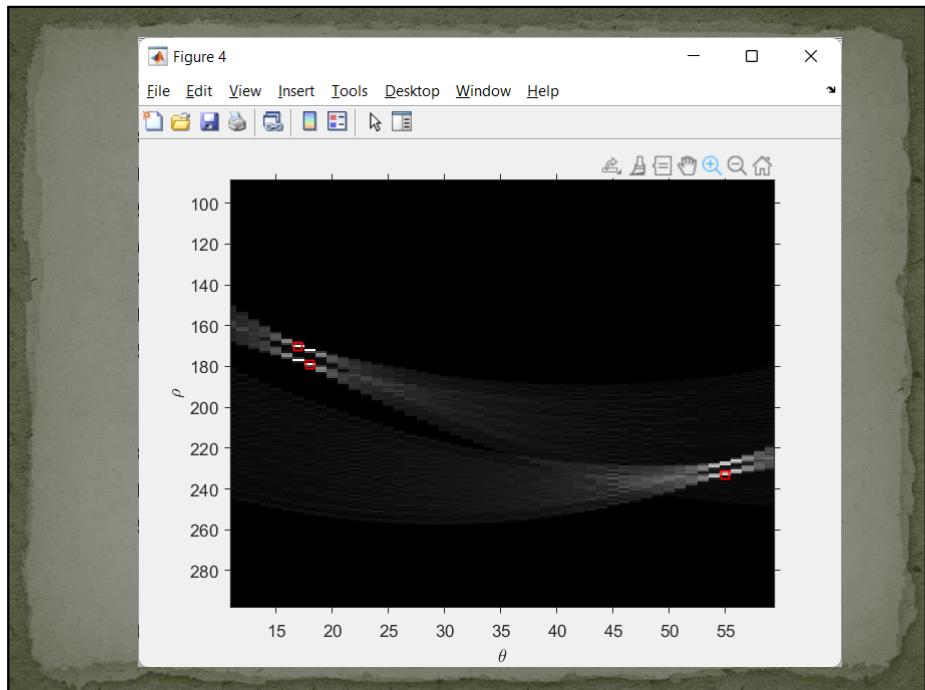
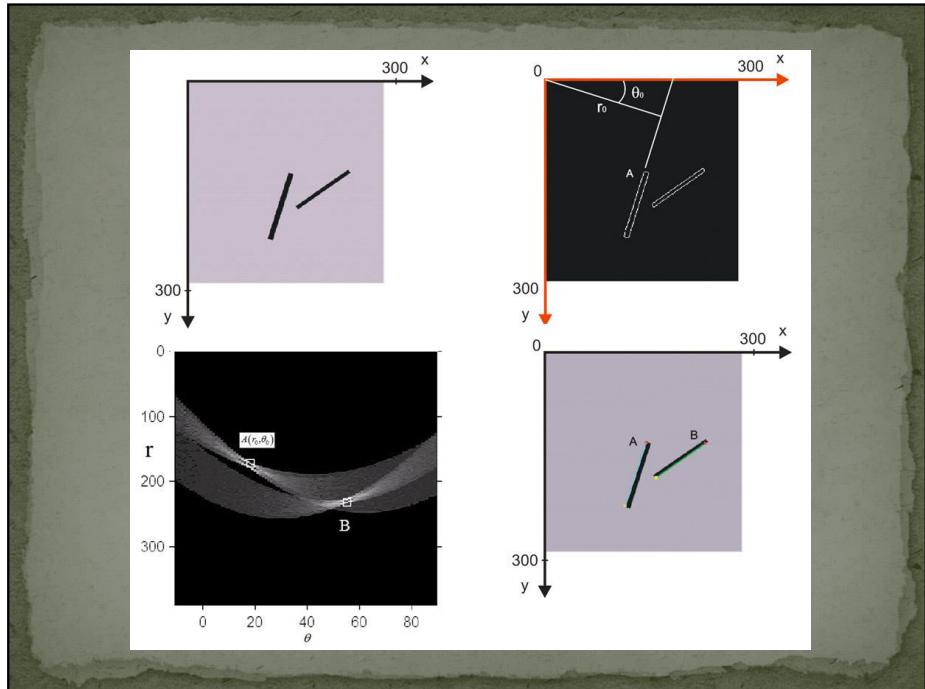
Đối với **các điểm nằm gần đường thẳng AC** trên mặt phẳng (x,y) , biến đổi Hough sẽ cho kết quả là các đường cong cắt nhau trong khu vực lân cận với điểm $X(r_0, \theta_0)$.



Các bước thực hiện quá trình xấp xỉ đường biên với biến đổi Hough:

- 1- Tìm giá trị gradient cho các điểm trên ảnh gốc; So sánh giá trị modul gradient với ngưỡng (cho trước) để tạo ra ảnh nhị phân hiển thị vị trí của các điểm biên.
- 2- Chia mặt phẳng (r, θ) thành các ô nhỏ.
- 3- Tìm ảnh xạ của tất cả các điểm khác 0 trong ảnh nhị phân sang mặt phẳng (r, θ) và tiến hành đánh giá "trọng lượng" từng ô trong mặt phẳng đó.
- 4- Chọn ra các ô có trọng lượng lớn nhất; tìm ra các điểm ảnh biên có thể xấp xỉ bằng đường thẳng có cùng tọa độ góc với tọa độ của ô có trọng lượng lớn, tương ứng với những đường cong đi qua ô đó.

```
I = imread('circuit.tif');
BW = edge(imrotate(I,0,'crop'),'canny');
[H,T,R] = hough(BW);
P = houghpeaks(H,2,'Threshold',15);
figure
imshow(H,[],'XData',T,'YData',R,'InitialMagnification','fit');
xlabel('\theta'), ylabel ('\rho');
axis on, axis normal, hold on;
plot(T(P(:,2)),R(P(:,1))), 's', 'color', 'white');
```



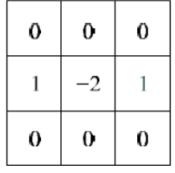
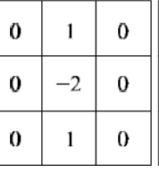
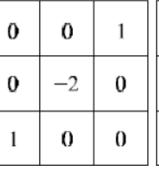
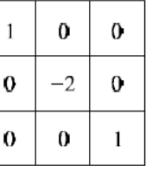
Bài tập

Ảnh nhị phân chứa các đường thẳng tạo ra các góc $0^\circ, +45^\circ, -45^\circ, 90^\circ$.
 Tìm 4 mặt nạ lọc để phát hiện các điểm đứt quãng (có chiều dài 1' điểm ảnh) trên các đường thẳng. Điểm ảnh trên đường thẳng có giá trị bằng 1, ảnh nền có giá trị bằng 0.

Bài tập

Ảnh nhị phân chứa các đường thẳng tạo ra các góc 0° , $+45^\circ$, -45° , 90° .
Tìm 4 mặt nạ lọc để phát hiện các điểm đứt quãng (có chiều dài 1' điểm ảnh) trên các đường thẳng. Điểm ảnh trên đường thẳng có giá trị bằng 1, ảnh nền có giá trị bằng 0.

Các bộ lọc dưới đây sẽ cho kết quả lọc bằng 0 khi đường thẳng không có điểm đứt quãng và cho giá trị bằng 2 tại các điểm đứt quãng

Ngang	Dọc	$+45^\circ$	-45°
			

*10.13 Используя материал Раздела 10.2.2:

- (a) Объясните, почему в результате преобразования Хафа на Рис. 10.20(б) точка I отображается в прямую.

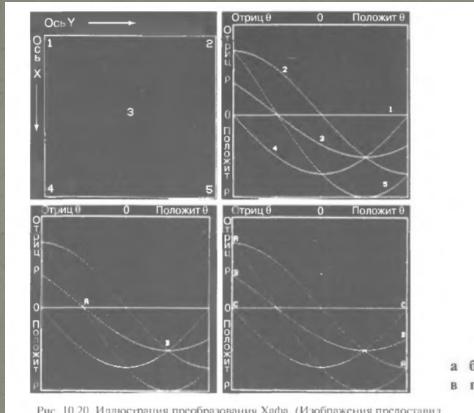


Рис. 10.20. Иллюстрация преобразования Хафа. (Изображения предоставлены Д. Р. Кент, компания Texas Instruments, Inc.).

(a) Point 1 has coordinates $x = 0$ and $y = 0$. Substituting into Eq. (10.2-3) yields $\rho = 0$, which, in a plot of ρ vs. θ , is a straight line.

(b) Only the origin $(0, 0)$ would yield this result.

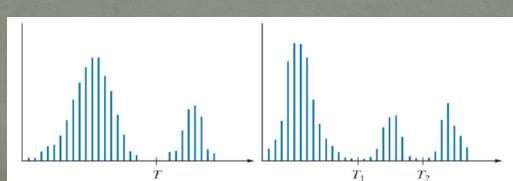
❖ Phân ngưỡng

Phân ngưỡng là kỹ thuật được sử dụng rộng rãi để tách các thành phần trong ảnh.

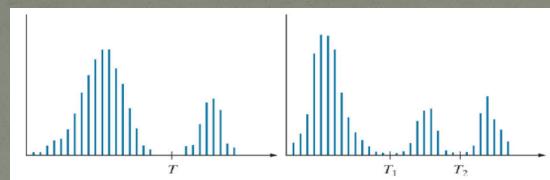
Giả sử ảnh gốc có lược đồ xám như trên hình, như vậy có thể nói ảnh sẽ bao gồm đối tượng có độ chói cao trên nền có độ chói thấp. Như vậy, để tách đối tượng ra khỏi nền, chúng ta chỉ cần so sánh độ chói $f(x,y)$ với mức ngưỡng T .

$g(x,y)$ sẽ là đối tượng cần tách.

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



❖ Phân ngưỡng

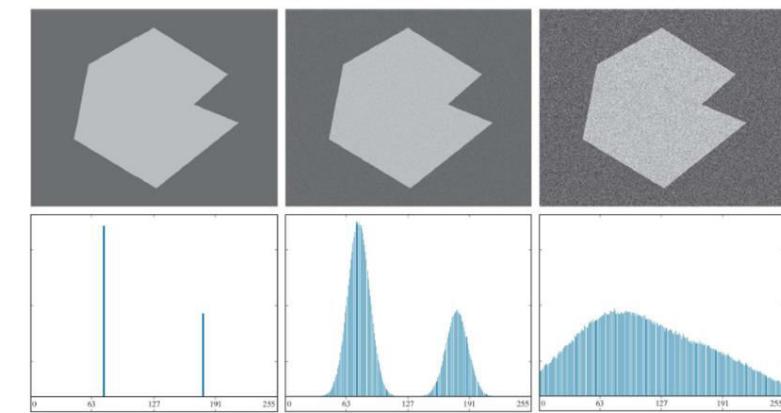


Trường hợp LĐX có cấu trúc phức tạp hơn, ảnh có thể được phân tách ra thành nhiều vùng theo các ngưỡng T khác nhau:

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

❖ Phân ngưỡng

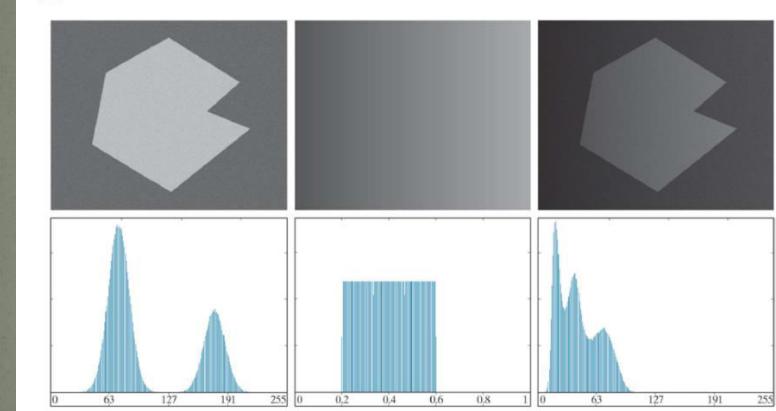
Tác động của nhiễu tới quá trình phân ngưỡng:



✓ Khử nhiễu trước khi phân ngưỡng

❖ Phân ngưỡng

Tác động của ánh sáng bên ngoài tới quá trình phân ngưỡng:



❖ Phân ngưỡng

•Tổng quan về phân vùng ảnh

Vùng ảnh (segment) là một chi tiết trong ảnh. Nó được hình thành bởi tập hợp các điểm ảnh có cùng hoặc gần **cùng một tính chất** nào đó như: mức xám, màu sắc, kết cấu (texture) v.v. Đường bao xung quanh vùng ảnh được gọi là **biên ảnh (boundary)**.

Phân vùng ảnh là quá trình phát hiện và tách ảnh thành các vùng hoặc các thành phần độc lập. Các thành phần độc lập có thể được xác định dựa trên cơ sở đường biên của chúng, hoặc dựa trên các tiêu chuẩn phân vùng liên thông có cùng độ chói, màu sắc hoặc có cùng các tham số khác của ảnh. Ngoài ra còn có các kỹ thuật phân vùng khác như phân vùng dựa vào **biên độ**, phân vùng theo **kết cấu (texture segmentation)**, phân vùng dựa trên tốc độ di chuyển của các chi tiết trong ảnh v.v.

➤ Phân vùng dựa trên xử lý ngưỡng (biên độ)

✓ Phương pháp phân vùng theo ngưỡng biên độ

Quá trình **xử lý ngưỡng** là quá trình so sánh ảnh với hàm T :

$$T = T[x, y, p(x, y), f(x, y)]$$

$f(x, y)$ - giá trị mức xám tại điểm (x, y) , $p(x, y)$ - hàm biểu diễn tính chất của vùng ảnh xung quanh điểm (x, y) .

Kết quả so sánh được tổng hợp trên ảnh $g(x, y)$ theo quy luật sau:

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

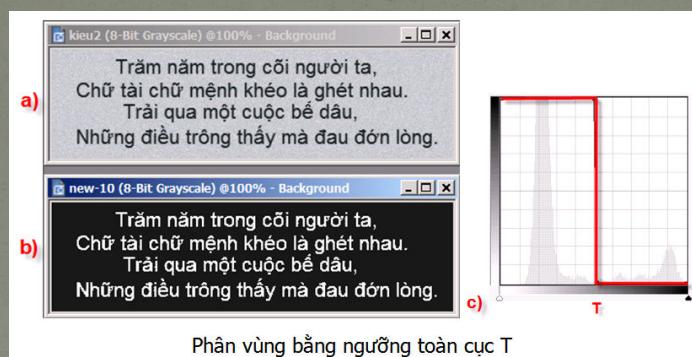
Chúng ta gọi các điểm ảnh trong $g(x, y)$ có giá trị bằng 1 là các điểm ảnh thuộc **vật thể** (chi tiết ảnh), các điểm có giá trị bằng 0 sẽ thuộc vùng **ảnh nền**.

Nếu T chỉ phụ thuộc vào $f(x, y)$ và không thay đổi trong toàn bộ quá trình xử lý ảnh thì T được gọi là **ngưỡng toàn cục**.

Nếu T phụ thuộc vào toạ độ không gian (x, y) ta gọi T là **ngưỡng cục bộ**.

Nếu ngưỡng cục bộ T phụ thuộc vào $p(x, y)$, ta gọi mức T là **ngưỡng thích nghi**.

✓ Phân vùng với ngưỡng toàn cục



Phân vùng bằng ngưỡng toàn cục T

Chọn ngưỡng như trên dựa trên cơ sở phân tích histogram của ảnh bằng mắt người quan sát.

Xác định ngưỡng T một cách tự động, chúng ta thực hiện các bước sau:

- 1- Lựa chọn mức ngưỡng ban đầu T.
- 2- Dùng T để phân vùng ảnh. Kết quả nhận được là hai nhóm điểm ảnh: nhóm G1 có giá trị độ chói lớn hơn T, G2 – có độ chói nhỏ hơn T.
- 3- Xác định giá trị độ chói trung bình m_1 và m_2 của hai nhóm điểm ảnh trên.
- 4- Xác định giá trị ngưỡng T mới bằng:

$$T = \frac{1}{2}(m_1 + m_2).$$

- 5- Thực hiện các bước 2-4 đến khi sai số giữa hai giá trị T kế tiếp không vượt quá giá trị T_0 cho trước.

Nếu vùng ảnh nền và vùng ảnh chứa đối tượng có độ chói cao có diện tích gần bằng nhau thì T ban đầu có thể chọn bằng độ chói trung bình của ảnh ngược lại thì $T = (i_{\max} + i_{\min})/2$.

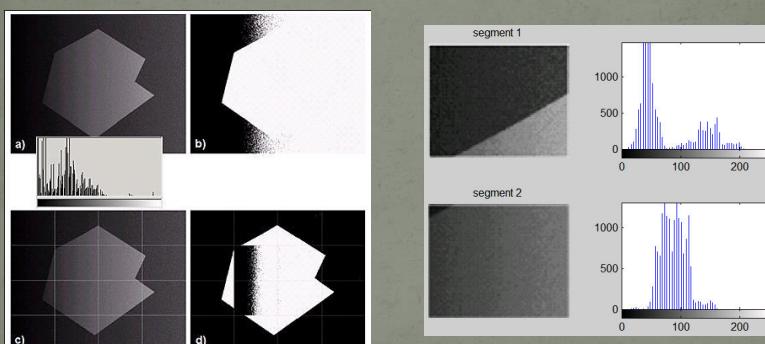
T_0 – dùng để dừng quá trình tìm kiếm ngưỡng



✓ Phân vùng với ngưỡng thích nghi

Quá trình xử lý với **mức ngưỡng thích nghi** theo từng vùng ảnh thực hiện như sau:

Ảnh gốc ra thành nhiều vùng nhỏ (segment), trong mỗi vùng chúng ta sẽ tìm giá trị ngưỡng cục bộ theo phương pháp đã mô tả ở trên. Giá trị T ban đầu được chọn bằng giá trị trung bình giữa mức chói cao nhất và mức chói thấp nhất trong vùng.



- a) Ảnh gốc.
- b) Phân vùng với ngưỡng toàn cục.
- c) Ảnh được chia vùng
- d) Phân vùng với ngưỡng thích nghi. (cục bộ)

✓ Tách ảnh theo các mức màu

Kỹ thuật này cho phép **tách những chi tiết có màu nhất định** ra khỏi ảnh gốc. Những thành phần màu khác trong ảnh kết quả sẽ được **biến đổi thành màu nền** (làm nổi những chi tiết màu cần tách).

Nếu **các màu được quan tâm nằm trong hình khôi** con có cạnh là D_0 , tâm hình khôi nằm tại điểm (a_1, a_2, a_3) trong không gian màu R, G, B, khi đó kỹ thuật tách ảnh theo các mức màu được mô tả theo biểu thức:

$$s_i = \begin{cases} 0,5 & |r_j - a_j| > \frac{D_0}{2} \\ r_j & |r_j - a_j| \leq \frac{D_0}{2} \end{cases}$$

với các $i=1, 2, 3$; $j=1, 2, 3$;

Để **tách vùng màu nằm trong hình cầu** ta dùng biến đổi:

$$s_i = \begin{cases} 0,5 & \sum_{j=1}^n (r_j - a_j)^2 > R_0^2 \\ r_j & \sum_{j=1}^n (r_j - a_j)^2 \leq R_0^2 \end{cases}$$

$i = 1, 2, \dots, n$, $j = 1, 2, \dots, n$ ($n=3$)

