



ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



Thị giác máy tính

Bài 4: Phát hiện biên

Nội dung

- Vai trò của biên và cách tiếp cận chung
- Phát hiện biên
 - Image gradient (độ dốc ảnh): Sobel, Prewitt
 - Canny detector
 - Laplacian
- Phát hiện đường thẳng
 - Biến đổi Hough (Hough transform)
 - Ransac

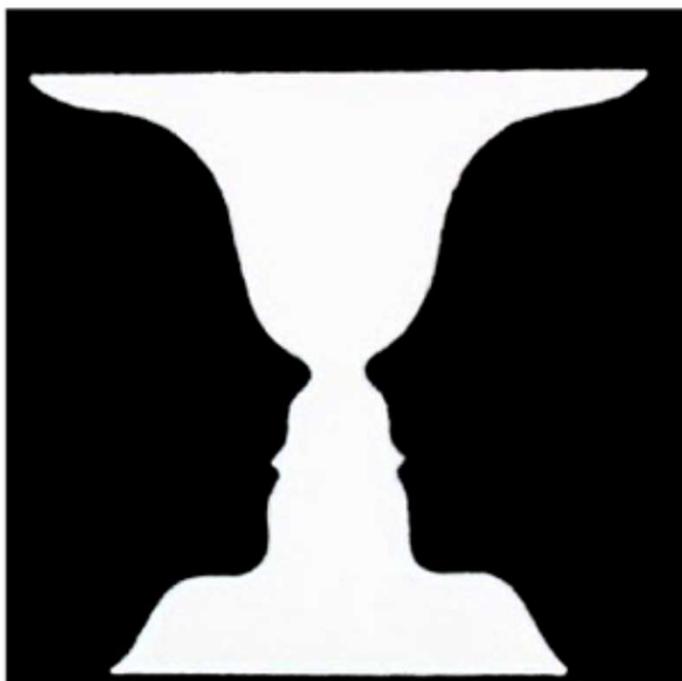
Biên là gì (edge/contour) ?

- Nơi có sự thay đổi cường độ sáng trong ảnh
- Thường xảy ra ở ranh giới giữa các vùng khác nhau trong ảnh
- Biên có thể tạo ra bởi nhiều lý do:



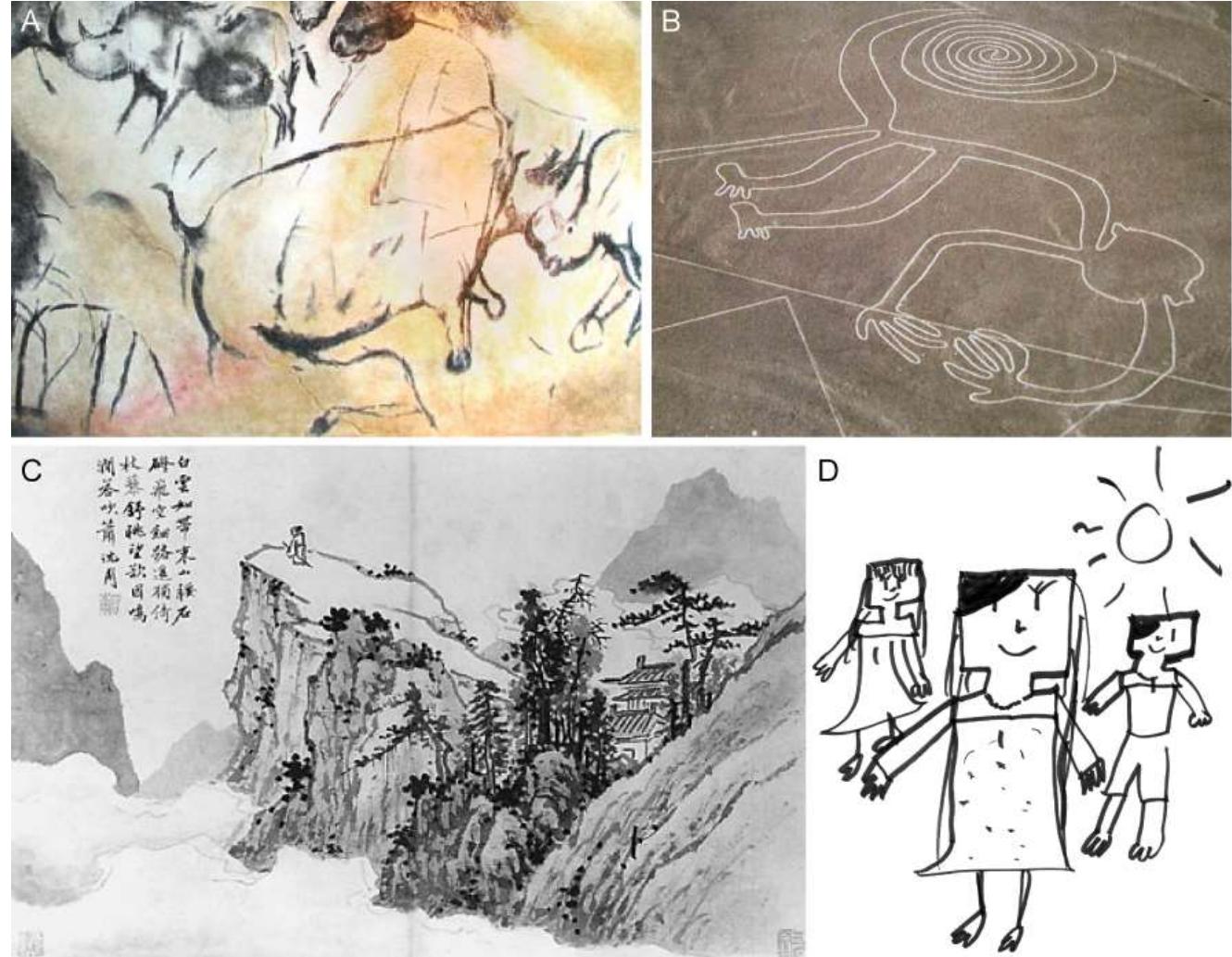
Vai trò của biên

- Bạn thấy gì?



Vai trò của biển

- (A) Các bức tranh trong hang động tại Chauvet, Pháp, khoảng 30,000 TCN;
- (B) Ảnh chụp trên không hình một con khỉ từ các đường geoglyph tại Nazca Lines, Peru, khoảng 700 – 200 TCN;
- (C) Shen Zhou (1427-1509 A.D.): Tranh vẽ đỉnh núi, trên giấy, Trung Quốc
- (D) Bức tranh vẽ bởi I. Lleras, 7 tuổi (2010).



<https://www.youtube.com/watch?v=IOHayh06LJ4>

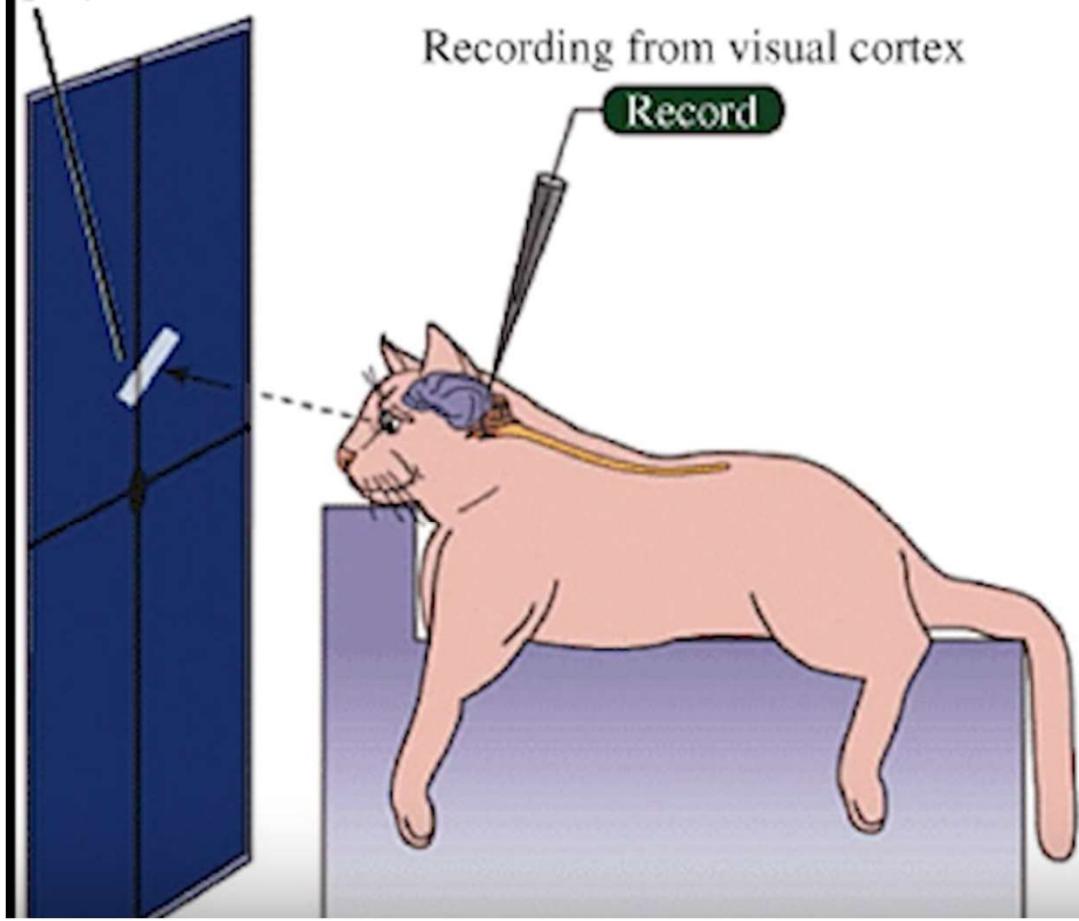
<https://www.youtube.com/watch?v=Cw5PKV9Rj3o>

A Experimental setup

Light bar stimulus
projected on screen

Recording from visual cortex

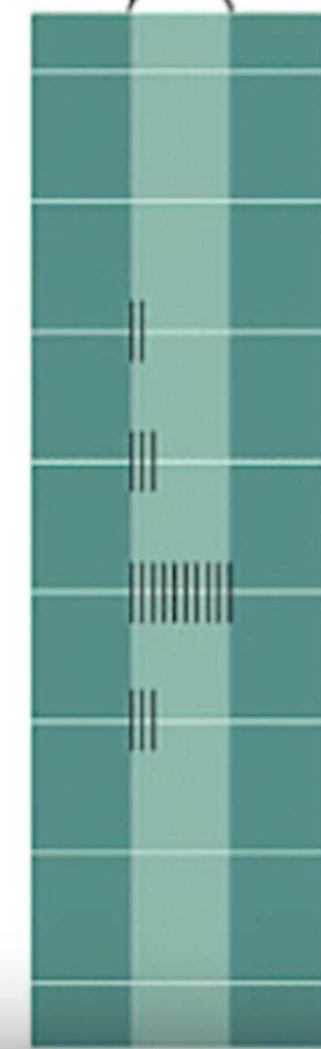
Record



B Stimulus orientation

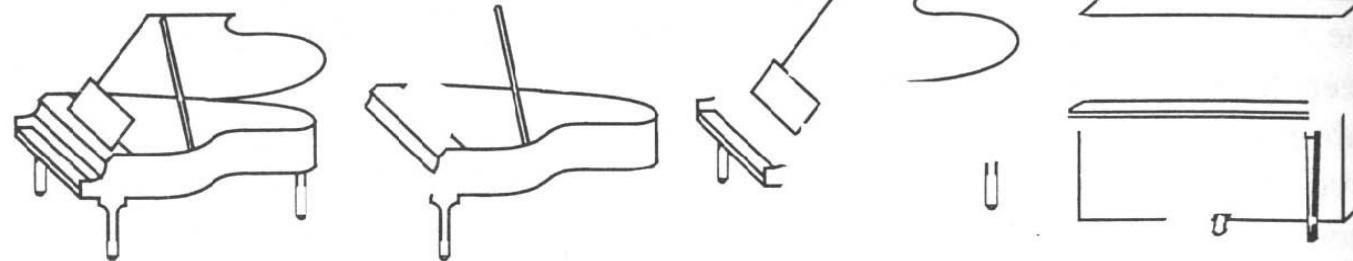
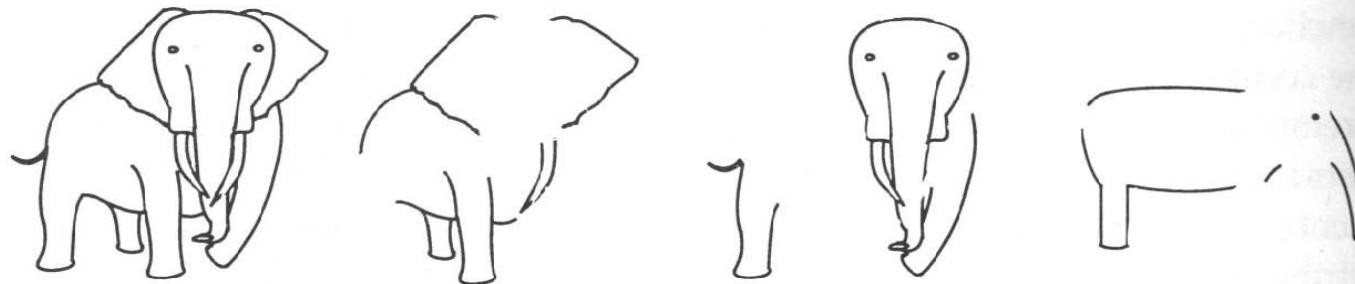


Stimulus presented



Hubel & Wiesel, 1960s

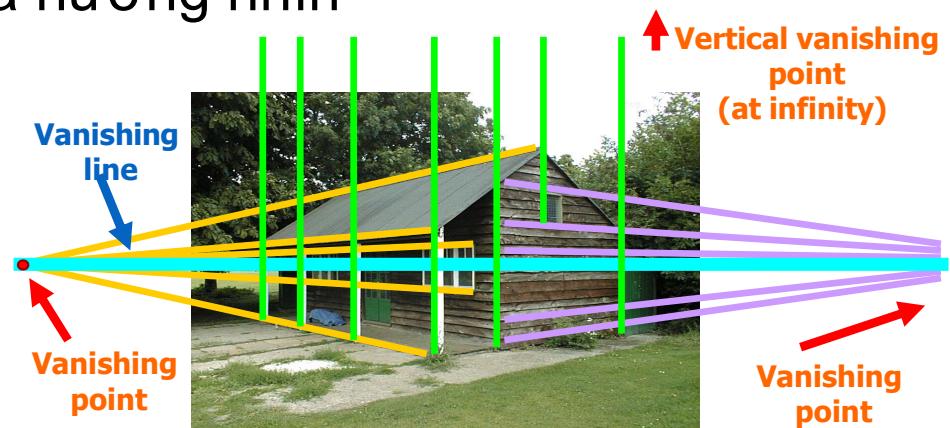
152 Biederman



Chúng ta có thể nhận dạng các vật thể này?

Phát hiện biên

- Mục tiêu: xác định nơi có sự thay đổi cường độ sáng trên ảnh
 - Về mặt trực quan, thông tin ngữ nghĩa hoặc hình dáng trong ảnh được thể hiện thông qua biên
 - Biên thể hiện thông tin cấp cao hơn so với điểm
- Lý do?
 - Trích chọn thông tin, nhận dạng đối tượng
 - Xác định thông tin hình học và hướng nhìn



Source: J. Hayes

Phát hiện biên ?

- Intensity profile:
 - tập giá trị điểm ảnh được lấy đều dọc theo một đường nào đó trên ảnh

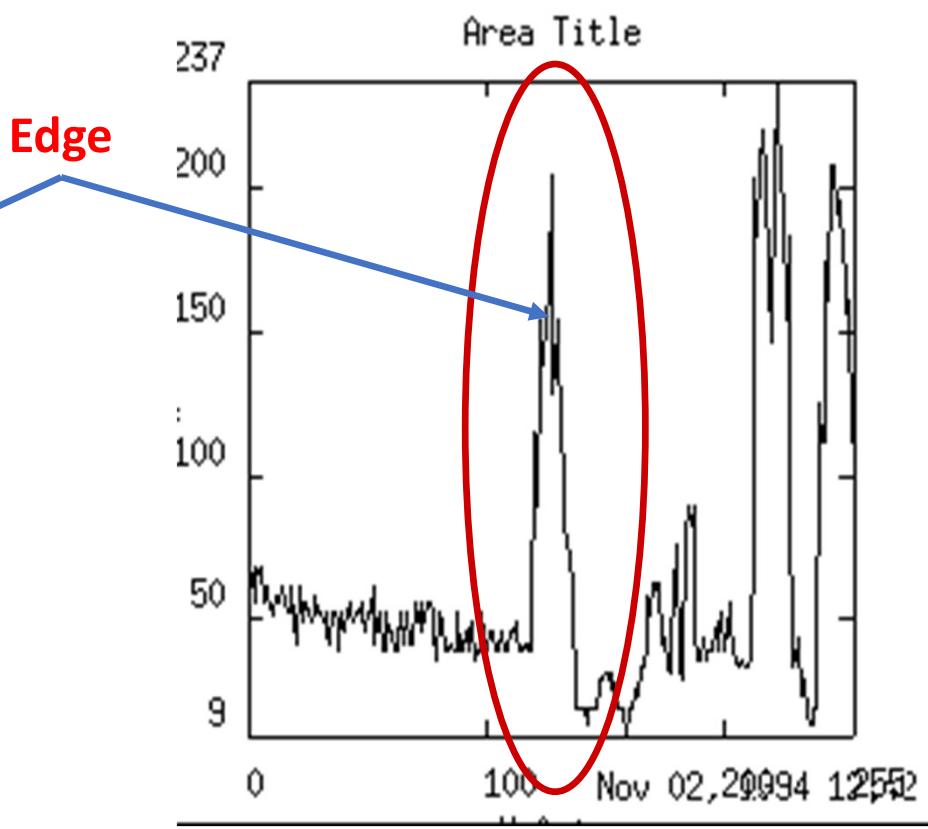
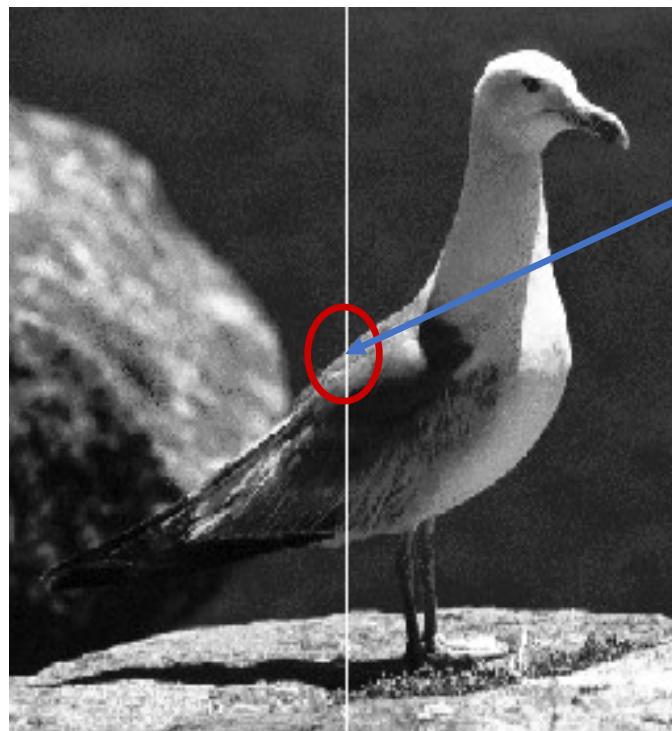
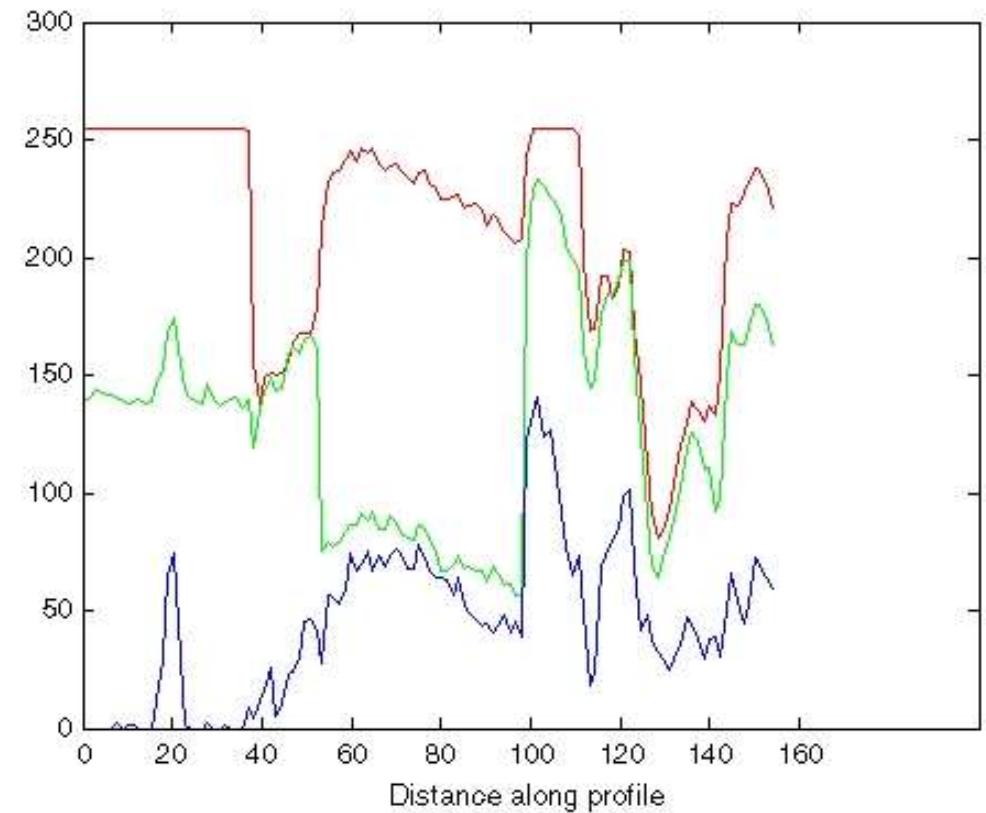
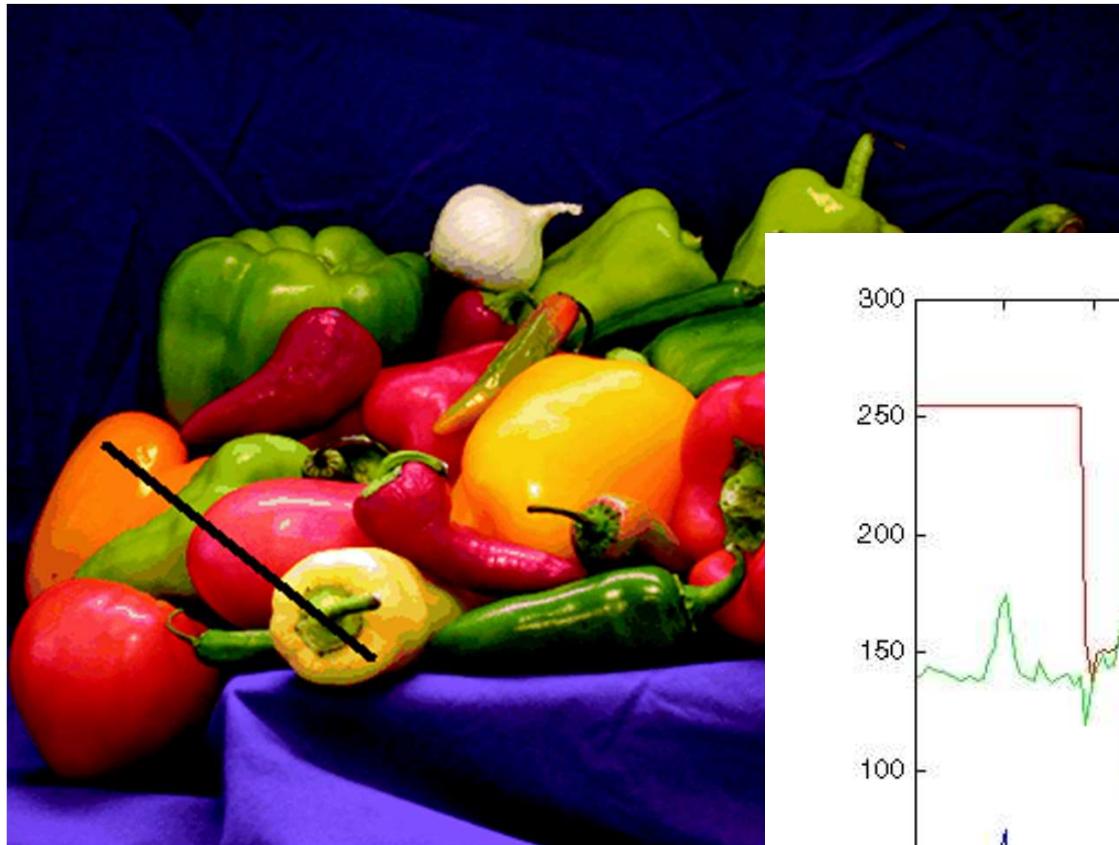


Image profile



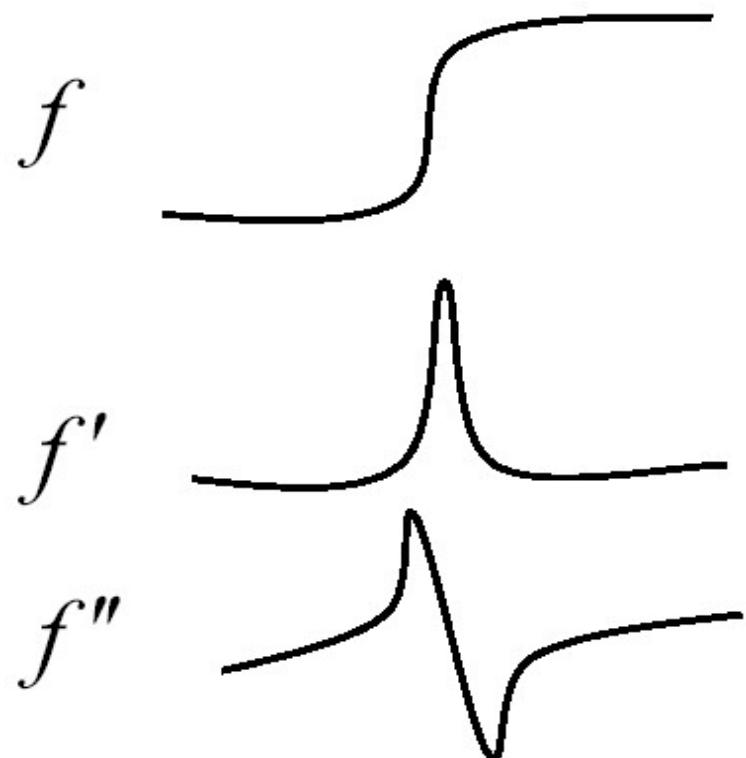
<https://www.mathworks.com/help/images/intensity-profile-of-images.html>

Phát hiện biên ?

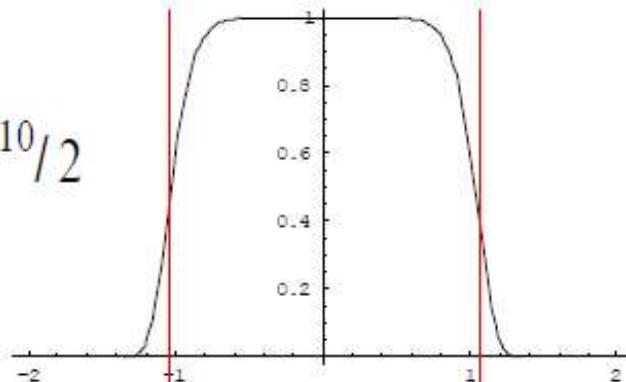
- Biên là vị trí có sự thay đổi nhanh về cường độ

- Đạt cực trị trên đạo hàm bậc 1

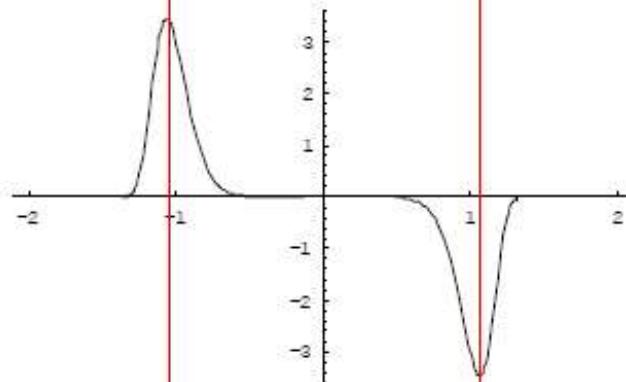
- Đi qua không (zero-crossing) ở đạo hàm bậc 2



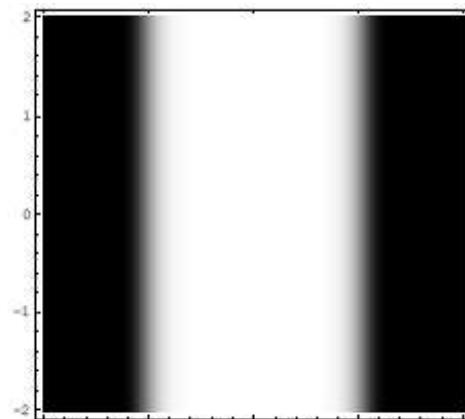
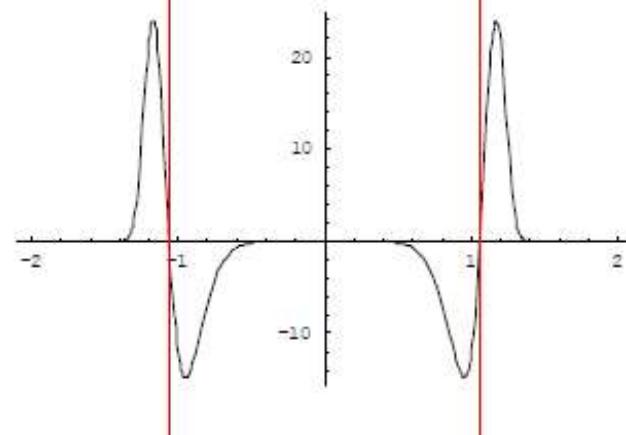
$$f(x, y) = e^{-x^{10}/2}$$



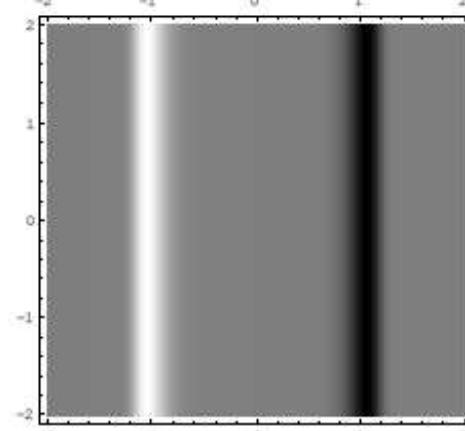
$$\frac{\partial f}{\partial x}$$



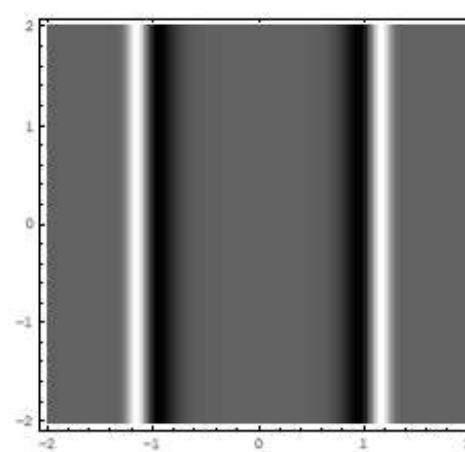
$$\frac{\partial^2 f}{\partial x^2}$$



Image



The first derivative



The second derivative

Source : Caroline Rougier. Traitement d'images (IFT2730). Univ. de Montréal.

Đạo hàm bậc 1

- Đạo hàm 1 chiều :

$$\frac{df}{dx} = \lim_{\Delta x \rightarrow 0} \frac{f(x) - f(x - \Delta x)}{\Delta x} = f'(x) = f_x$$

- Đạo hàm rời rạc 1 chiều:

$$\frac{df}{dx} = \frac{f(x) - f(x - 1)}{1} = f'(x)$$

$$\frac{df}{dx} = f(x) - f(x - 1) = f'(x)$$

Một số cách tính đạo hàm bậc 1

Lui

$$\frac{df}{dx} = f(x) - f(x-1) = f'(x)$$

Tiến

$$\frac{df}{dx} = f(x) - f(x+1) = f'(x)$$

Trung tâm

$$\frac{df}{dx} = f(x+1) - f(x-1) = f'(x)$$

Bộ lọc tính đạo hàm bậc 1

- Lọc quay lui (backward):

$$\frac{df}{dx} = f(x) - f(x-1) = f'(x) \quad [0 \quad 1 \quad -1]$$

- Lọc tiến (Forward):

$$\frac{df}{dx} = f(x) - f(x+1) = f'(x) \quad [-1 \quad 1 \quad 0]$$

- Lọc trung tâm (Central):

$$\frac{df}{dx} = f(x+1) - f(x-1) = f'(x) \quad [1 \quad 0 \quad -1]$$

Ví dụ:

- Backward filter:

$$[0 \quad 1 \quad -1]$$

$$f(x) = 10 \quad 15 \quad 10 \quad 10 \quad 25 \quad 20 \quad 20 \quad 20$$

$$f'(x) = 0 \quad 5 \quad -5 \quad 0 \quad 15 \quad -5 \quad 0 \quad 0$$

$$f(x) : \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 50 \quad 50 \quad 50 \quad 50 \quad 50$$

$$f'(x) : \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 50 \quad 0 \quad 0 \quad 0 \quad 0$$

Đạo hàm rời rạc trên ảnh

Given function

$$f(x, y)$$

Gradient vector

$$\nabla f(x, y) = \begin{bmatrix} \frac{\partial f(x, y)}{\partial x} \\ \frac{\partial f(x, y)}{\partial y} \end{bmatrix} = \begin{bmatrix} f_x \\ f_y \end{bmatrix}$$

Gradient magnitude

$$|\nabla f(x, y)| = \sqrt{f_x^2 + f_y^2}$$

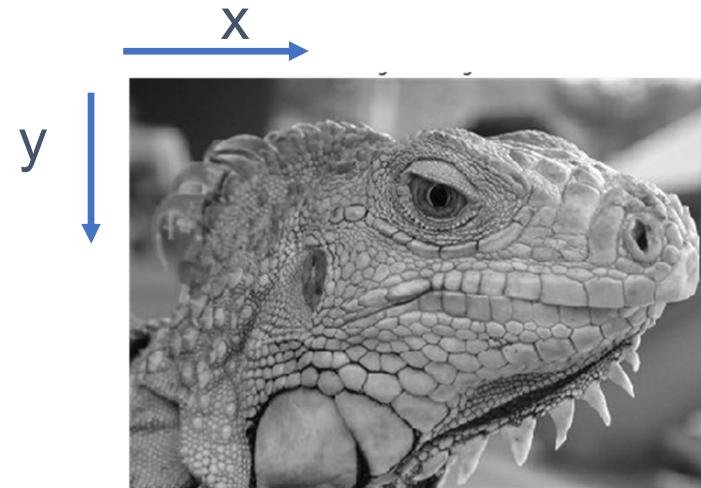
Gradient direction

$$\theta = \tan^{-1} \frac{f_x}{f_y}$$

Đạo hàm bậc 1 theo x và y

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

0
1
-1



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

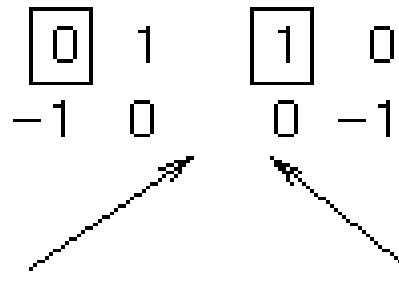
-1
1
0

1
0
-1

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

Đạo hàm bậc 1 theo x và y

- Lọc Robert (bộ lọc xấp xỉ đầu tiên dựa vào đạo hàm - 1965)

$$\begin{matrix} 0 & 1 \\ -1 & 0 \end{matrix} \quad \begin{matrix} 1 & 0 \\ 0 & -1 \end{matrix}$$


- Lọc Prewitt

$$1/3 \times \begin{matrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & -0 & -1 \end{matrix}$$

$$1/3 \times \begin{matrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{matrix}$$

- Lọc Sobel

$$1/4 \times \begin{matrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & -0 & -1 \end{matrix}$$

$$1/4 \times \begin{matrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{matrix}$$

Image I (9 x 8)

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	0	0
0	0	0	100	100	100	100	100	100
0	0	0	100	100	100	100	100	100
0	0	0	100	100	100	100	100	100
0	0	0	100	100	100	100	100	100
0	0	0	100	100	100	100	100	100
0	0	0	100	100	100	100	100	100

M_x : Lọc đạo hàm phương x

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

1
0
-1

M_y : Lọc đạo hàm phương y

$I_x = I * M_x$ = đạo hàm 1 phương x của I

		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	100	100	0	0	0	0
		0	100	100	0	0	0	0
		0	100	100	0	0	0	0
		0	100	100	0	0	0	0
		0	100	100	0	0	0	0

Image I (9 x 8)

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	0	0
0	0	0	100	100	100	100	100	100
0	0	0	100	100	100	100	100	100
0	0	0	100	100	100	100	100	100
0	0	0	100	100	100	100	100	100
0	0	0	100	100	100	100	100	100
0	0	0	100	100	100	100	100	100

M_x : Lọc đạo hàm phương x

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

1
0
-1

M_y : Lọc đạo hàm phương y

$$Ix = I * Mx = \text{đạo hàm 1 phương y của } I$$

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	100	100	100	100	100	100
0	0	100	100	100	100	100	100
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Lọc 3x3 gradient (độ dốc)

$$\frac{1}{3} \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

$$\frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

Derivative in x direction

Derivative in y direction

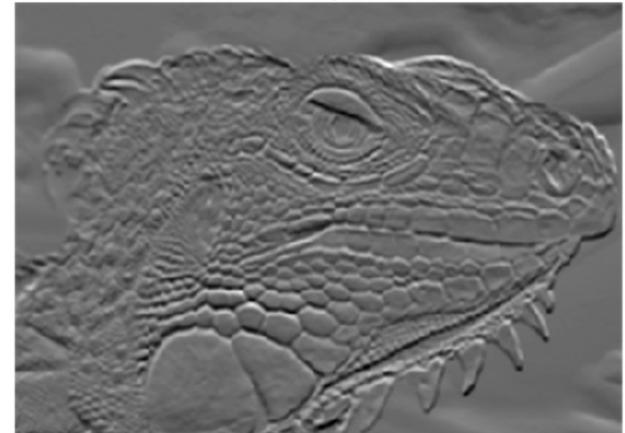
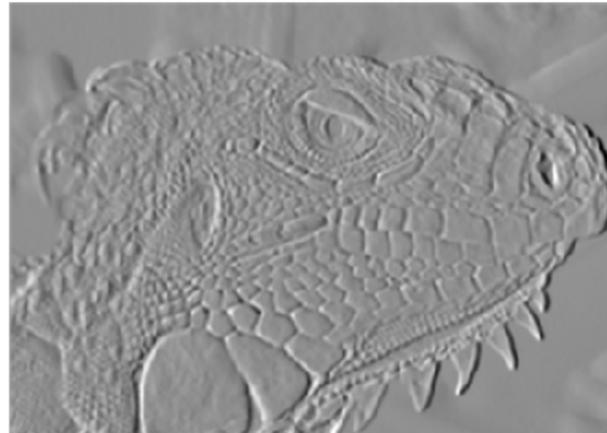
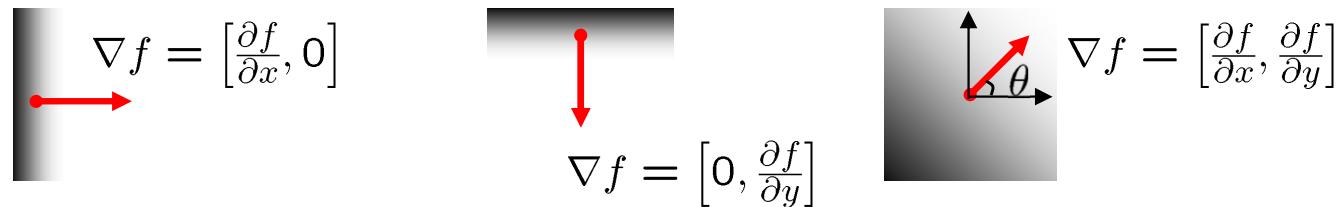


Image gradient (độ dốc ảnh)

- Độ dốc của 1 ảnh:

$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right]$$



- Véc-tor gradient chỉ về hướng tang lớn nhất về mức xám
- Hướng gradient được tính bằng $\theta = \tan^{-1} \left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x} \right)$
 - Có mối quan hệ thế nào tới hướng của biên?
- Độ lớn của cạnh được tính bằng độ lớn dốc (gradient magnitude)

$$\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \approx |\partial f/\partial x| + |\partial f/\partial y|$$

Source: Steve Seitz

Ví dụ

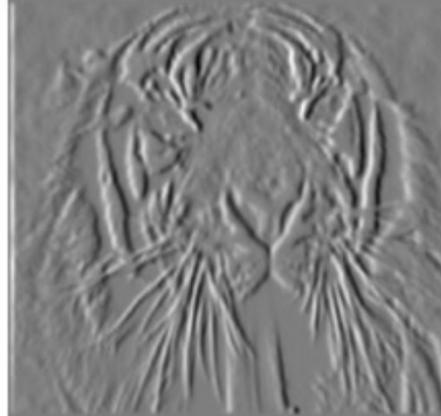
Ảnh gốc



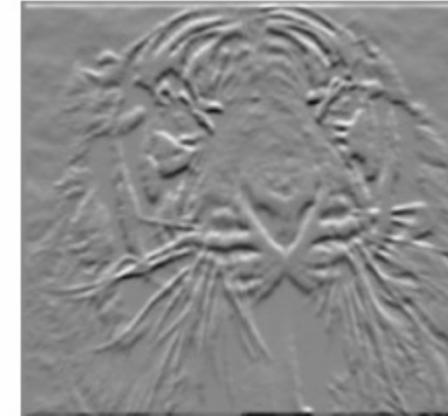
Cường độ
dốc



Theo
hướng x

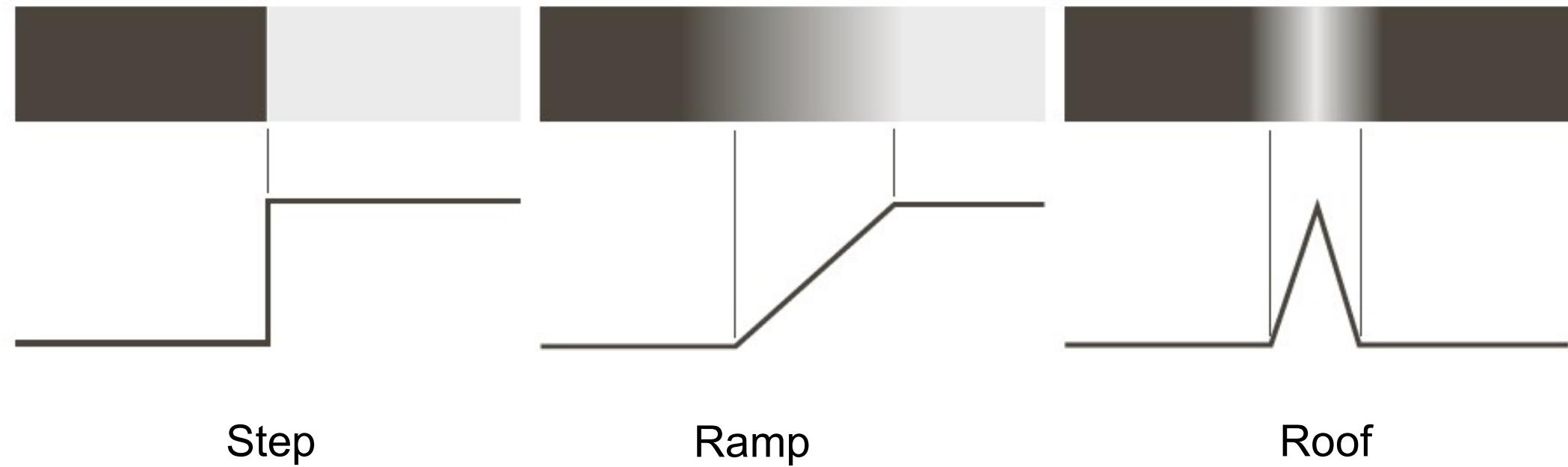


Theo
hướng y



- Đâu là độ dốc theo phương x và đâu là theo phương y?

Phát hiện biên - Các kiểu biên



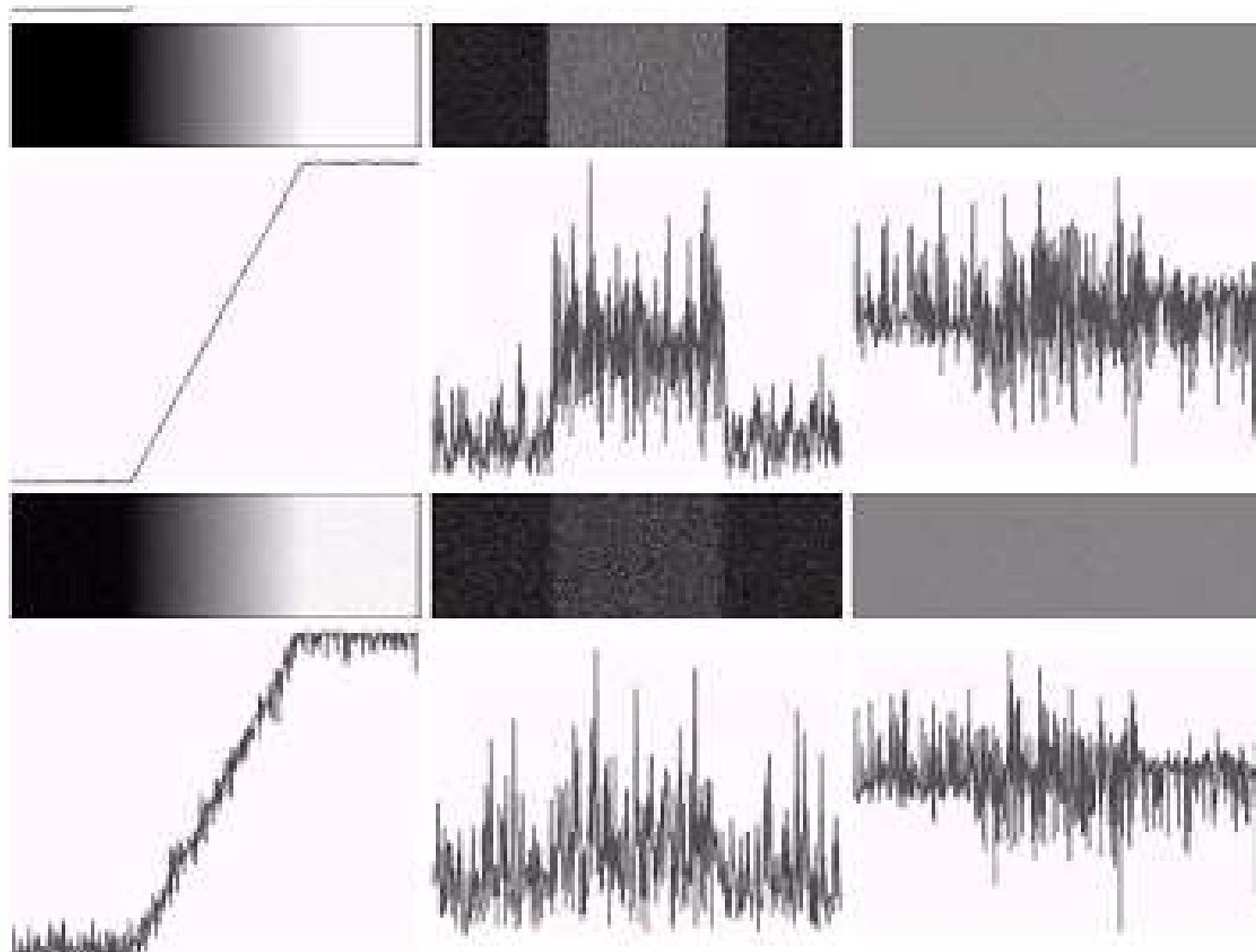
Step

Ramp

Roof

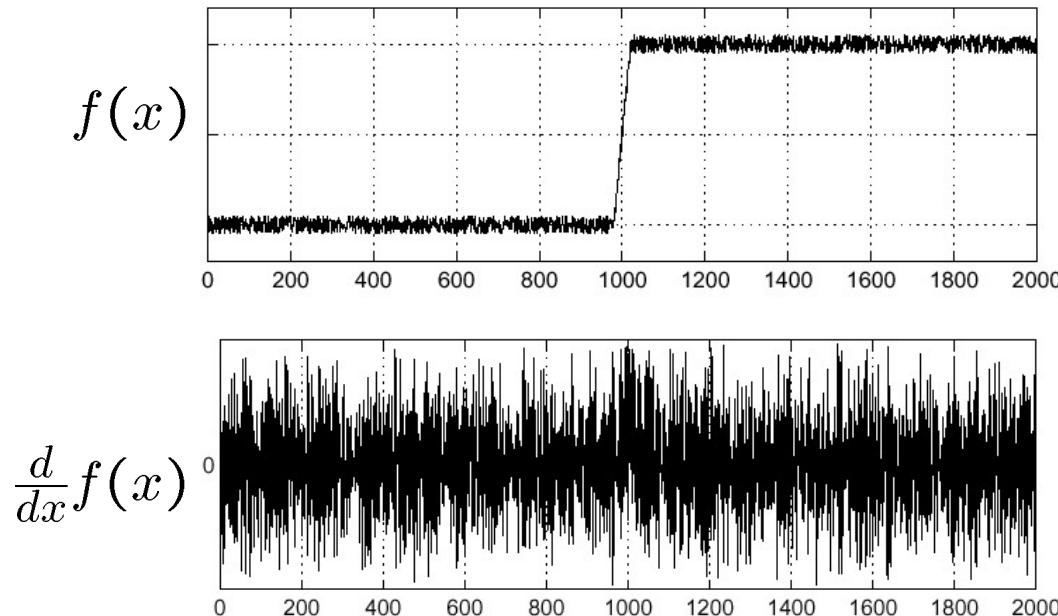
Source : Gonzalez and Woods. Digital Image Processing 3ed. Prentice-Hall, 2008.

Nhiễu trên biên



Source : Gonzalez and Woods. Digital Image Processing 3ed. Prentice-Hall, 2008.

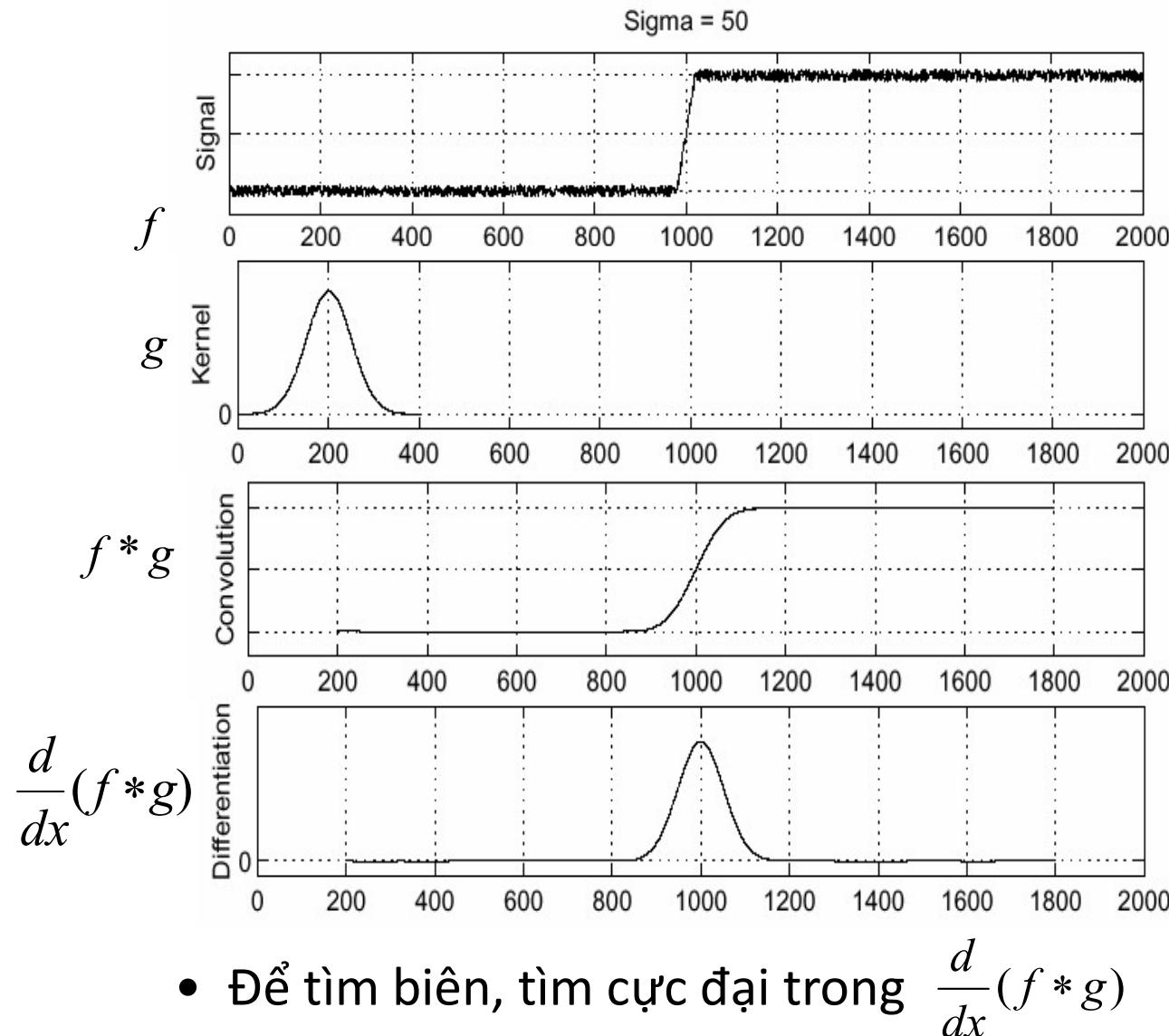
Ảnh hưởng của nhiễu



Biên ở đâu?

→ Giải pháp: làm trơn ảnh trước

Giải pháp: làm trơn + tính đạo hàm



- Để tìm biên, tìm cực đại trong $\frac{d}{dx}(f * g)$

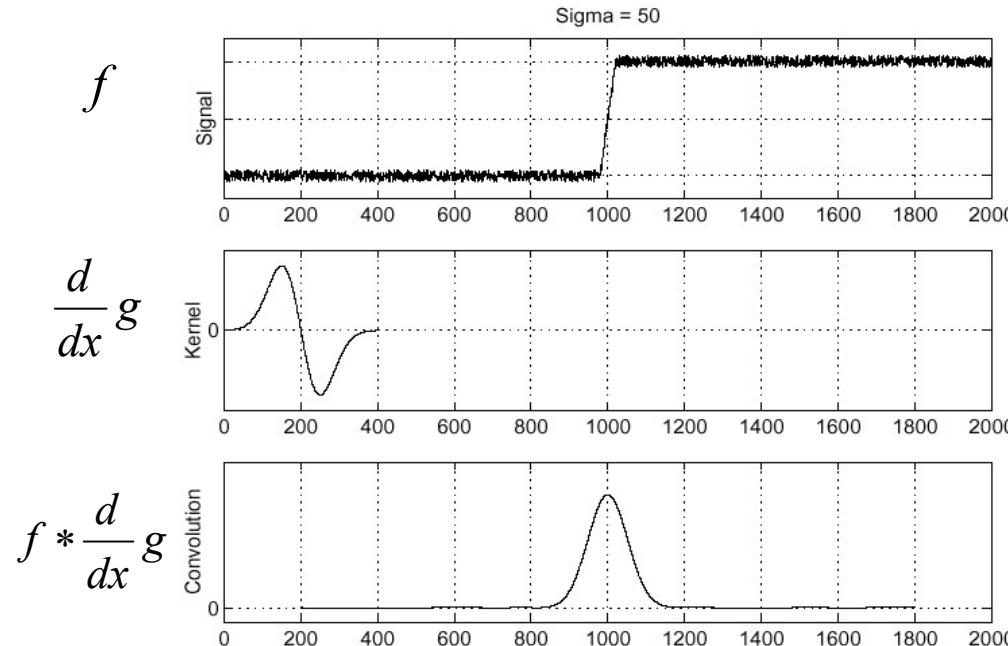
Source: S. Seitz

Lý thuyết đạo hàm của nhân chập

- Tính chất hữu ích:

$$\frac{d}{dx}(f * g) = \frac{df}{dx} * g = f * \frac{dg}{dx}$$

- Điều này giúp giảm phép toán:



Source: S. Seitz

Toán tử Sobel

- Làm trơn Gaussian + đạo hàm

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} [1 \quad 0 \quad -1]$$

Gaussian smoothing

differentiation

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} [1 \quad 2 \quad 1]$$

→ Ít nhạy với nhiễu hơn

Toán tử Prewitt

- Lọc trung bình + đạo hàm

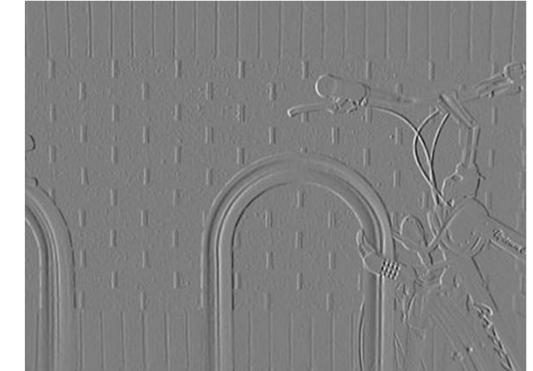
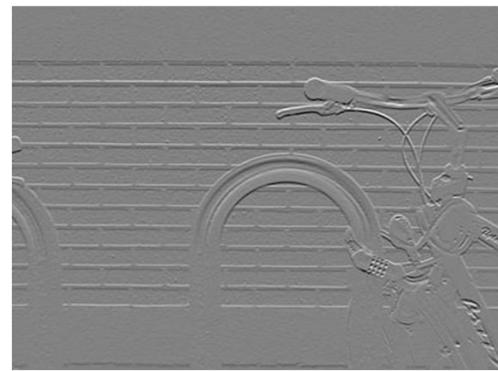
$$Gx = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} [1 \quad 0 \quad -1]$$

$$Gy = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} [1 \quad 1 \quad 1]$$

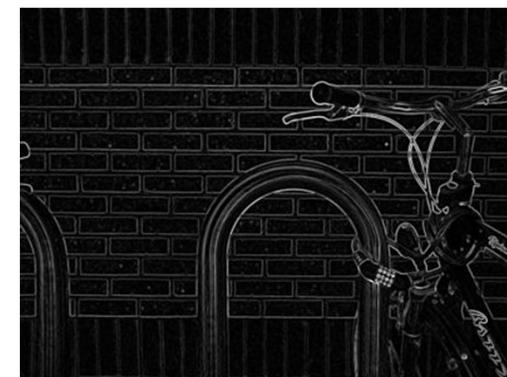
→ Ít nhạy với nhiễu hơn

Phát hiện cạnh đơn giản sử dụng đạo hàm bậc 1

- Nhận chập ảnh với 2 mặt nạ để xấp xỉ đạo hàm bậc 1 theo x và y



- Tính độ lớn của gradient



- Lấy ngưỡng: cạnh là điểm có độ lớn gradient > T

Phát hiện cạnh đơn giản sử dụng đạo hàm bậc 1

Ảnh gốc



Ngưỡng
 $T = 25$



Độ lớn dốc
Toán tử Sobel



Ngưỡng
 $T = 60$



Source : Caroline Rougier. Traitement d'images (IFT2730). Univ. de Montréal.

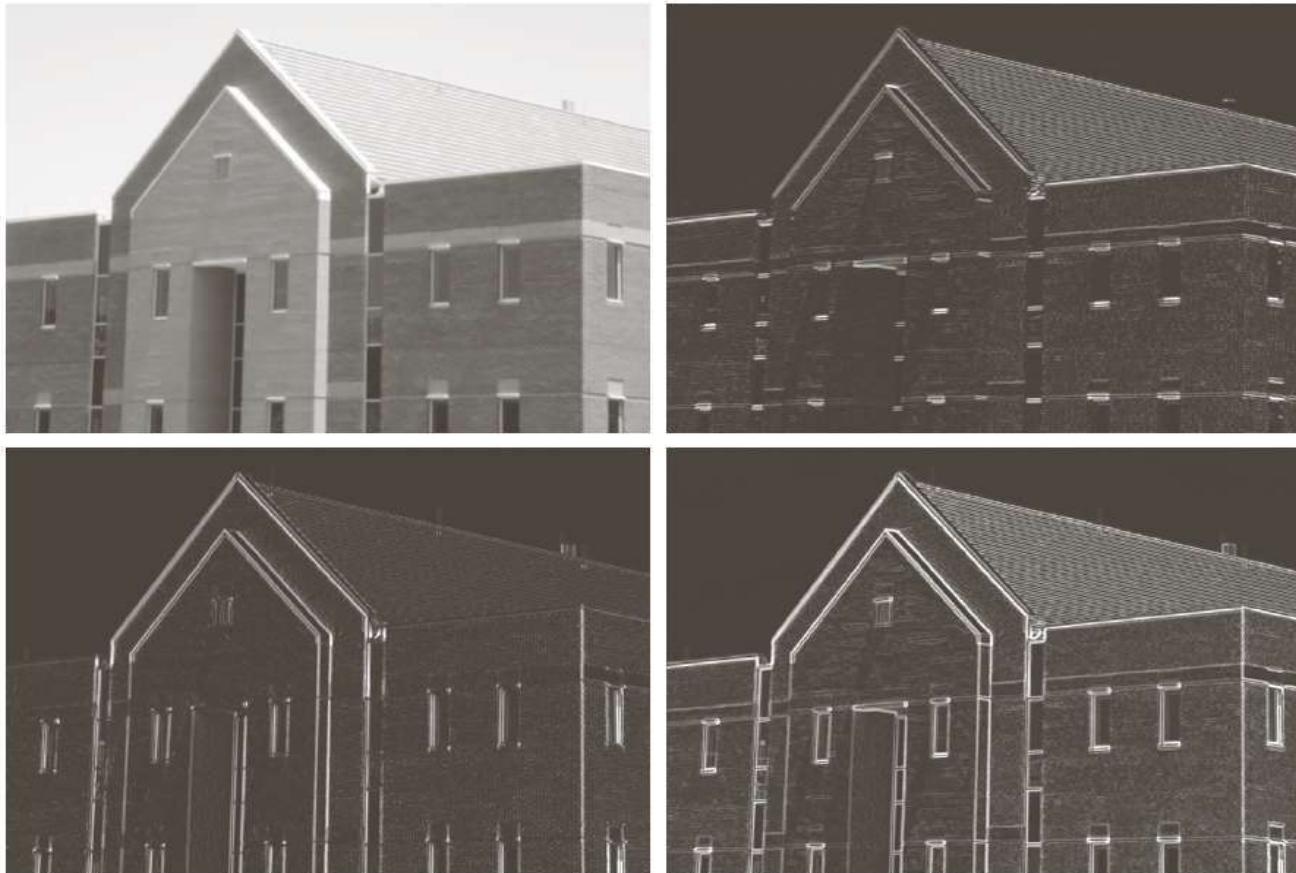
Không làm trơn với Gauss trước khi tính đạo hàm



a b
c d

FIGURE 10.16
(a) Original image of size 834×1114 pixels, with intensity values scaled to the range $[0, 1]$.
(b) $|g_x|$, the component of the gradient in the x -direction, obtained using the Sobel mask in Fig. 10.14(f) to filter the image.
(c) $|g_y|$, obtained using the mask in Fig. 10.14(g).
(d) The gradient image, $|g_x| + |g_y|$.

Làm trơn với Gauss trước khi tính đạo hàm

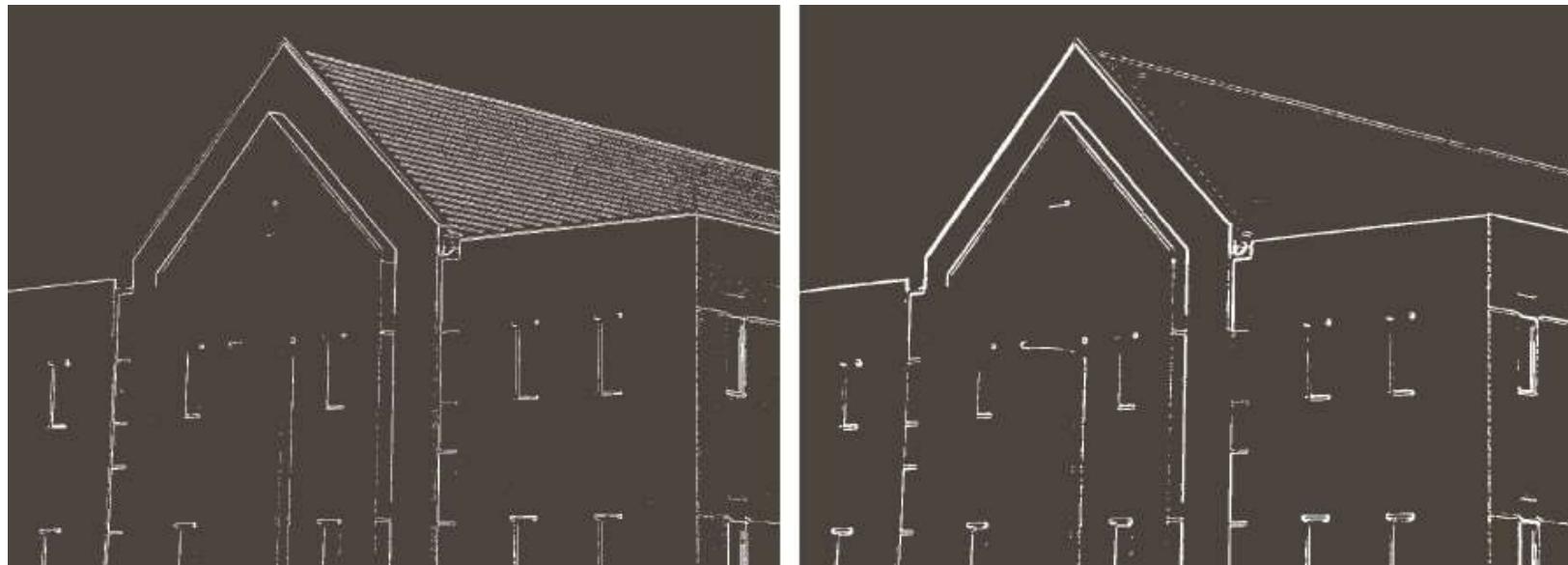


a b
c d

FIGURE 10.18
Same sequence as in Fig. 10.16, but with the original image smoothed using a 5×5 averaging filter prior to edge detection.

Không làm trơn với Gauss
trước tính đạo hàm

Làm trơn với Gauss
trước tính đạo hàm

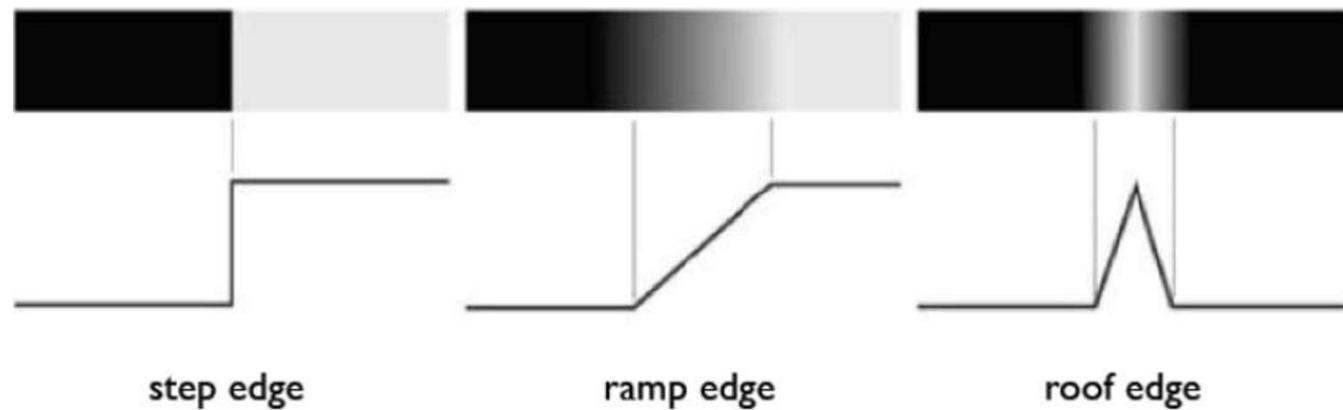


a | b

FIGURE 10.20 (a) Thresholded version of the image in Fig. 10.16(d), with the threshold selected as 33% of the highest value in the image; this threshold was just high enough to eliminate most of the brick edges in the gradient image. (b) Thresholded version of the image in Fig. 10.18(d), obtained using a threshold equal to 33% of the highest value in that image.

Vấn đề

- Vị trí không chính xác (biên dày)
- Giá trị ngưỡng ưu ái cạnh theo 1 vài hướng hơn là các hướng khác
 - Có thể thiếu các đường biên chéo hơn là biên ngang hoặc dọc → bỏ sót biên



Một số mặt nạ khác

- Để tránh ưu ái cạnh theo 1 vài hướng → sử dụng kỹ thuật la bàn:
 - Nhân chập ảnh với 8 mặt nạ theo 8 hướng (0, 45, 90, ...)
 - Cộng kết quả nhân chập lại

$$H_1 = \begin{pmatrix} 5 & 5 & -3 \\ 5 & 0 & -3 \\ -3 & -3 & -3 \end{pmatrix}$$

$$H_2 = \begin{pmatrix} 5 & 5 & 5 \\ -3 & 0 & -3 \\ -3 & -3 & -3 \end{pmatrix}$$

$$H_3 = \begin{pmatrix} -3 & 5 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & -3 \end{pmatrix}$$

$$H_4 = \begin{pmatrix} -3 & -3 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & 5 \end{pmatrix}$$

$$H_5 = \begin{pmatrix} -3 & -3 & -3 \\ -3 & 0 & 5 \\ -3 & 5 & 5 \end{pmatrix}$$

$$H_6 = \begin{pmatrix} -3 & -3 & -3 \\ -3 & 0 & -3 \\ 5 & 5 & 5 \end{pmatrix}$$

$$H_7 = \begin{pmatrix} -3 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & 5 & -3 \end{pmatrix}$$

$$H_8 = \begin{pmatrix} 5 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & -3 & -3 \end{pmatrix}$$

Một số mặt nạ khác

- Mặt nạ Prewitt, Sobel cho phát hiện biên chéo

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

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

Prewitt

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

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

Sobel

a	b
c	d

FIGURE 10.15
Prewitt and Sobel
masks for
detecting diagonal
edges.

Một số mặt nạ khác

- Mặt nạ Prewitt, Sobel cho phát hiện biên chéo



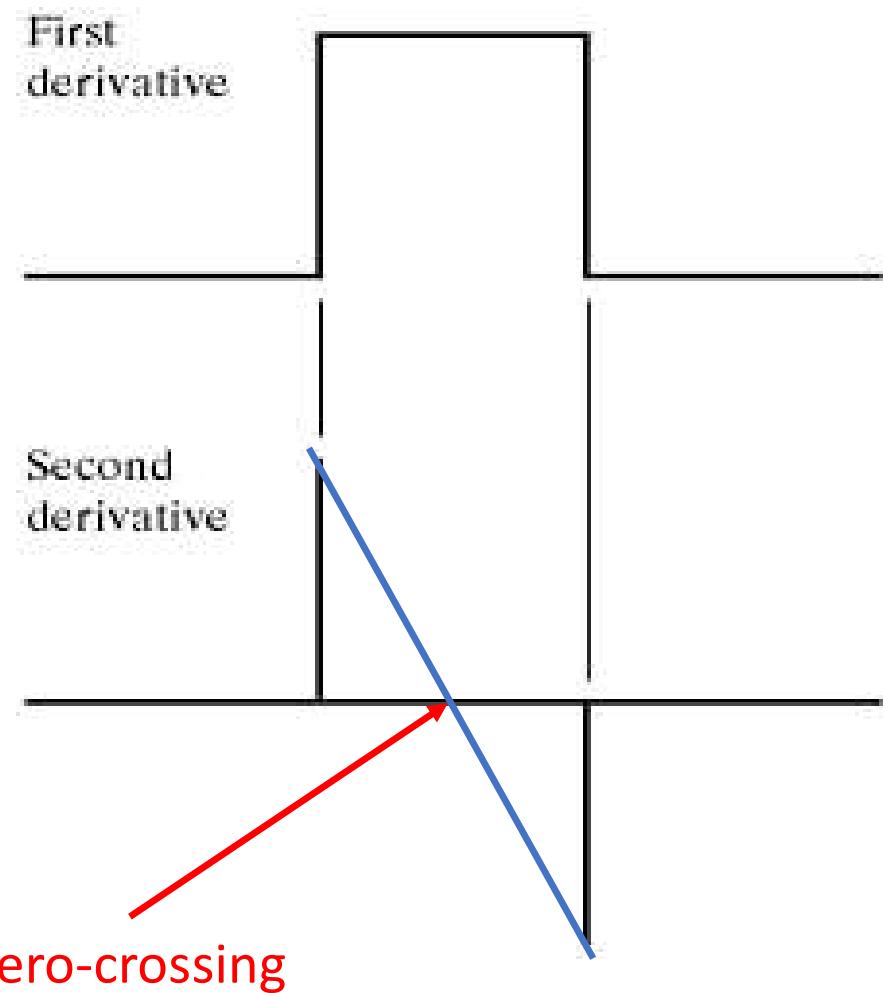
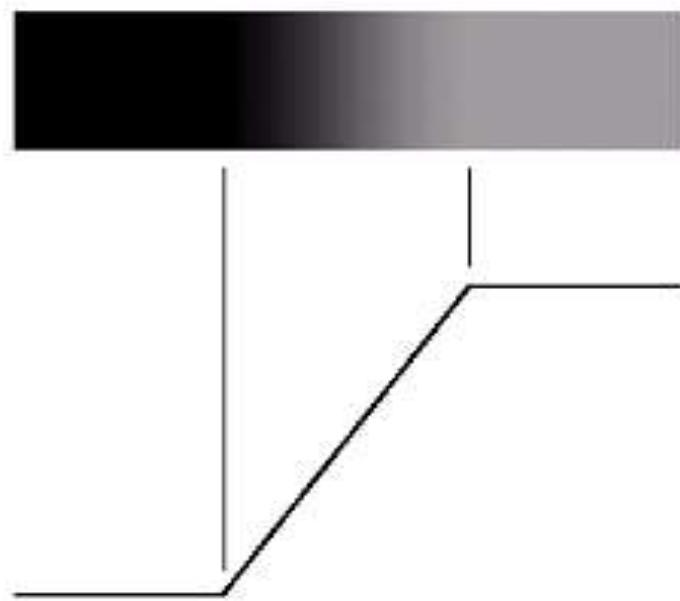
a b

FIGURE 10.19

Diagonal edge detection.

(a) Result of using the mask in Fig. 10.15(c).
(b) Result of using the mask in Fig. 10.15(d). The input image in both cases was Fig. 10.18(a).

Phát hiện biên với đạo hàm bậc 2



Phát hiện biên với đạo hàm bậc 2

- Sử dụng bộ lọc Laplacian:
 - Nhân chập ảnh với một trong 2 mặt nạ

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

or

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

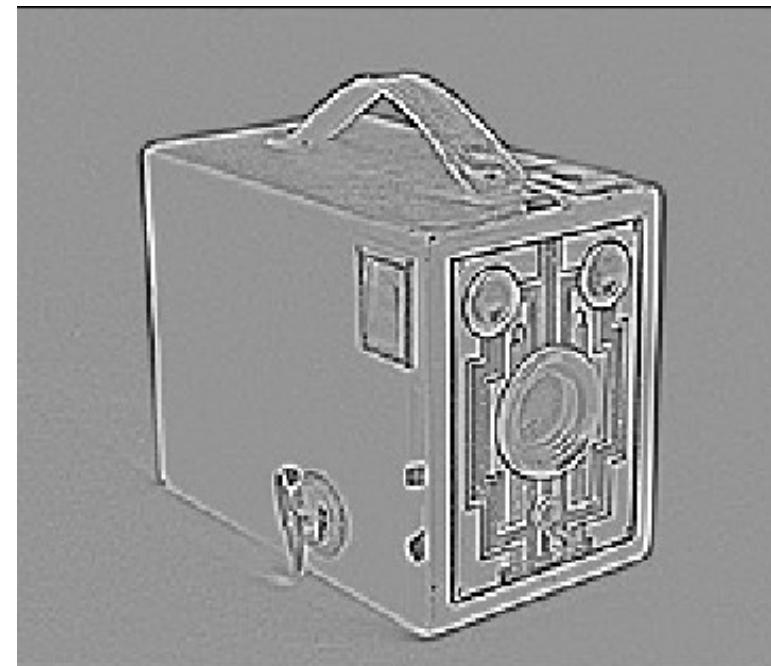
- Phát hiện biên:
 - Tính đạo hàm bậc 2
 - Tìm điểm qua 0 → cạnh

Phát hiện biên với đạo hàm bậc 2

Image



Laplacian



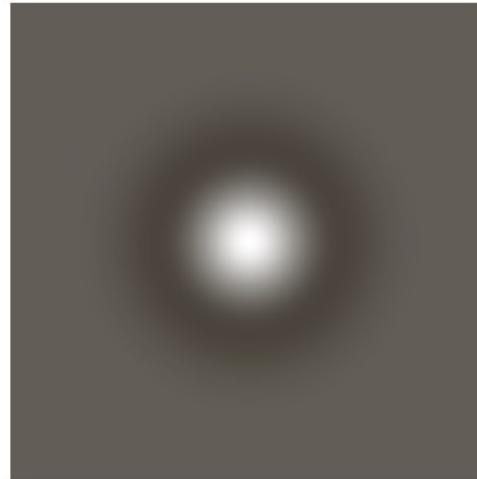
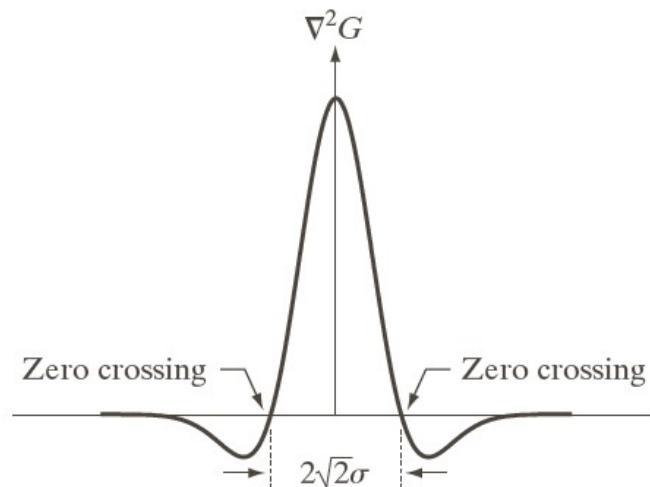
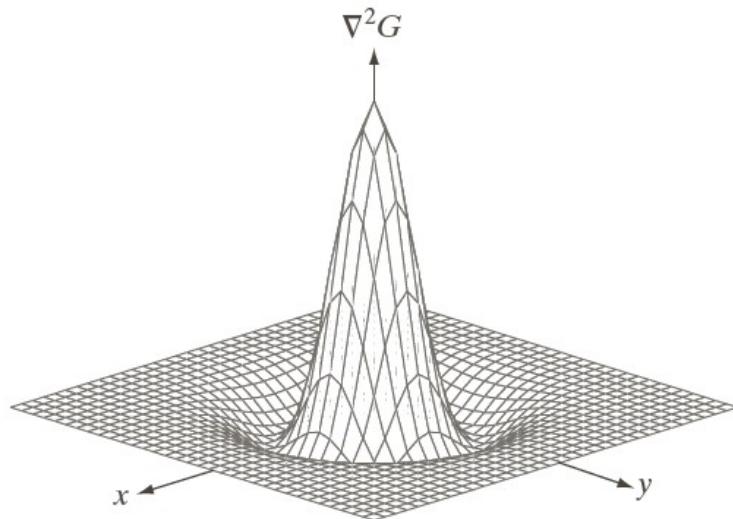
- Một đáp ứng biên
- Nhạy với nhiễu

Laplace of Gaussian (LoG)

- Laplace nhạy với nhiễu \rightarrow làm trơn ảnh trước khi nhân chập với laplace
- $I * G * L = I * (G * L) \rightarrow G * L$: bộ lọc LoG
- Đạo hàm bậc 2 = Ảnh * LoG

$$LoG(x, y) = -\frac{1}{\pi\sigma^4} \left[1 - \frac{x^2 + y^2}{2\sigma^2} \right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Laplace of Gaussian (LoG)



a b
c d

FIGURE 10.21

- (a) Three-dimensional plot of the *negative* of the LoG. (b) Negative of the LoG displayed as an image. (c) Cross section of (a) showing zero crossings. (d) 5×5 mask approximation to the shape in (a). The negative of this mask would be used in practice.

0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0



a
b
c
d

FIGURE 10.22

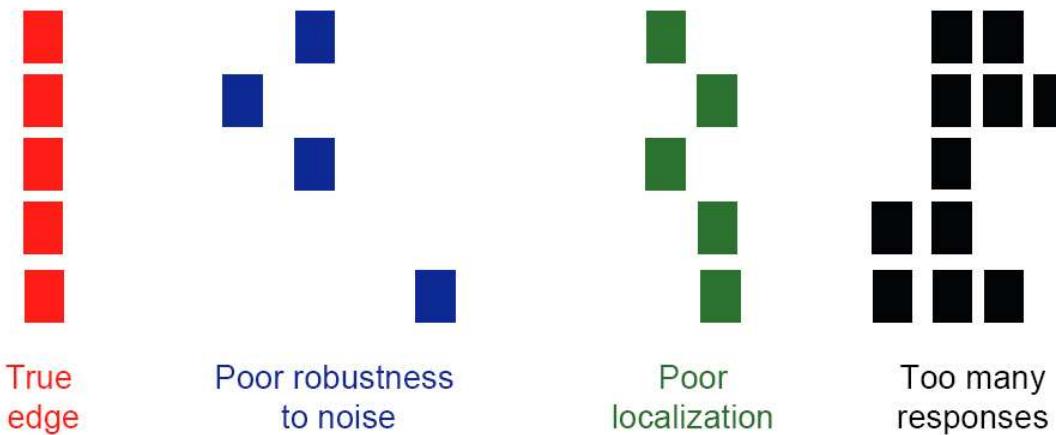
- (a) Original image of size 834×1114 pixels, with intensity values scaled to the range $[0, 1]$.
- (b) Results of Steps 1 and 2 of the Marr-Hildreth algorithm using $\sigma = 4$ and $n = 25$.
- (c) Zero crossings of (b) using a threshold of 0 (note the closed-loop edges).
- (d) Zero crossings found using a threshold equal to 4% of the maximum value of the image in (b). Note the thin edges.

Còn gọi phương pháp Marr–Hildreth (tên 2 tác giả)

Bộ phát hiện cạnh “tối ưu”

- Tiêu chí:

- **Good detection:** bộ phát hiện tối thiểu "nhận nhầm" cũng như "bỏ sót"
- **Good localization:** cạnh phát hiện phải gần biên đúng nhất
- **Single response:** bộ phát hiện 1 điểm duy nhất tại vị trí biên; nghĩa là tối thiểu số lượng các cực đại địa phương xung quanh cạnh



Bộ phát hiện Canny

- Được sử dụng phổ biến nhất
- Giả thiết mô hình:
 - Những cạnh “step” bị ảnh hưởng bởi nhiễu cộng Gaussian
- Canny:
 - Đào hàm bậc 1 của Gaussian gần đúng với toán tử tối ưu hóa tỉ lệ tín hiệu/nhiễu (signal-to-noise) và phát hiện biên đúng
- Đặc điểm :
 - Detection: **phát hiện được cạnh yếu**
 - Good location: **gần với biên thực**
 - Unique response: **độ dày cạnh = 1**

Bộ phát hiện Canny: các bước

1) Áp dụng lọc với bộ lọc Gaussian

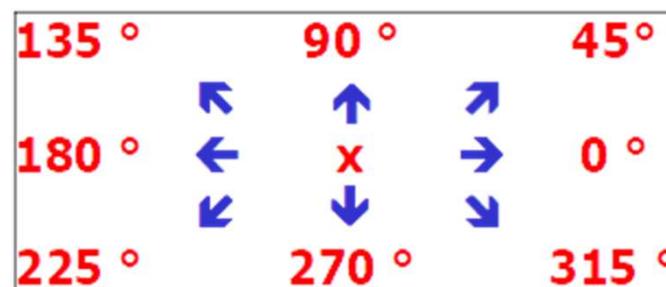
- Bộ lọc thông thấp để lọc nhiễu

2) Tính độ lớn gradient của các điểm ảnh

- Áp dụng bộ lọc Sobel theo hướng X và Y
- Tính độ lớn gradient $|G| = |G_x| + |G_y|$

3) Tính hướng gradient

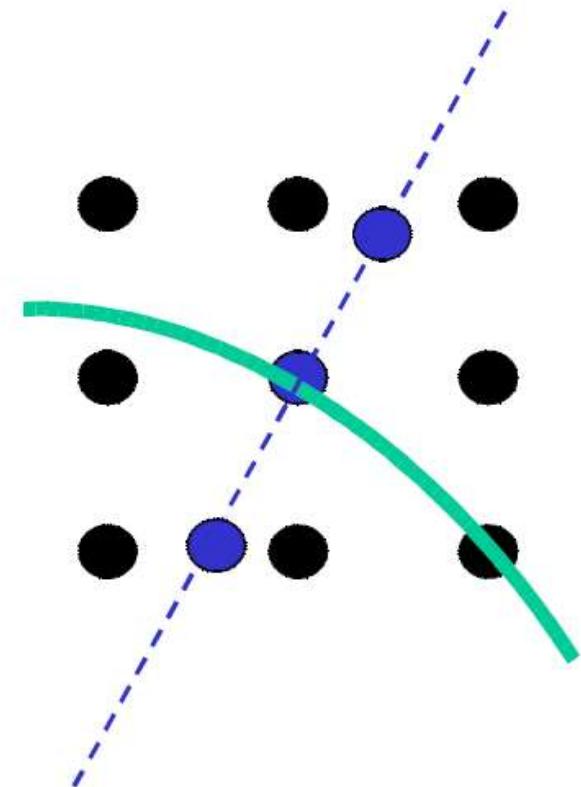
- Hướng gradient $\theta = \arctan(G_y / G_x)$
- Làm tròn về 8 hướng cách nhau 45°



Bộ phát hiện Canny: các bước

4) Loại bỏ các điểm không phải cực trị (non-maxima suppression)

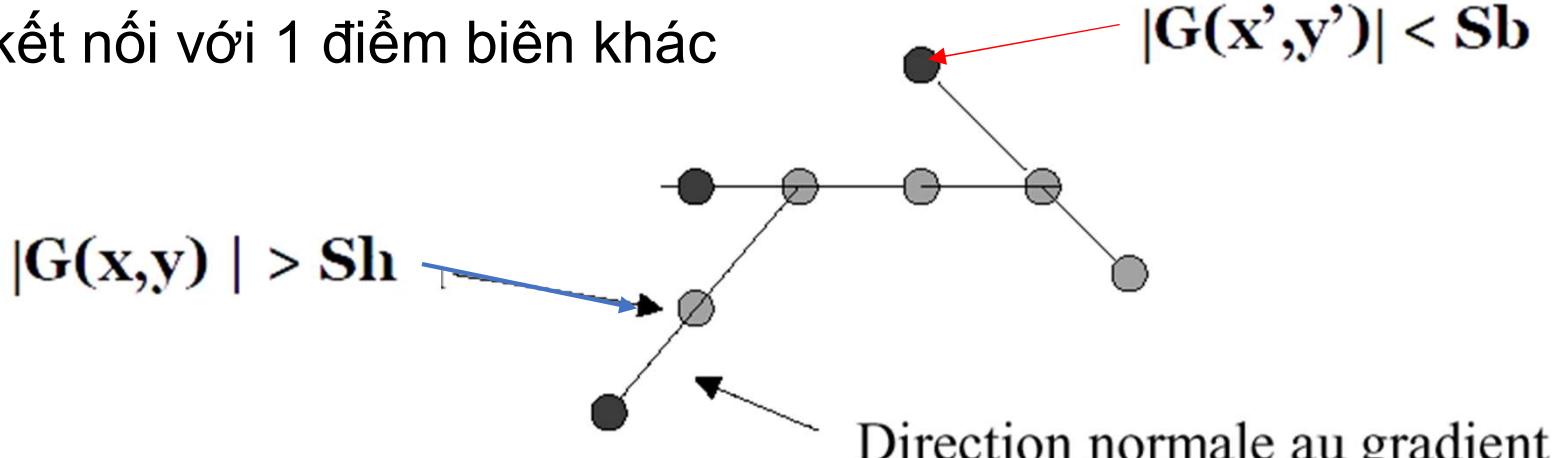
- Nếu độ lớn gradient tại điểm (x,y)
bé hơn 1 trong 2 hàng xóm theo
hướng gradient
- đặt giá trị độ lớn gradient tại
 (x,y) bằng 0



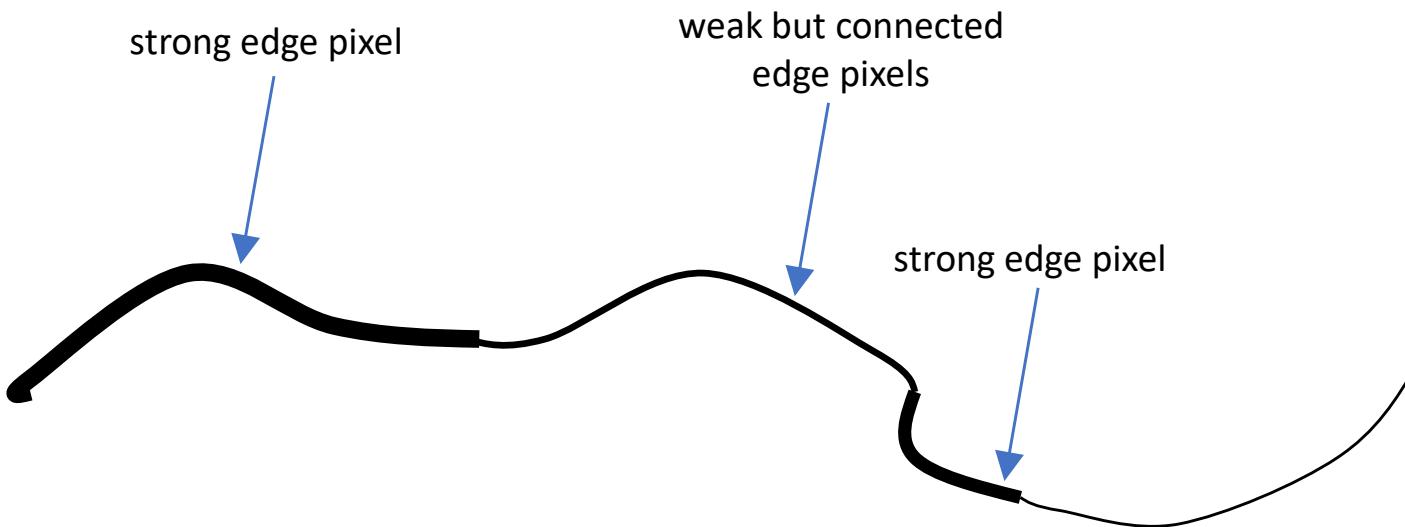
Bộ phát hiện Canny: các bước

5) Lấy ngưỡng (hysteresis thresholding)

- Sử dụng 2 ngưỡng: một ngưỡng cao (S_h) và một ngưỡng thấp (S_b)
- Tại mỗi điểm:
 - IF $\text{magnitude}(x,y) < S_b$, THEN đặt pixel = 0 (không phải biên)
 - IF $\text{magnitude}(x,y) > S_h$, THEN đặt pixel = 1 (biên)
 - IF $S_b \leq \text{magnitude}(x,y) \leq S_h$, THEN pixel là biên nếu nó được kết nối với 1 điểm biên khác



Hysteresis thresholding



Source: S. Seitz

Bộ phát hiện Canny

Ảnh đầu vào



Sobel



Non-maxima suppression



Cắt ngưỡng

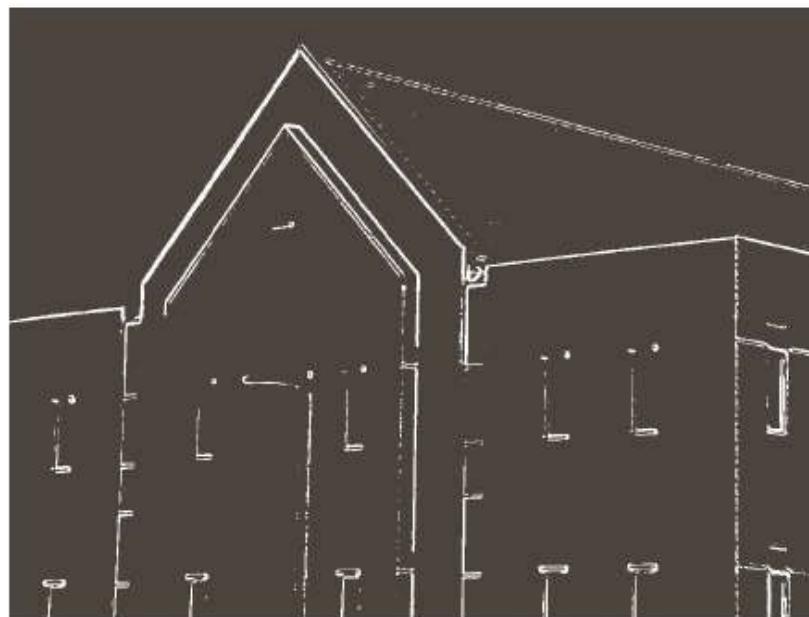


a b
c d

FIGURE 10.25

- (a) Original image of size 834×1114 pixels, with intensity values scaled to the range $[0, 1]$.
(b) Thresholded gradient of smoothed image.
(c) Image obtained using the Marr-Hildreth algorithm.
(d) Image obtained using the Canny algorithm. Note the significant improvement of the Canny image compared to the other two.

$$T_L = 0.04; T_H = 0.10; \sigma = 4 \text{ and a mask of size } 25 \times 25$$





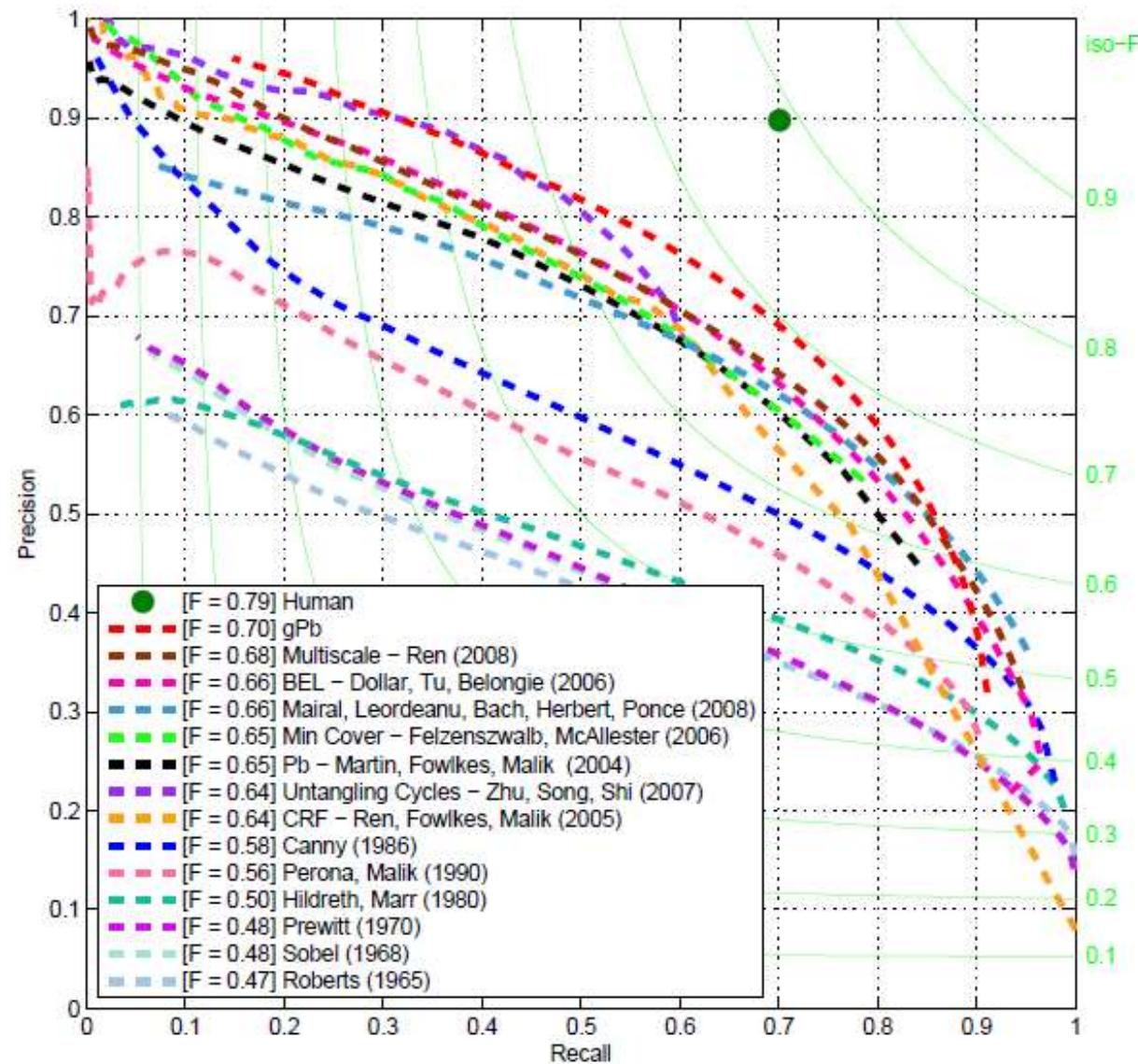
a b
c d

FIGURE 10.26

- (a) Original head CT image of size 512×512 pixels, with intensity values scaled to the range $[0, 1]$.
 - (b) Thresholded gradient of smoothed image.
 - (c) Image obtained using the Marr-Hildreth algorithm.
 - (d) Image obtained using the Canny algorithm.
- (Original image courtesy of Dr. David R. Pickens, Vanderbilt University.)

$$T_L = 0.05; T_H = 0.15; \sigma = 2 \text{ and a mask of size } 13 \times 13$$

45 năm phát triển các kỹ thuật phát hiện biên ảnh



Source: Arbelaez, Maire, Fowlkes, and Malik. TPAMI 2011 (pdf)

Nội dung

- Vai trò của biên và cách tiếp cận chung
- Phát hiện biên
 - Image gradient: sobel, prewitt
 - Canny detector
 - Laplacian
- Phát hiện đường thẳng
 - Hough transform
 - Ransac

Hough transform (HT)

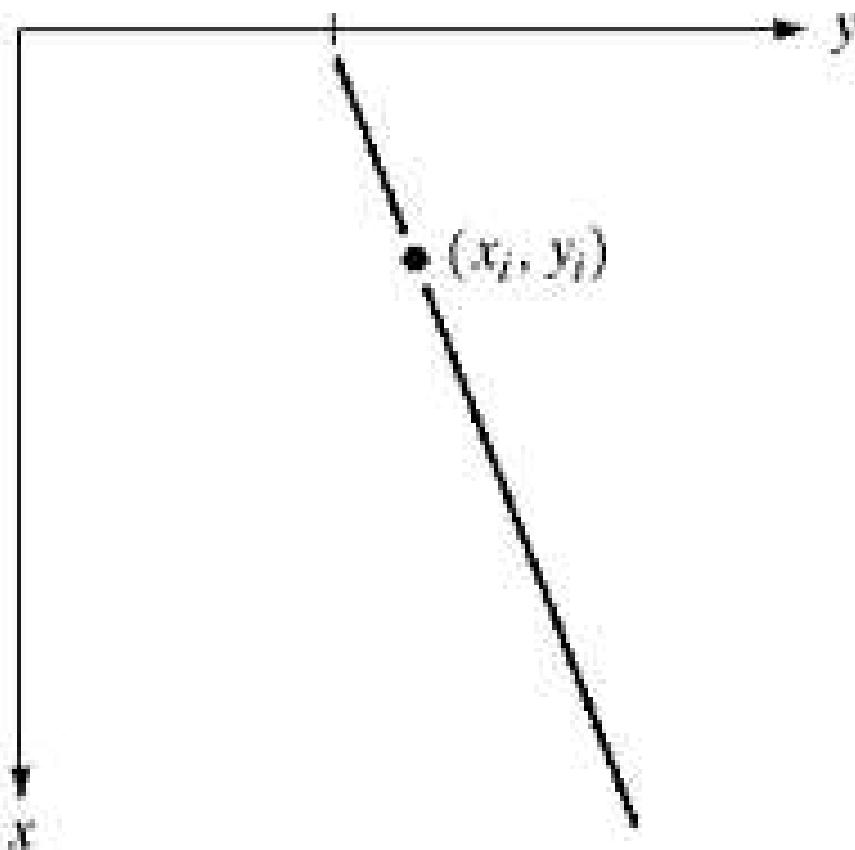
- HT
 - Dùng để phát hiện các đường thẳng
 - Giới thiệu năm 1962 (Hough 1962) và đầu tiên được sử dụng để phát hiện đường năm 1972 (Duda 1972)
 - Goal: tìm vị trí các đường thẳng trong ảnh
- HT có thể phát hiện các đường thẳng (*lines*), đường tròn (*circles*) và các cấu trúc khác (*structures*) Chỉ nếu *phương trình tham số* là xác định
- Phát hiện **hiệu quả** khi có *nhiều* hay bị *che khuất* 1 phần

Hough transform

- Cách tiếp cận **toàn cục (Global)** để phát hiện các cạnh/biên liên tục
 - *Từ mặt phẳng (plane) x-y sang mặt phẳng tham số (parametric plane) a-b*
- **Mặt phẳng x-y**
 - $y_i = a x_i + b$
 - Một số lượng vô hạn đường đi qua cặp điểm (x_i, y_i)
 - Một đường duy nhất cho mỗi cặp (a, b)
- **Mặt phẳng tham số a-b**
 - $b = -x_i a + y_i$
 - Một đường đơn cho mỗi cặp (x_i, y_i)
 - Một số lượng vô hạn đường đi qua cặp điểm (a, b)

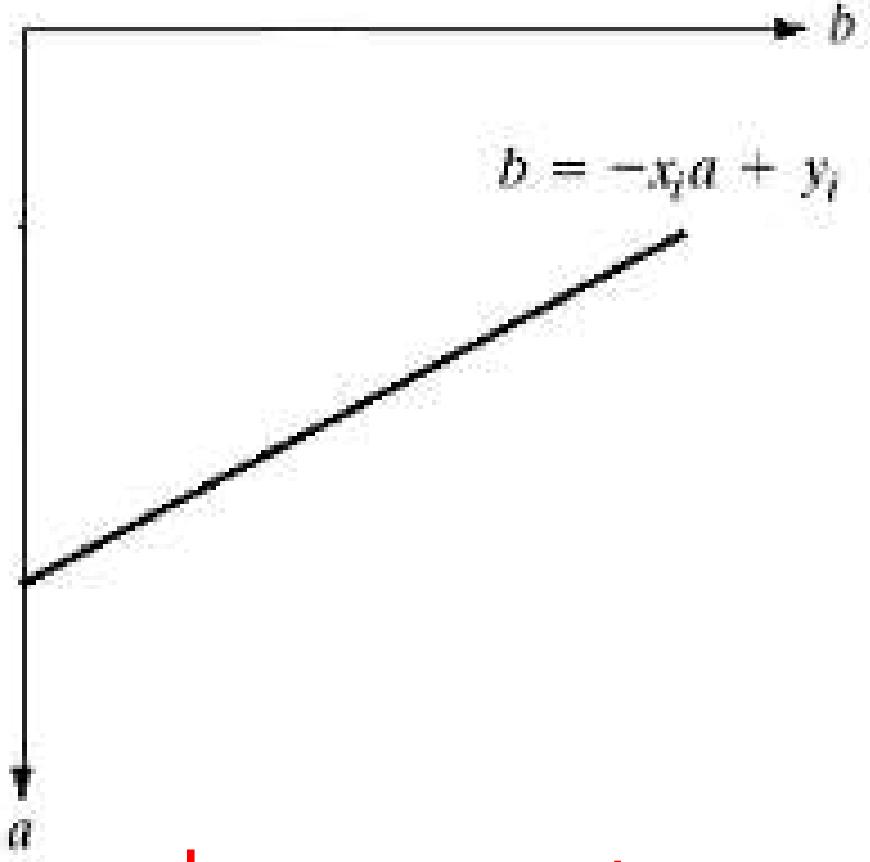
Mặt phẳng x-y và a-b

x-y plane



$$y_i = a x_i + b$$

a-b parametric plane

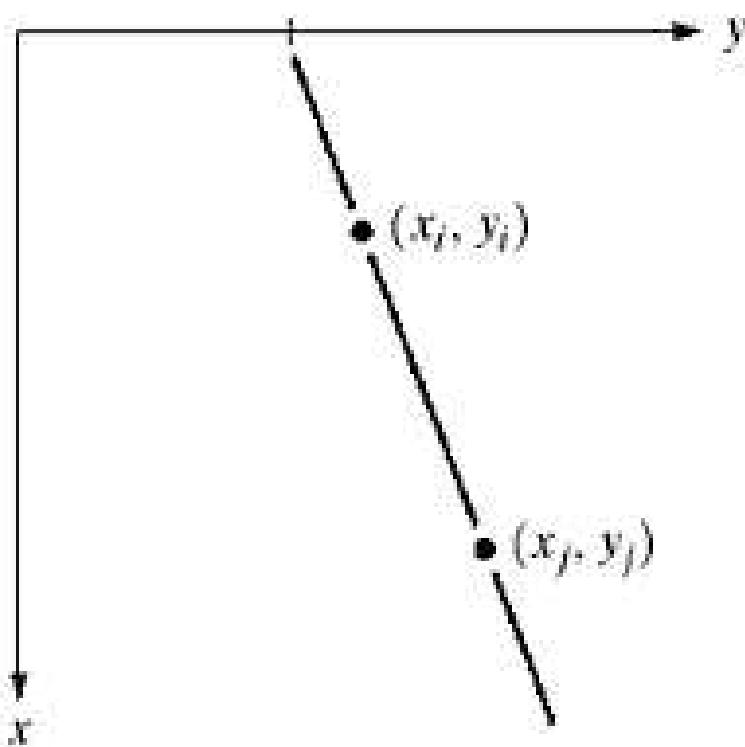


$$b = -x_i a + y_i$$

Đường vs Điểm

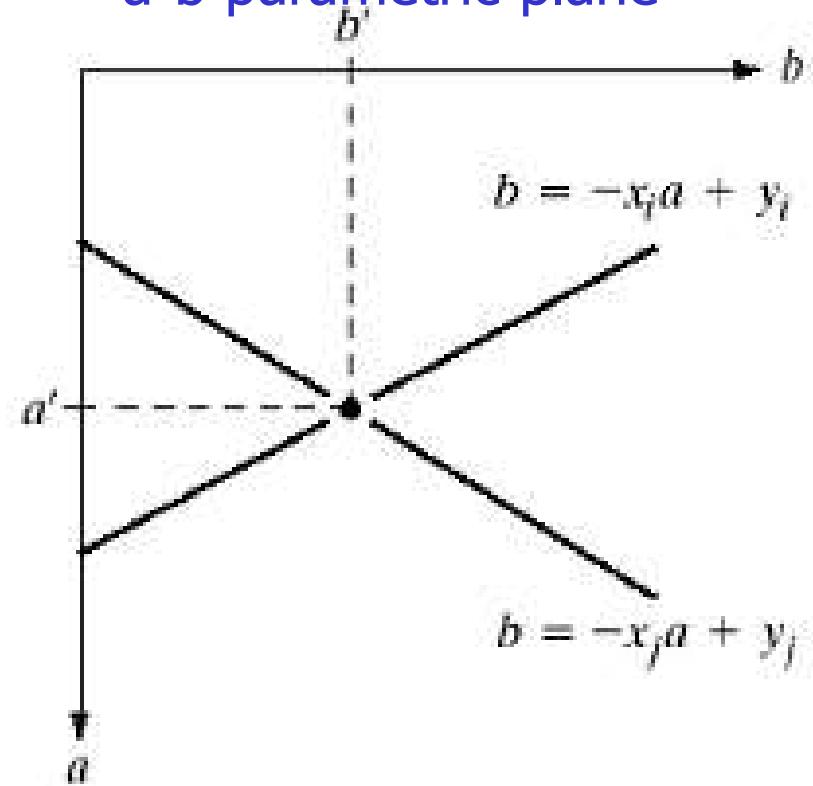
Tất cả các điểm (x, y) trên một đường thẳng trong mặt phẳng **x-y plane** đi qua một điểm đơn (a', b') trong mặt phẳng tham số **a-b**

x-y plane



$$y_i = a x_i + b$$

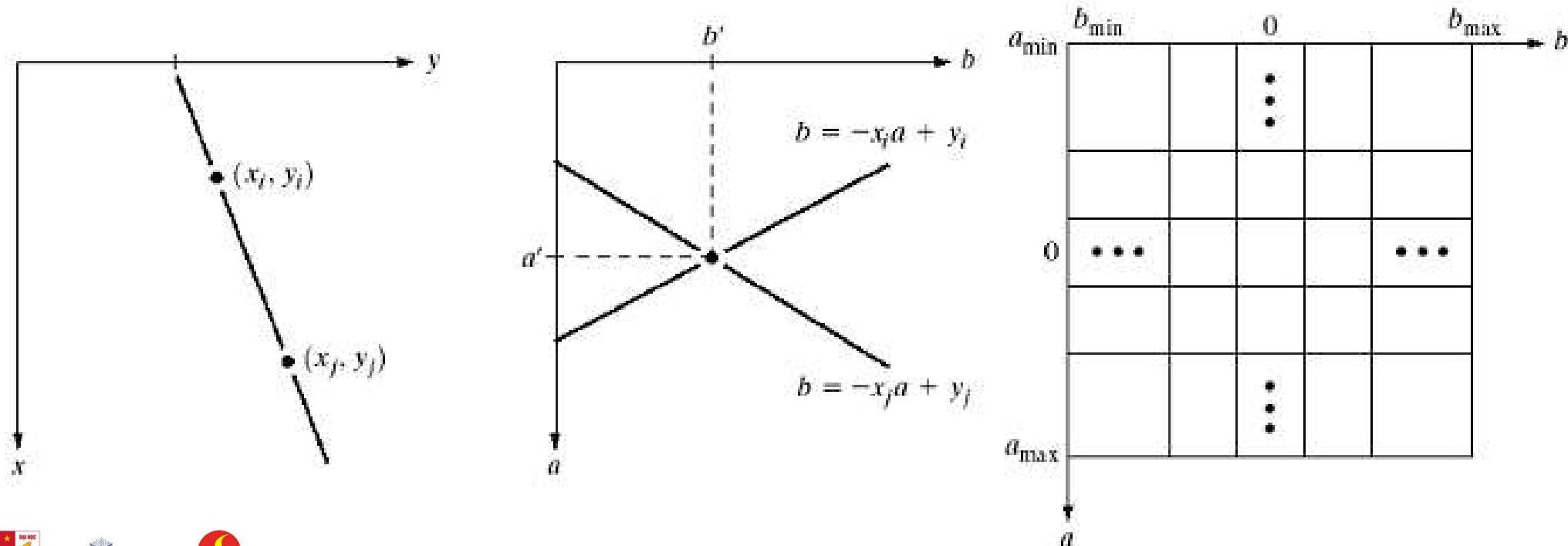
a-b parametric plane



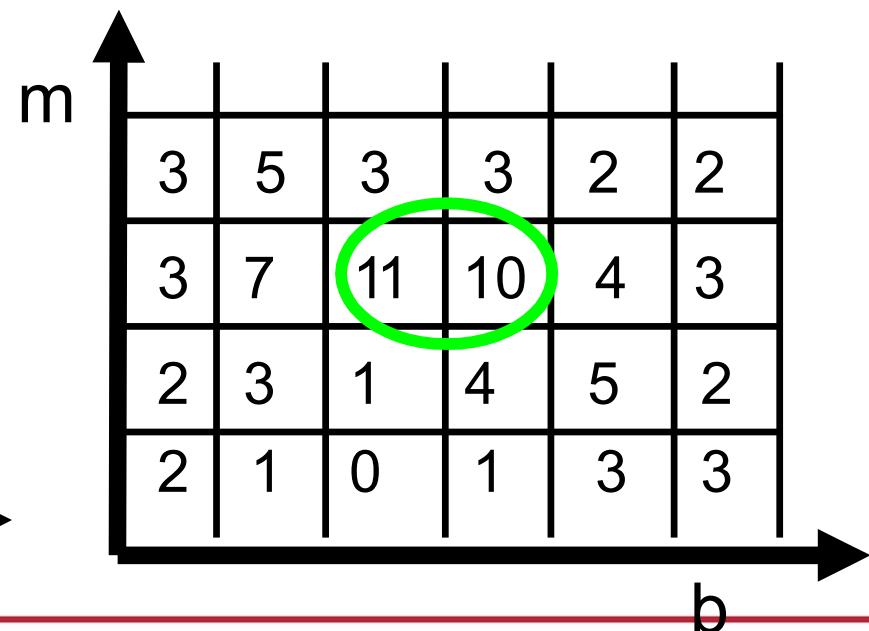
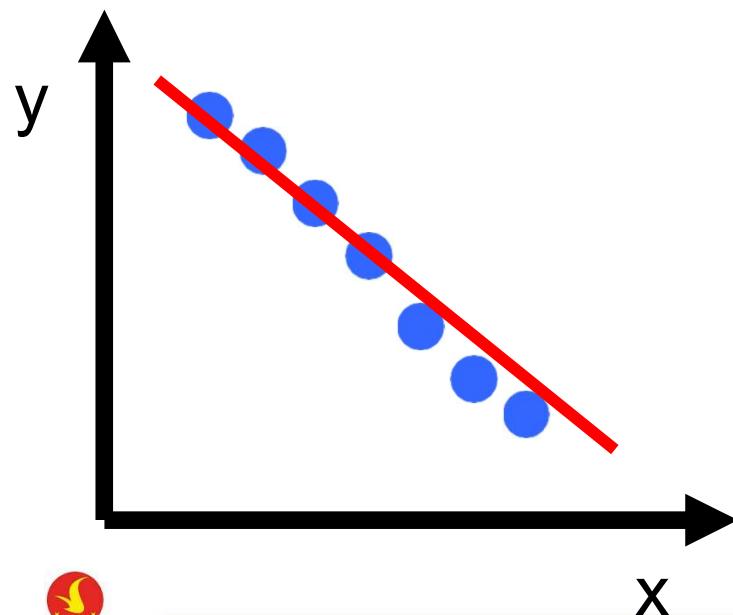
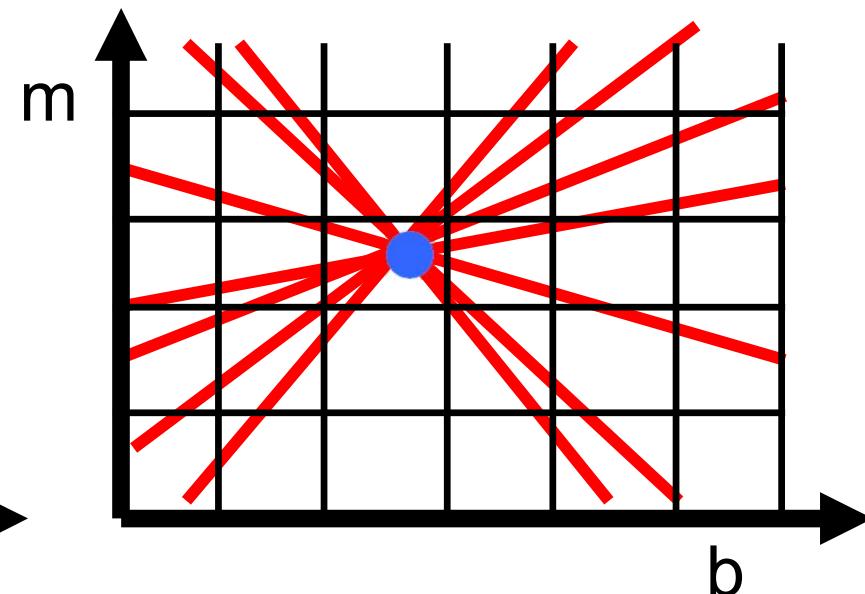
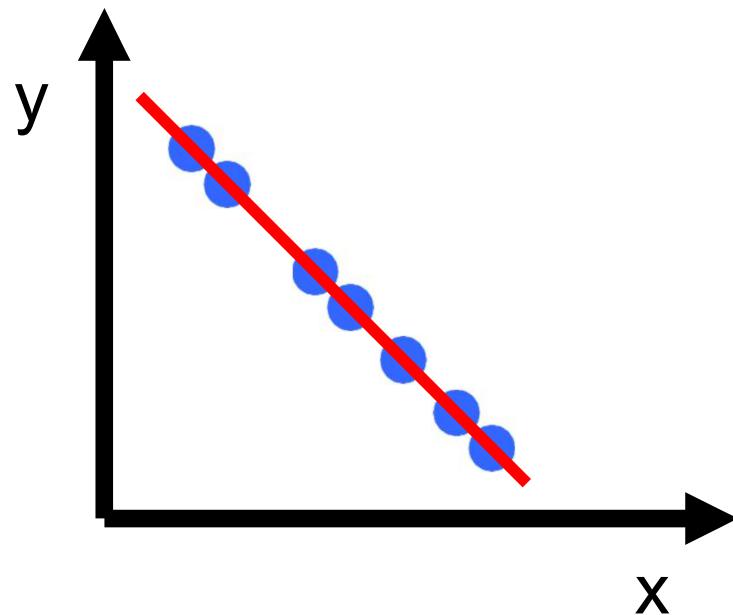
$$b = -x_i a + y_i$$

Biến đổi Hough: ý tưởng chính

- Tạo ma trận tính tích lũy - M (a,b)
- **Bỏ phiếu**
 - Mỗi điểm bỏ phiếu cho đường đi qua nó
- Đường nào nhận được nhiều phiếu thì được giữ lại



Biến đổi Hough



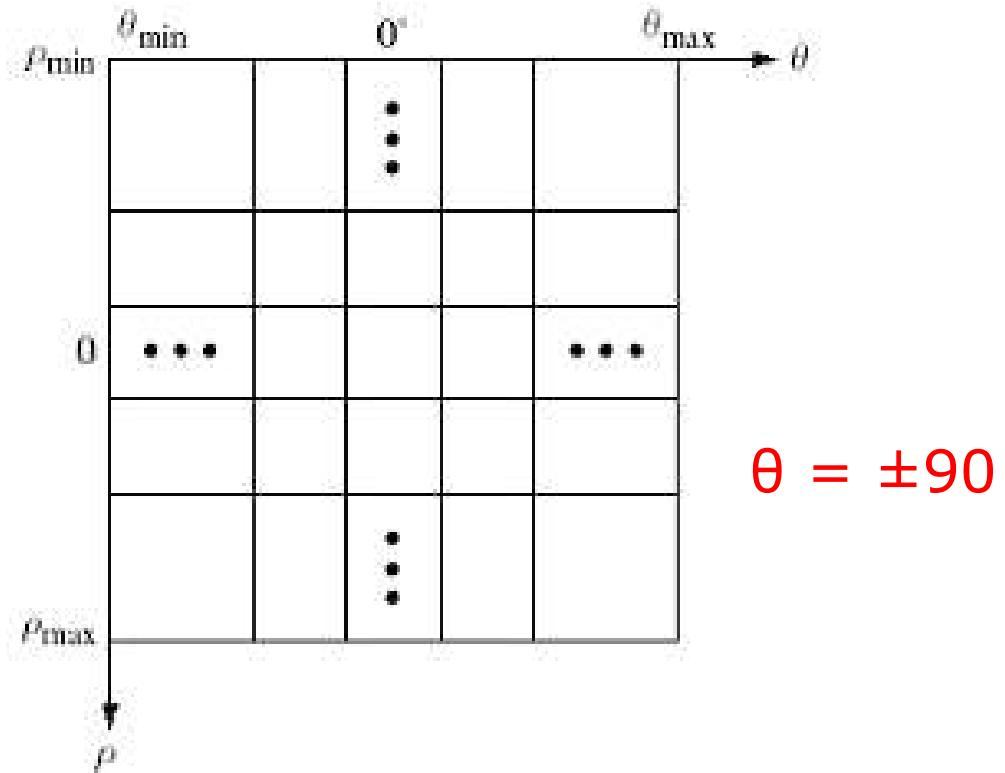
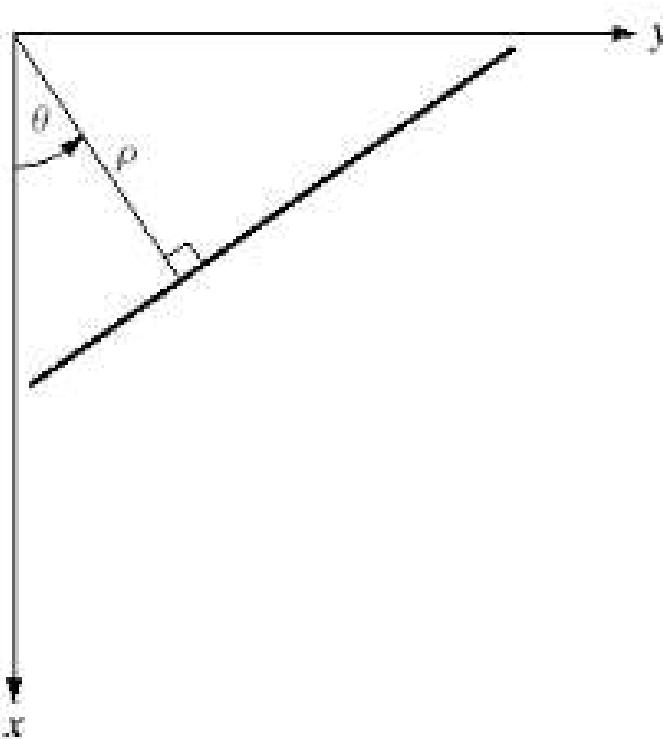
Thực hiện biến đổi Hough

- Xác định **các điểm biên** của ảnh đầu vào
 - *Sobel, Prewitt, Canny, ...*
- Với mỗi điểm biên, tính 1 đường trong không gian **(a,b)**
 - *1 đường thẳng trong kgian a-b tương ứng 1 điểm biên trong kgian (x,y)*
- Các **điểm đạt cực đại** trong không gian tham số a-b tương ứng với các **đường có nhiều điểm nhất** trong không gian x-y
 - *Điểm giao của các đường trong kgian a-b là tương ứng các đường thẳng trong kgian x-y*

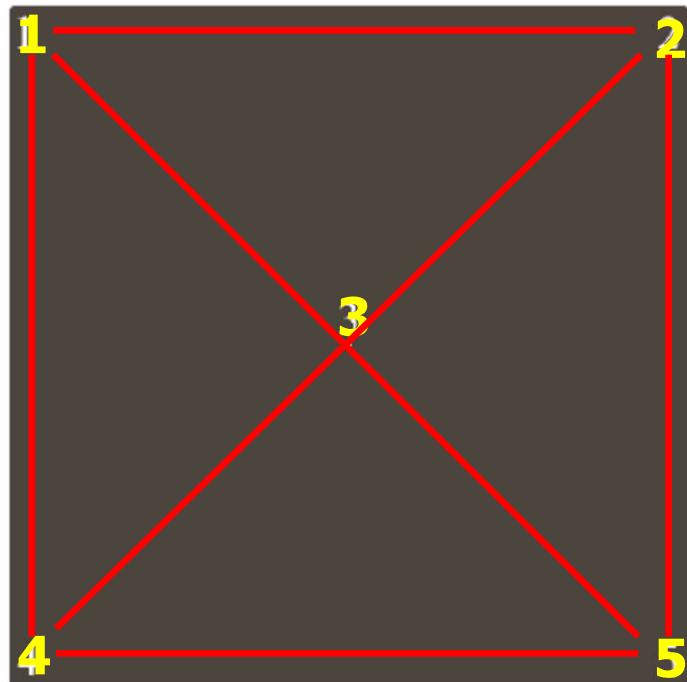
Vấn đề với không gian (a,b)

- Vấn đề: Với mỗi đường theo trực dọc, $a=\infty$!
- Giải pháp: biểu diễn sử dụng hệ tọa độ cực (ρ, θ)

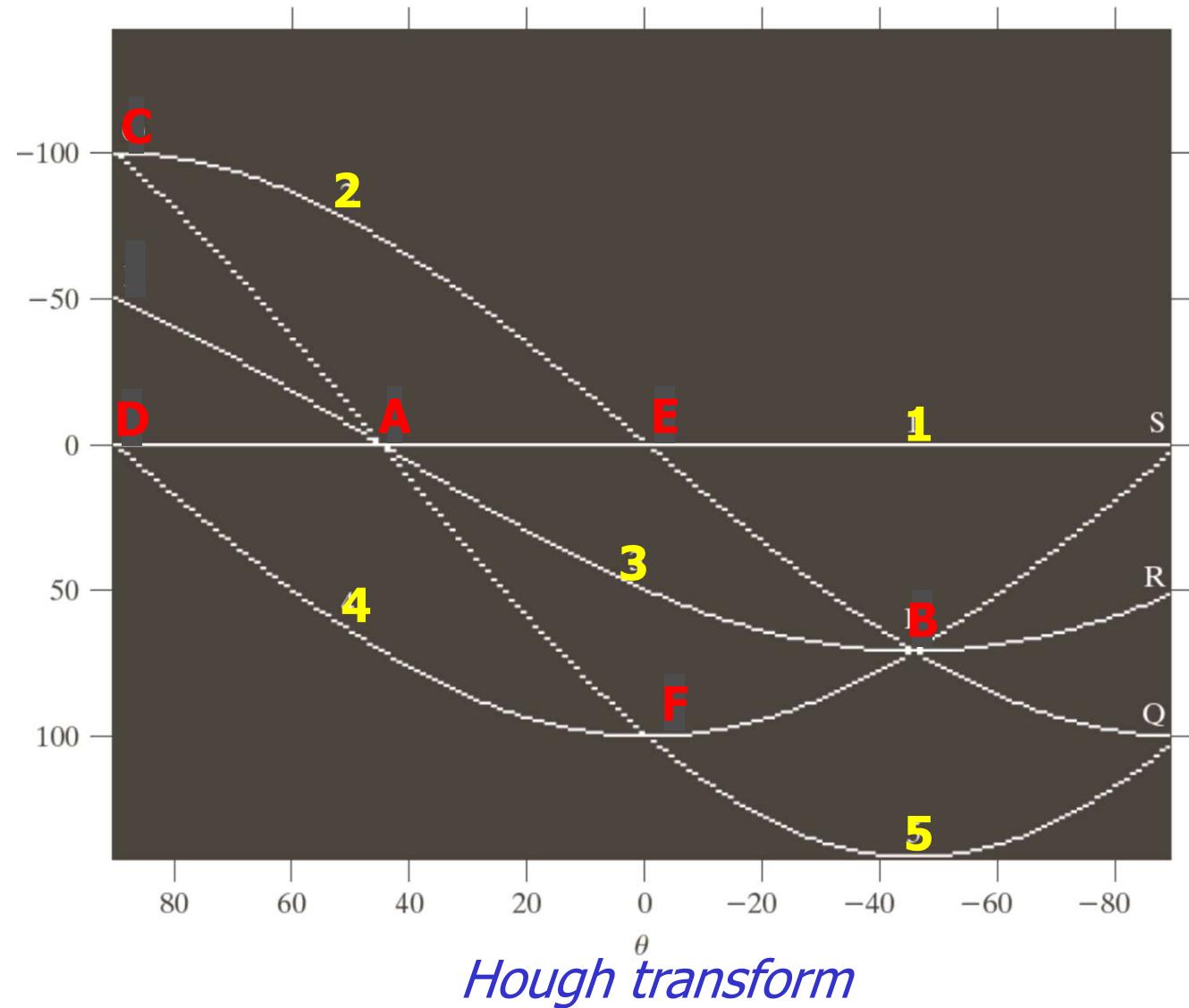
$$\rho = x \cos \theta + y \sin \theta$$



Biến đổi Hough: ví dụ



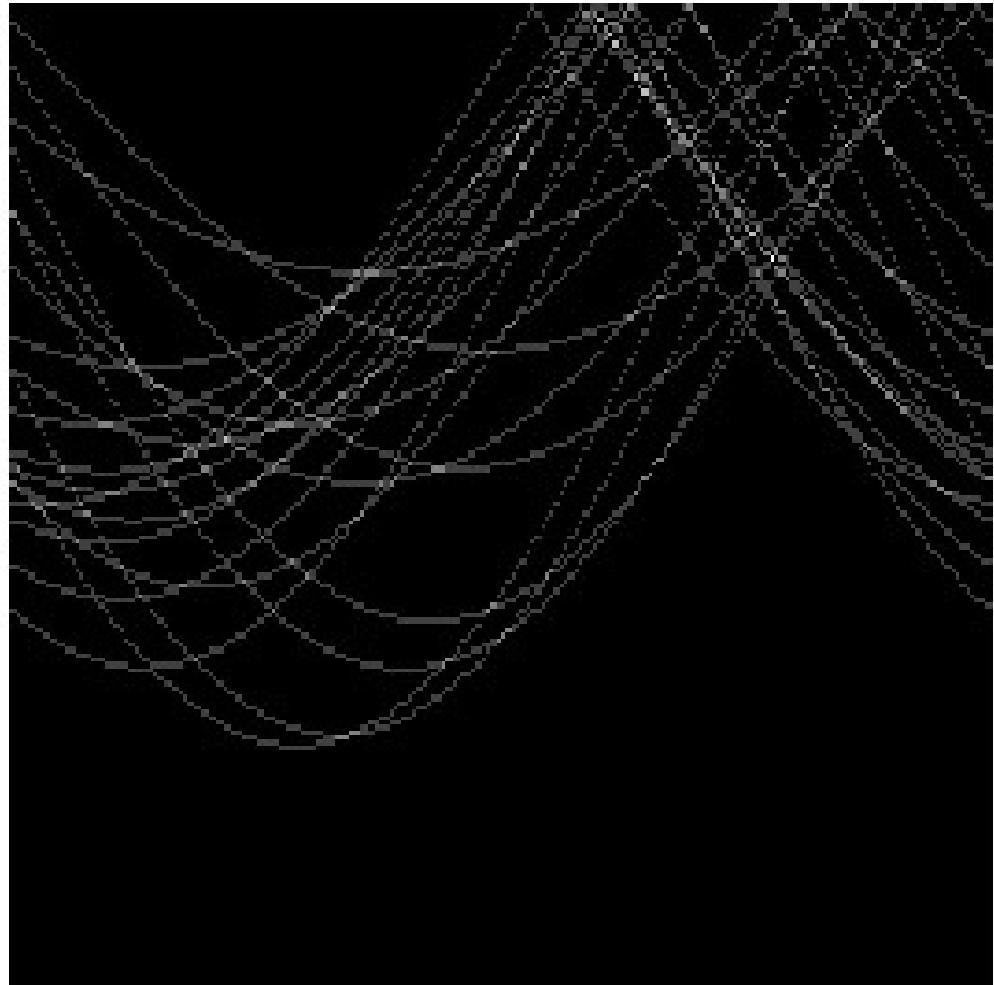
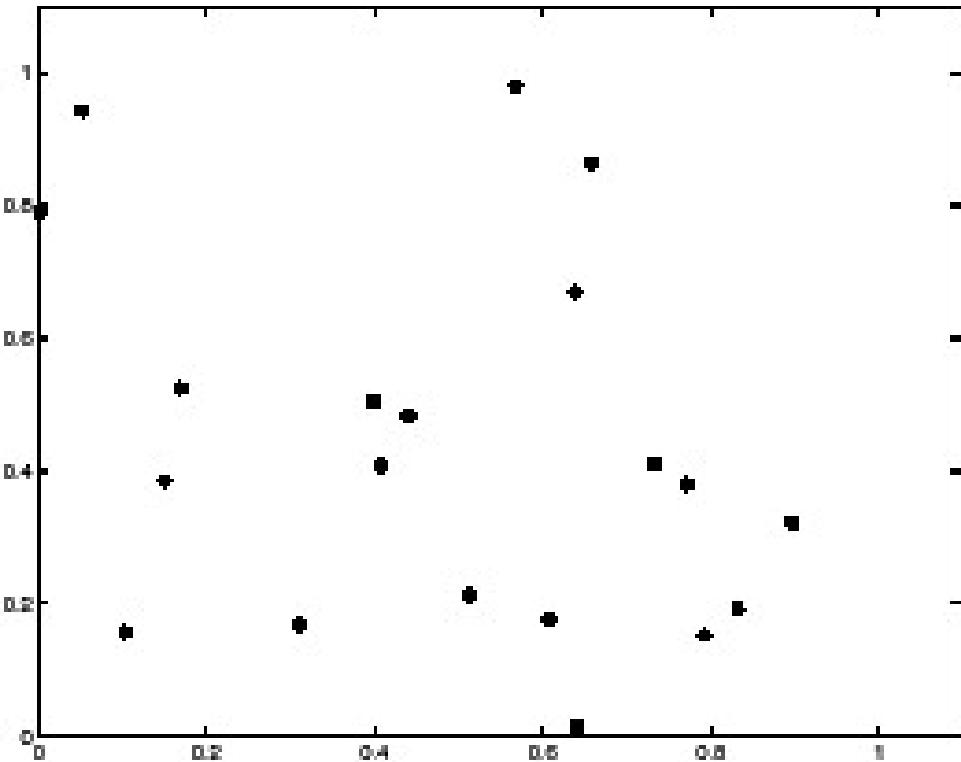
Ảnh với 5 điểm



Hough transform

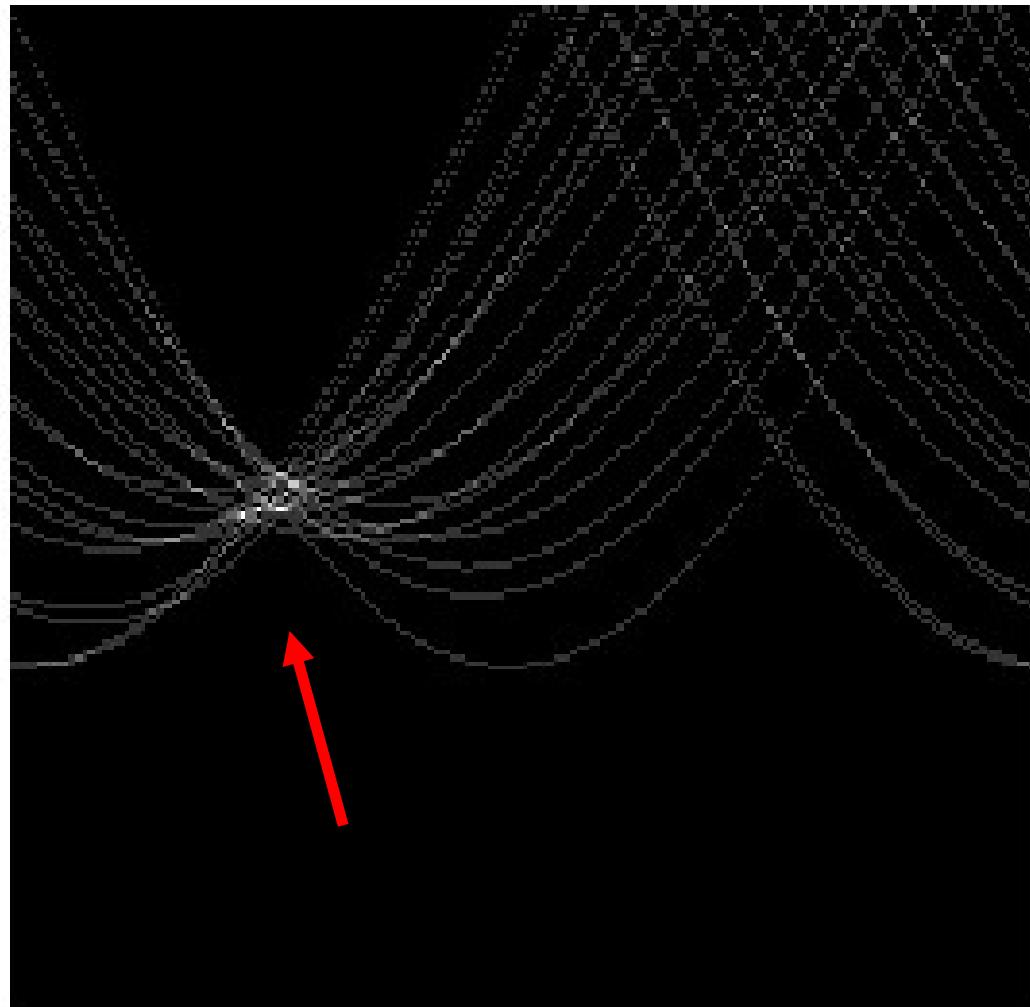
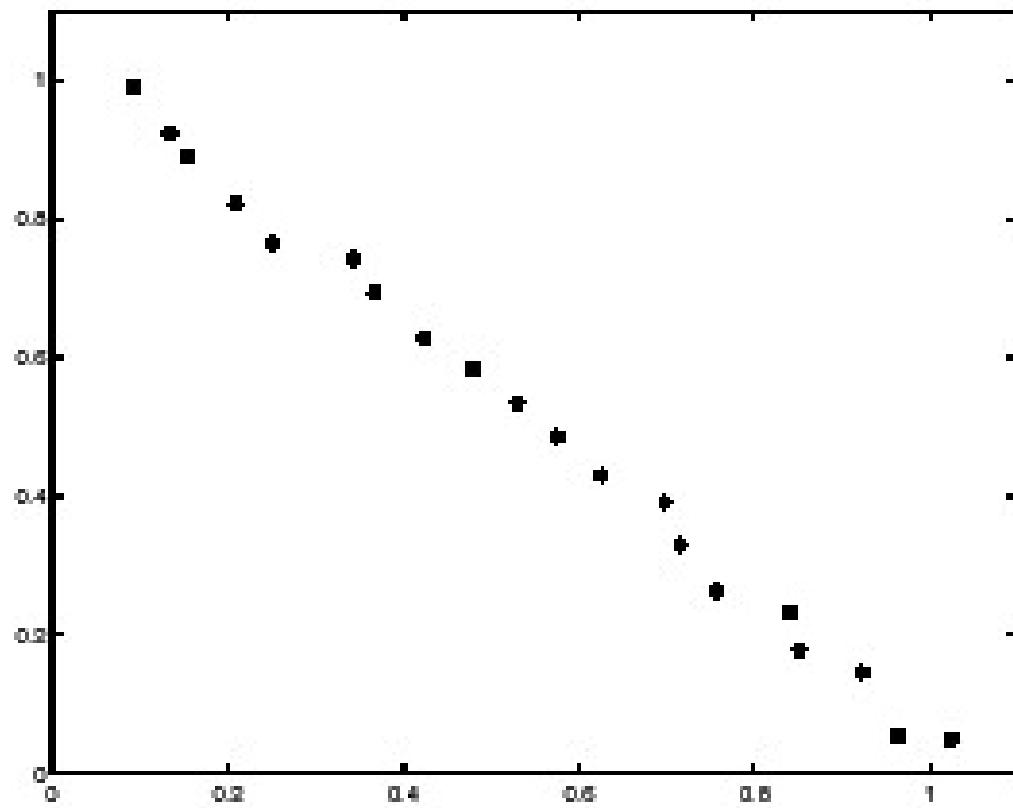
Source : Gonzalez and Woods. Digital Image Processing 3ed. Prentice-Hall, 2008.

HT: ví dụ với tập điểm ngẫu nhiên



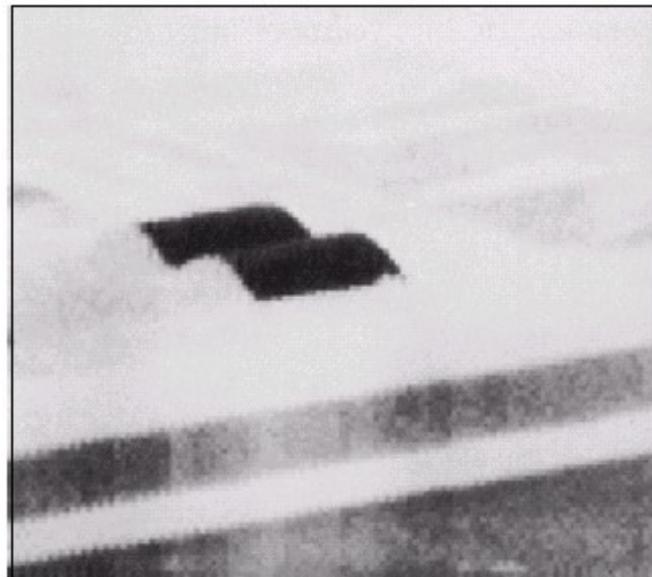
Phép biến đổi của các điểm ngẫu nhiên không cho kết quả nào chính xác

HT: ví dụ với tập điểm tạo thành đường thẳng

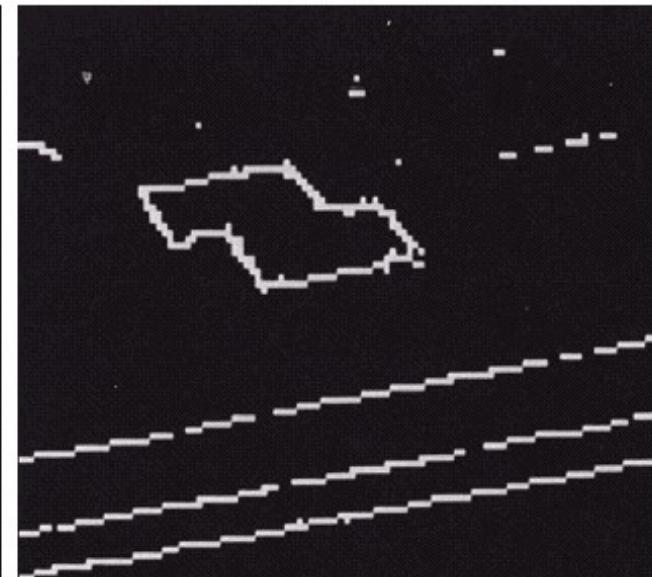
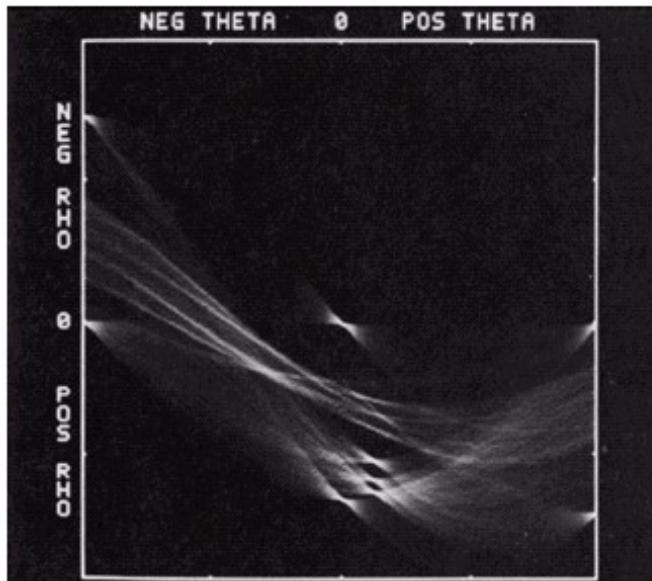


HT: ví dụ

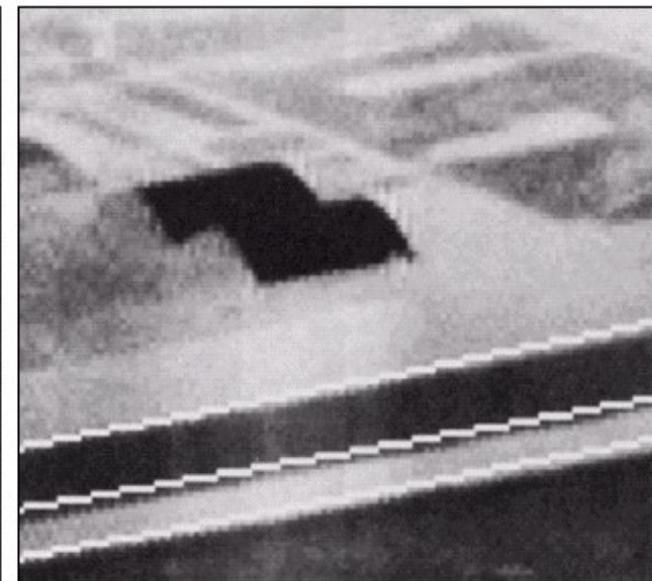
Ảnh



Hough



Gradient



Final

Ví dụ

Image



Hough



Canny

Final

Lines



Source : Gonzalez and Woods. Digital Image Processing. Prentice-Hall 3ed, 2008.

Đánh giá về biến đổi Hough

- **Ưu điểm**

- Không nhạy với ngoại lệ, mỗi điểm bỏ phiếu độc lập với nhau
- Khá hiệu quả (thường nhanh hơn là việc thiết lập một tập các tham số)
- Có nhiều lựa chọn các hàm so khớp (fit) tốt

- **Nhược điểm**

- Có nhạy với nhiễu
- “Bin size” quyết định sự đánh đổi giữa chống chịu nhiễu, độ chính xác cũng như giữa tốc độ và bộ nhớ
 - Rất khó để tìm ra điểm hoàn hảo

- **Không thích hợp cho nhiều tham số**

- Kích thước grid tăng theo cấp số nhân

- **Ứng dụng**

- Tìm đường (cũng như đường tròn, e-líp,...)
- Phát hiện đối tượng tức thời (các tham số là biến đổi affine – giữ cạnh và sự song song)
- Nhận diện lớp đối tượng (tham số là vị trí/scale)

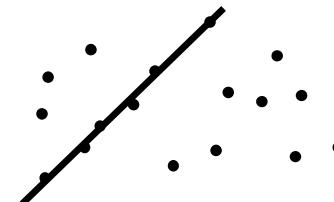
Ransac

- Phương pháp khớp mô hình:
 - Phương pháp học để ước lượng tham số cho 1 mô hình từ việc lấy mẫu ngẫu nhiên từ dữ liệu quan sát
 - Sử dụng cho:
 - Phát hiện đường thẳng (tròn, elipse, etc)
 - Bài toán về so khớp (matching between 2 sets of features): tìm phép biến đổi giữa 2 tập dữ liệu
 - ...

Khó khăn trong việc so khớp đường (Difficulty of Line Fitting)

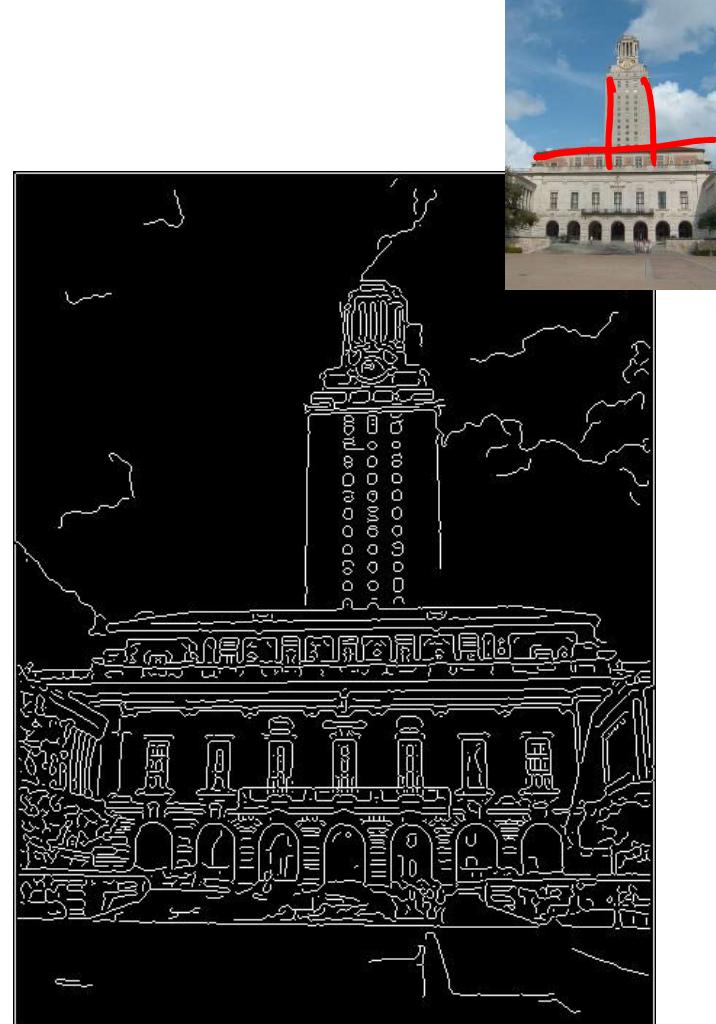
- **Những điểm biên thêm (lộn xộn), nhiều mô hình:**

– Đường nào đi theo đường nào (nếu có)?



- Một vài phần của mỗi đường được phát hiện, những phần khác của đường **bị mất**:

– Làm sao để tìm một đường mà những đoạn nối bị mất?



- **Nhiều** trong những điểm cảnh, có hướng :

– Làm sao để phát hiện ra tham số thật sự?

Bỏ phiếu (Voting)

- Không thể kiểm tra tất cả các bộ kết hợp đặc trưng bằng việc khớp (fit) một mô hình cho các tập con
- Bỏ phiếu là kỹ thuật thông dụng khi chúng ta cần **đặc trưng bỏ phiếu cho tất cả các mô hình** tương thích với nó.
 - Duyệt qua các đặc trưng, bỏ phiếu cho tham số mô hình.
 - Tìm tham số mô hình có nhiều phiếu nhất.
- Đặc trưng **Nhiễu và lộn xộn (Noise & clutter)** cũng bỏ phiếu nhưng thường phiếu của chúng không ổn định khi so với **hầu hết các đặc trưng “tốt”**.
- Nếu vài đặc trưng không được đánh giá, mô hình có thể mở rộng thành nhiều mảnh.

RANSAC

[Fischler & Bolles 1981]

- **RAN**dom **SA**mple **C**onsensus
- Cách tiếp cận:
 - Chúng ta muốn tránh tác động của điểm ngoại biên (outlier), nên tập trung vào điểm nội biên - “inliner” và chỉ sử dụng những điểm này
 - Trực giác: nếu một điểm ngoại biên được chọn để tính toán so khớp, đường kết quả sẽ không được sự hỗ trợ từ những điểm khác

RANSAC

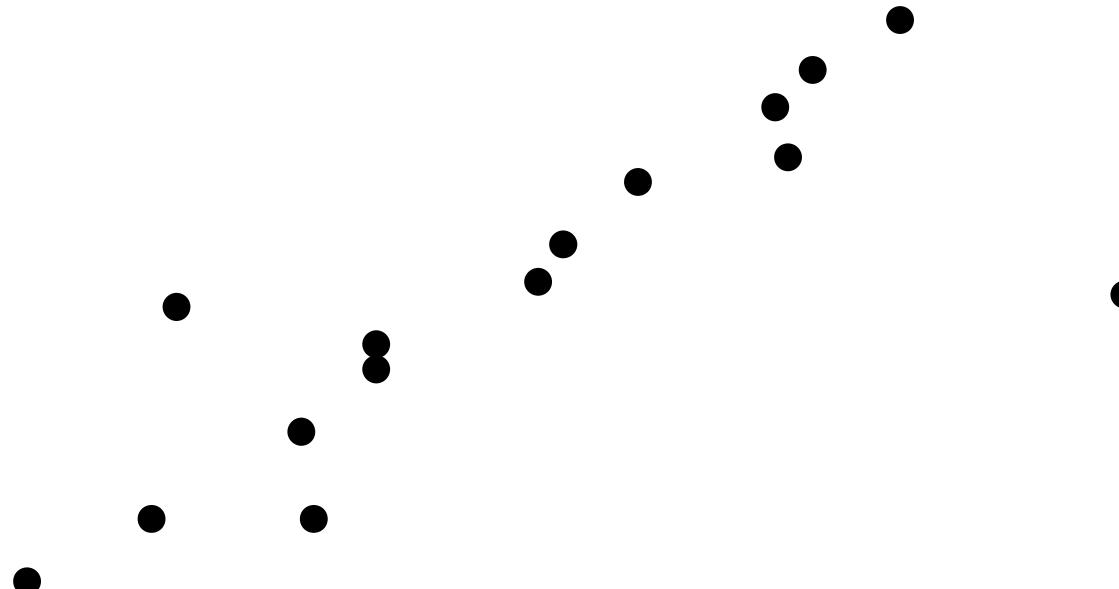
[Fischler & Bolles 1981]

RANSAC loop:

1. Ngẫu nhiên **lựa chọn** một nhóm **hạt giống** (seed group) của những điểm dựa trên ước lượng biến đổi
 2. **Tính toán biến đổi** từ nhóm hạt giống
 3. **Tìm** các “*inliers*” với biến đổi này
 4. Nếu số lượng các “inlier” đủ lớn, tính toán lại ước lượng bình phương tối thiểu (least-square) của biến đổi trên tất cả các **inlier**
- Giữ lại phép biến đổi với số lượng các **inlier** lớn nhất

Ví dụ: RANSAC Line Fitting

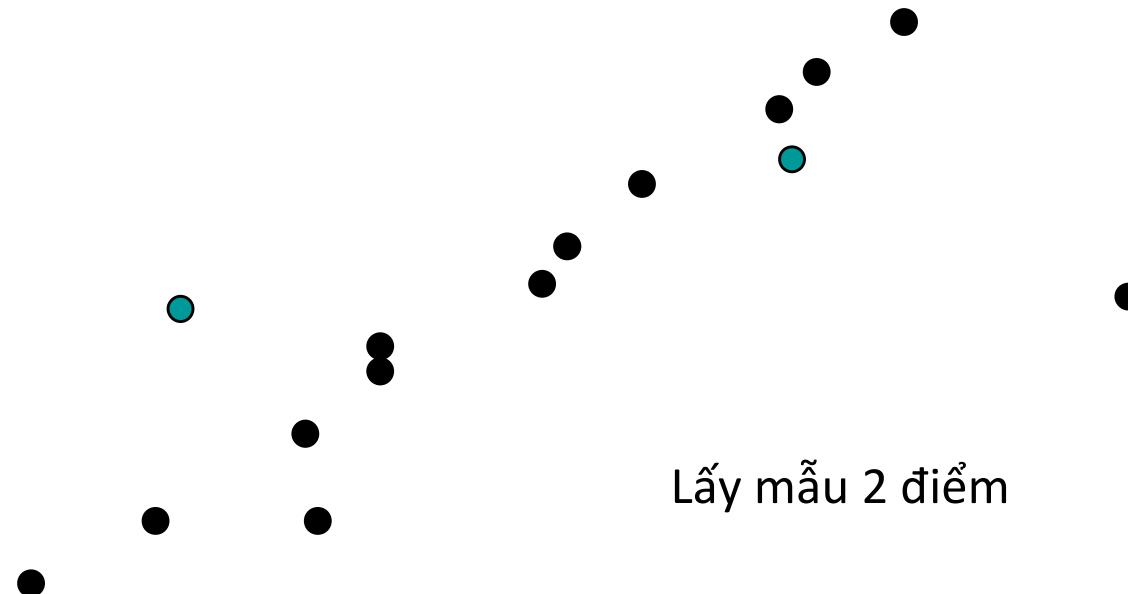
- Nhiệm vụ: Ước lượng đường tốt nhất
 - *Bao nhiêu điểm chúng ta cần để ước lượng đường?*



Slide credit: Kristen Grauman

Ví dụ: RANSAC Line Fitting

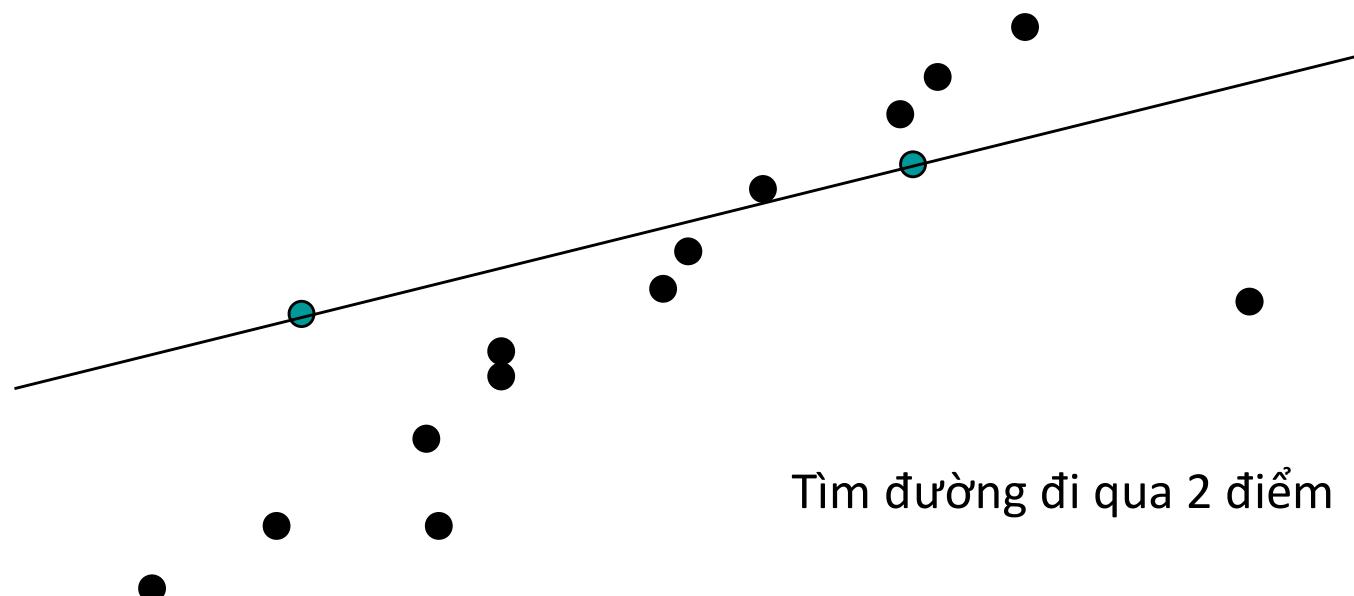
- Nhiệm vụ: Ước lượng đường tốt nhất



Slide credit: Kristen Grauman

Ví dụ: RANSAC Line Fitting

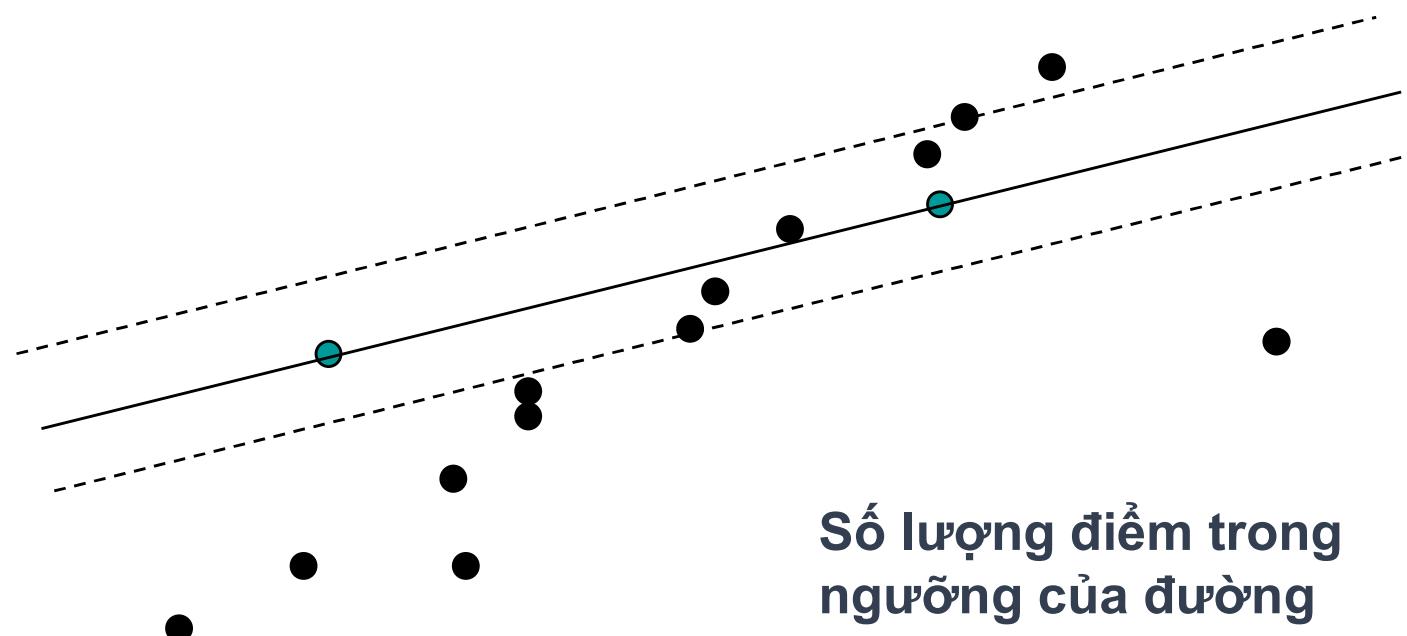
- Nhiệm vụ: Ước lượng đường tốt nhất



Slide credit: Kristen Grauman

Ví dụ: RANSAC Line Fitting

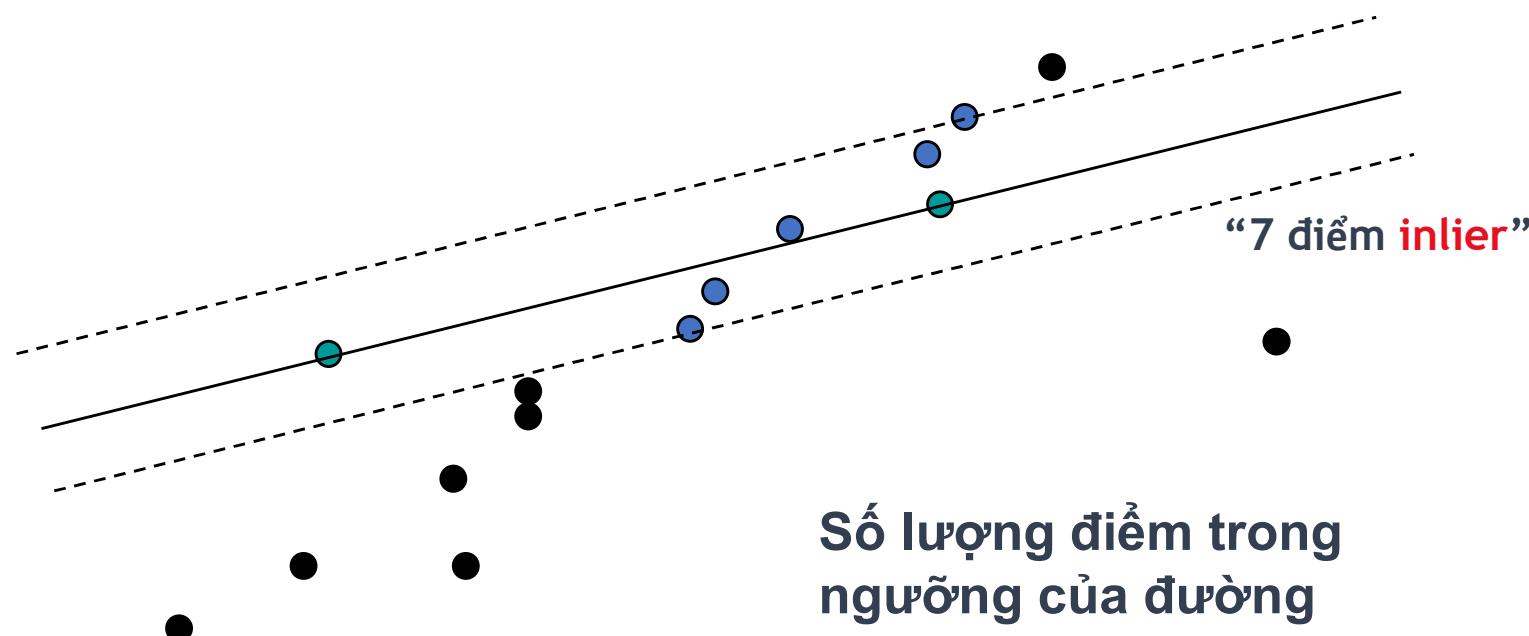
- Nhiệm vụ: Ước lượng đường tốt nhất



Slide credit: Kristen Grauman

Ví dụ: RANSAC Line Fitting

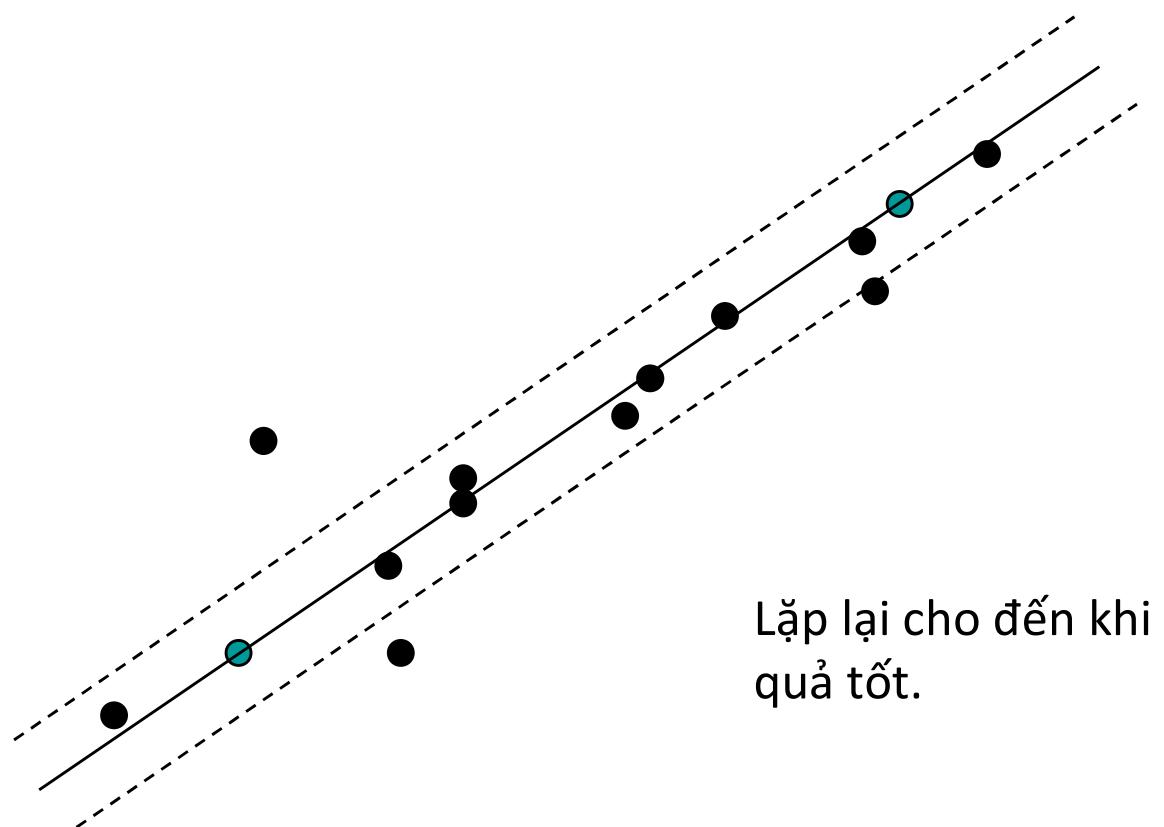
- Nhiệm vụ: Ước lượng đường tốt nhất



Slide credit: Kristen Grauman

Ví dụ: RANSAC Line Fitting

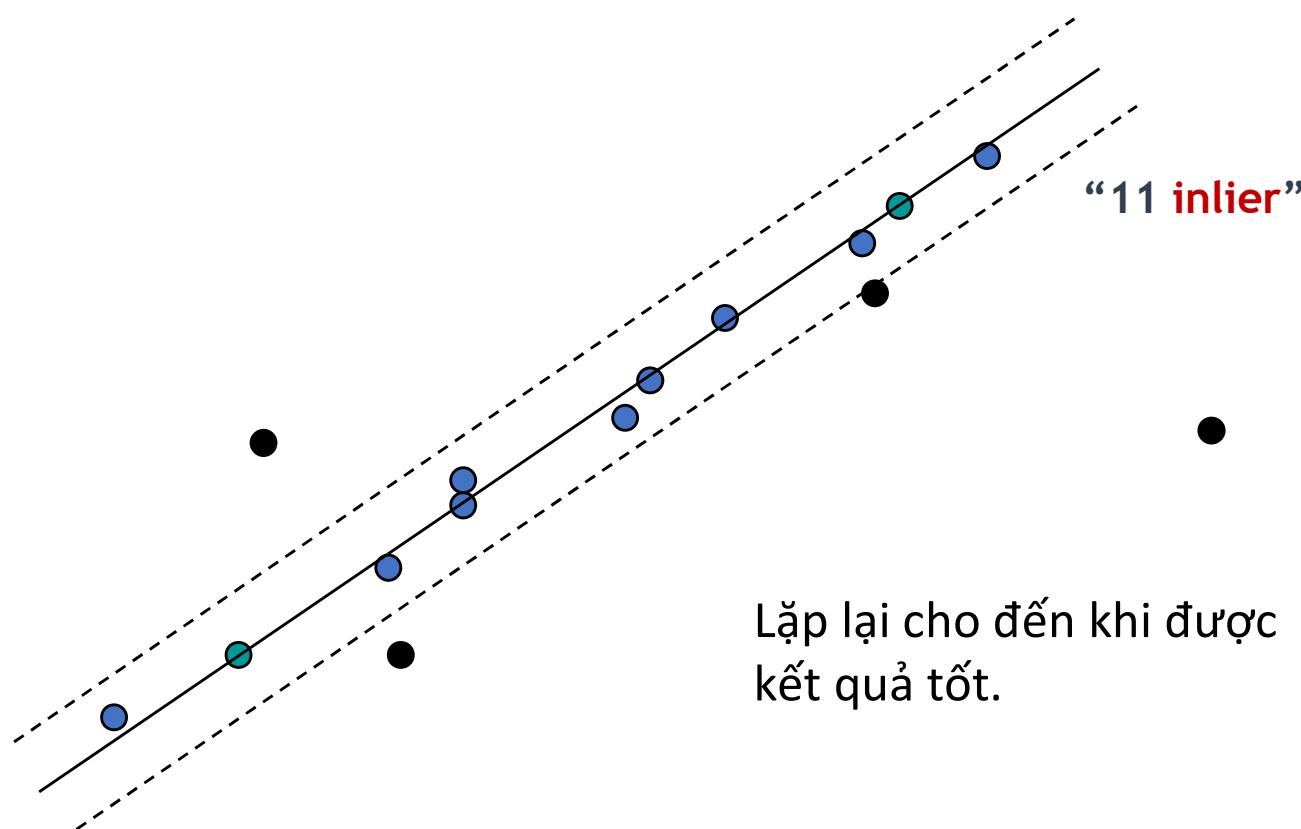
- Nhiệm vụ: Ước lượng đường tốt nhất



Slide credit: Kristen Grauman

Ví dụ: RANSAC Line Fitting

- Nhiệm vụ: Ước lượng đường tốt nhất



Slide credit: Kristen Grauman

Algorithm 15.4: RANSAC: fitting lines using random sample consensus

Determine:

n — the smallest number of points required

k — the number of iterations required

t — the threshold used to identify a point that fits well

d — the number of nearby points required

to assert a model fits well

Until k iterations have occurred

 Draw a sample of n points from the data

 uniformly and at random

 Fit to that set of n points

 For each data point outside the sample

 Test the distance from the point to the line

 against t ; if the distance from the point to the line

 is less than t , the point is close

 end

 If there are d or more points close to the line

 then there is a good fit. Refit the line using all

 these points.

end

Use the best fit from this collection, using the

fitting error as a criterion

RANSAC: Bao nhiêu mẫu?

- Cần tất cả bao nhiêu mẫu?
 - Coi w là 1 phần của inliers (những điểm thuộc đường)
 - n điểm cần để định nghĩa giả thiết (2 cho đường thẳng)
 - k mẫu được chọn
 - p : xác suất mong muốn để đạt được đường tốt
 - Xác suất p để **một** mẫu của n điểm đúng là : w^n
 - Xác suất để **tất cả** k mẫu sai là: : $(1 - w^n)^k$
- ⇒ Chọn **k** đủ lớn để đảm bảo xác suất $(1-p)$ đủ nhỏ

RANSAC: Tính toán k ($p=0.99$)

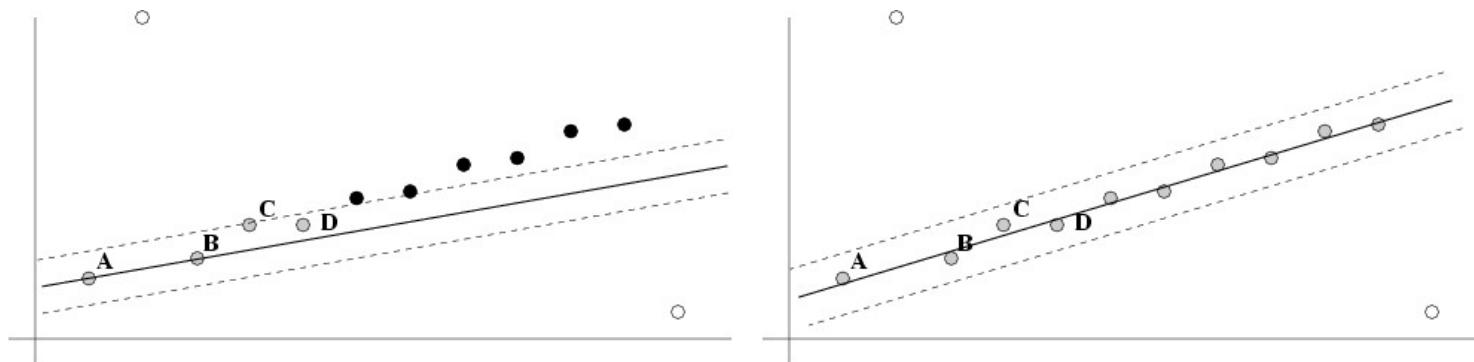
Sample size n	Proportion of outliers						
	5%	10%	20%	25%	30%	40%	50%
2	2	3	5	6	7	11	17
3	3	4	7	9	11	19	35
4	3	5	9	13	17	34	72
5	4	6	12	17	26	57	146
6	4	7	16	24	37	97	293
7	4	8	20	33	54	163	588
8	5	9	26	44	78	272	1177

p: xác suất sẽ có được mẫu tốt

Slide credit: David Lowe

Sau RANSAC

- RANSAC chia dữ liệu thành **inliers** và **outliers** and yields estimate computed from minimal set of inliers.
- Improve **this initial estimate** with estimation over all inliers (e.g. with standard least-squares minimization).
- But this may change inliers, so alternate fitting with re-classification as inlier/outlier.



Slide credit: David Lowe

RANSAC: Ưu và nhược điểm

- **Ưu:**

- Chống chịu tốt với điểm ngoại biên
- Phương thức chung phù hợp với nhiều vấn đề so khớp mô hình (nhiều tham số hơn biến đổi Hough)
- **Dễ** thực thi và tính toán tỉ lệ thất bại

- **Nhược:**

- **Chỉ xử lý được một lượng ngoại biên nhất định** mà không tăng chi phí toán quá lớn
- Nhiều vấn đề thực tế có **tỉ lệ điểm ngoại biên lớn** (đôi khi việc lựa chọn tập mẫu con có thể hỗ trợ giải quyết vấn đề)
 - Chiến lược bỏ phiếu, biến đổi Hough có thể xử lý vấn đề tỉ lệ điểm ngoại biên lớn
- Không tốt khi muốn tùy chọn các hàm so khớp khác nhau

- **Ứng dụng cụ thể**

- Ghép ảnh (như ảnh panorama hay ảnh HDR)
- **Ước lượng ma trận cơ bản** (multiple view)

References

- Lecture 3: CS231 - Juan Carlos Niebles and Ranjay Krishna, Stanford Vision and Learning Lab
- Vision par Ordinateur, Alain Boucher, IFI



DAI HOC
BACH KHOA
25
YEARS ANNIVERSARY
SOICT

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

**Thank you for
your attention!**

