Edge Detection

Edge detection

Canny edge detector

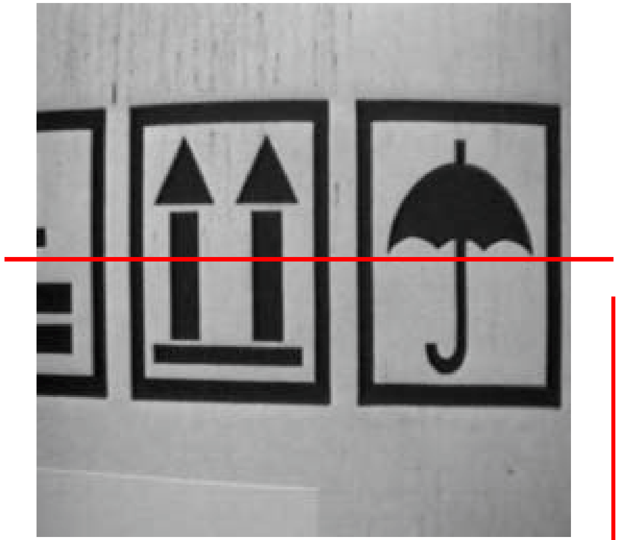
Susan edge detector

Lecturer: Sang Hwa Lee

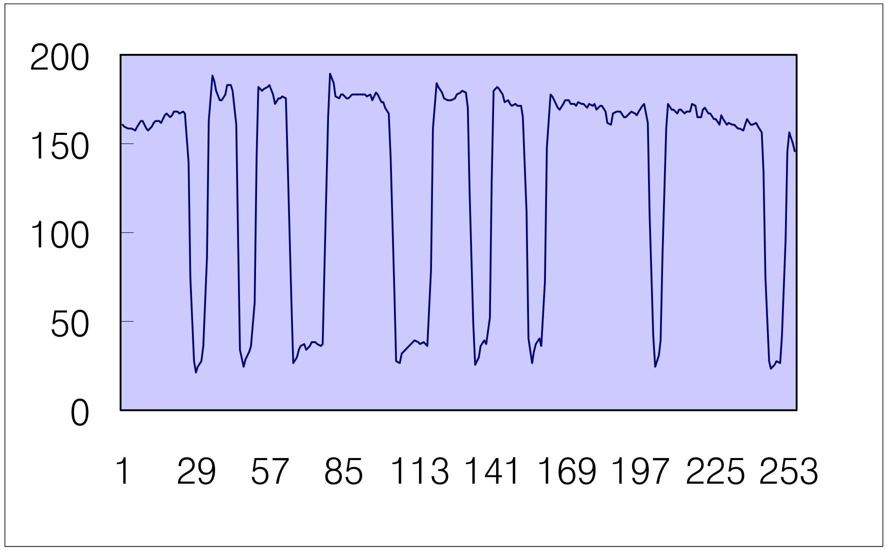
Edge (?)

- A discontinuity/abrupt change in the intensity or color
- Object boundaries in the images
- Means to recognize objects in the images
- Usually represented as binary images (Edge map)

Ideal Edge Function



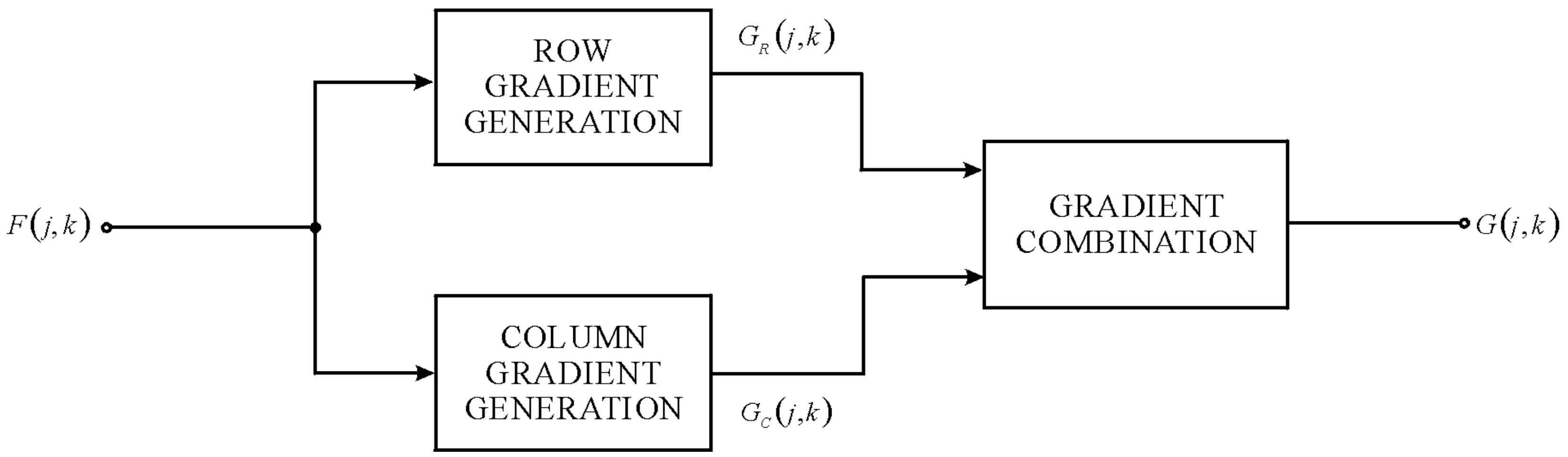
The profile of an ideal step change in image intensity



Gradient (1): First-order derivative

- Detecting significant local changes
- 2-D equivalent of the first derivative

$$G(\ f(x,y)) = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix}$$



Gradient (2) : First-order derivative

- Magnitude

$$1.\ G(f(x,y)) = \sqrt{G_x^{\,2} + G_y^{\,2}}$$

$$2.\ G(f(x,y)) = |G_x| + |G_y|$$

$$3.\ G(f(x,y)) = \max(|G_x|, |G_y|)$$

- Direction

$$\alpha(x,y) = \tan^{-1}\frac{G_y}{G_x}$$

Mask Operators (1)

3x3 edge detection

A_0	A_1	A_2
A_7	$f(j,k)$	A_3
A_6	A_5	A_4

$$G_x(j,k) = \frac{1}{K+2}\left[\left(A_2 + KA_3 + A_4\right) - \left(A_0 + KA_7 + A_6\right)\right]$$

$$G_y(j,k) = \frac{1}{K+2}\left[\left(A_0 + KA_1 + A_2\right) - \left(A_6 + KA_5 + A_4\right)\right]$$

K=1

-1	-1	-1
1	1	1

-1		1
-1		1
-1		1

; Prewitt operator

K=2

-1	-2	-1
1	2	1

-1		1
-2		2
-1		1

; Sobel operator

K= $\sqrt{2}$

-1	$-\sqrt{2}$	-1
1	$\sqrt{2}$	1

-1		-1
$-\sqrt{2}$		$\sqrt{2}$
1		1

; Chen-Frei operator

Mask Operators (2)

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

G_x

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

G_y

Original image

Result of Sobel operator (threshold = 70)

Mask Operators (3)

Original image

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

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

G

G > threshold

Result of Sobel operator (threshold = 70)

Mask Operators (4)

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

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

Original image

G_x

G_y

Result of Sobel operator (threshold = 130)

Mask Operators (5)

Original image

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

G

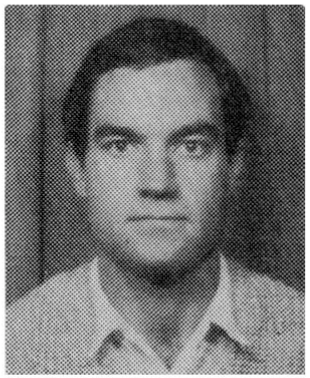
G > threshold

Result of Sobel operator (threshold = 130)

IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. PAMI-8, NO. 6, NOVEMBER 1986 679

A Computational Approach to Edge Detection

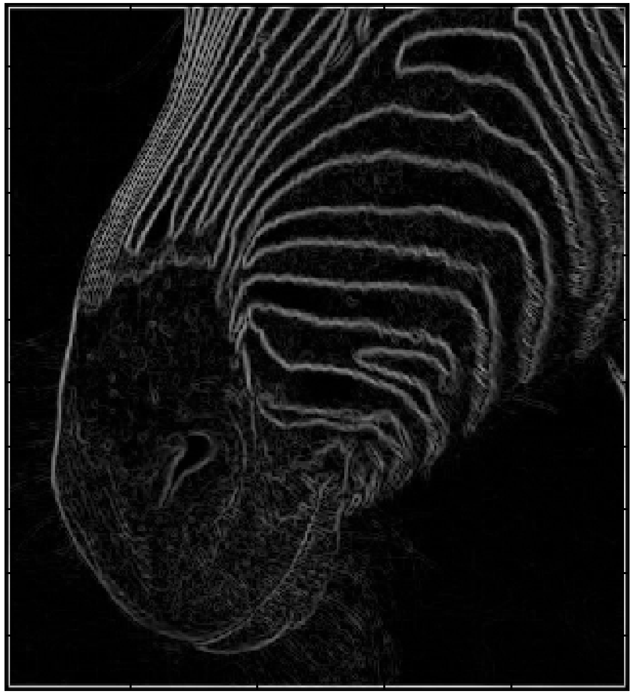
JOHN CANNY, MEMBER, IEEE



John Canny (S'81–M'82) was born in Adelaide, Australia, in 1958. He received the B.Sc. degree in computer science and the B.E. degree from Adelaide University in 1980 and 1981, respectively, and the S.M. degree from the Massachusetts Institute of Technology, Cambridge, in 1983.

He is with the Artificial Intelligence Laboratory, M.I.T. His research interests include low-level vision, model-based vision, motion planning for robots, and computer algebra.

Mr. Canny is a student member of the Association for Computing Machinery.



Gradient magnitudes at scale 1

Gradient magnitudes at scale 2

Issues:

1) The gradient magnitude at different scales is different; which should we choose?
2) The gradient magnitude is large along thick trail; how do we identify the significant points?
3) How do we link the relevant points up into curves?
4) Noise.

Canny Edge Detector (1)

1. Smoothed images (Gaussian filter)

$$S(i,j) = G(i,j;\sigma) * f(i,j)$$

2. First-difference approximation

$$G_x(i,j) \cong (\, S(i,j+1) - S(i,j) + S(i+1,j+1) - S(i+1,j)\,)\,/\,2$$

$$G_y(i,j) \cong (\, S(i,j) - S(i+1,j) + S(i,j+1) - S(i+1,j+1)\,)\,/\,2$$

Canny Edge Detector (2)

3. Compute gradient magnitude and orientation

magnitude $$M(i,j) = \sqrt{G_x(i,j)^2 + G_y(i,j)^2}$$

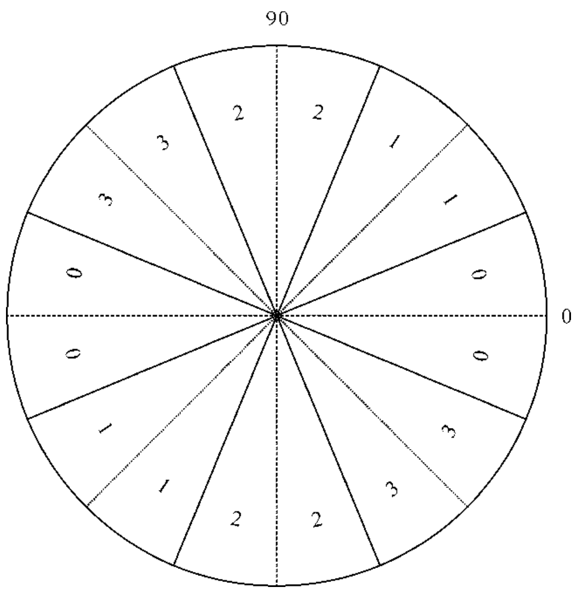
angle $$\theta(i,j) = \tan^{-1}\frac{G_x(i,j)}{G_y(i,j)}$$

$\longrightarrow$ partition into four sectors

(horizontal, vertical, diagonal, anti-diagonal)

Canny Edge Detector (3)

4. Nonmaximal suppression (1)



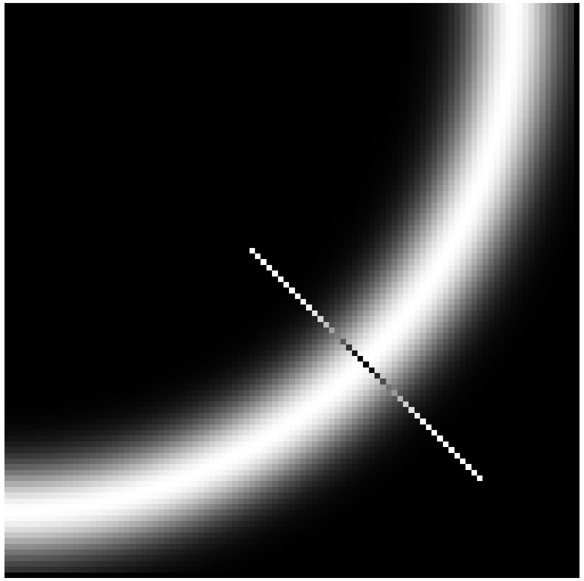
- to identify edges, the broad ridges in the magnitude array $M(i, j)$ must be thinned so that only the magnitude at the points of greatest local change remain, resulting in thinned edges
- reduces the angle $\theta(i, j)$ of the gradient into one of the four sectors

Canny Edge Detector (4)

4. Nonmaximal suppression (2)

- if the magnitude array value $M(i, j)$ at the center of the 3×3 neighborhood is not greater than both of the neighbor magnitudes along the gradient line, then $M(i,j)$ is set to zero

3	2	1
0		0
1	2	3



Canny Edge Detector (5)

5. Thresholding and connected edge preserving

- apply a threshold to the nonmaximal-suppressed gradient magnitude to reduce the number of false edge fragments

- double thresholding algorithm:

  τ_1 and τ_2, with $\tau_2 \cong 2\tau_1$, to produce two thresholded edge images $T_1(i, j)$ and $T_2(i, j)$

  τ_2 : fewer false edges, but may have gaps in the contours

Use the results with τ_1

Canny Edge Detector (6)

Example of Non-maximal suppression

courtesy of G. Loy

Original image

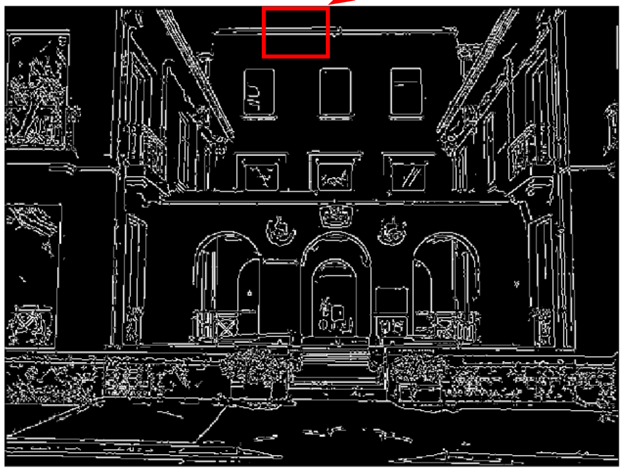
Gradient magnitude

Non-maxima suppressed

Canny Edge Detector (7)

Original image

gap is gone



Strong + connected weak edges

Strong edges only

Weak edges

courtesy of G. Loy

Canny Edge Detector (8)

$\sigma = 2$

Original image

Smoothed image

Canny Edge Detector (9)

Enhancement

Nonmaximal suppression

Canny Edge Detector (10)

$\sigma = 1$

Thresholding (high=130, low=20)

Thresholding (high=100, low=15)

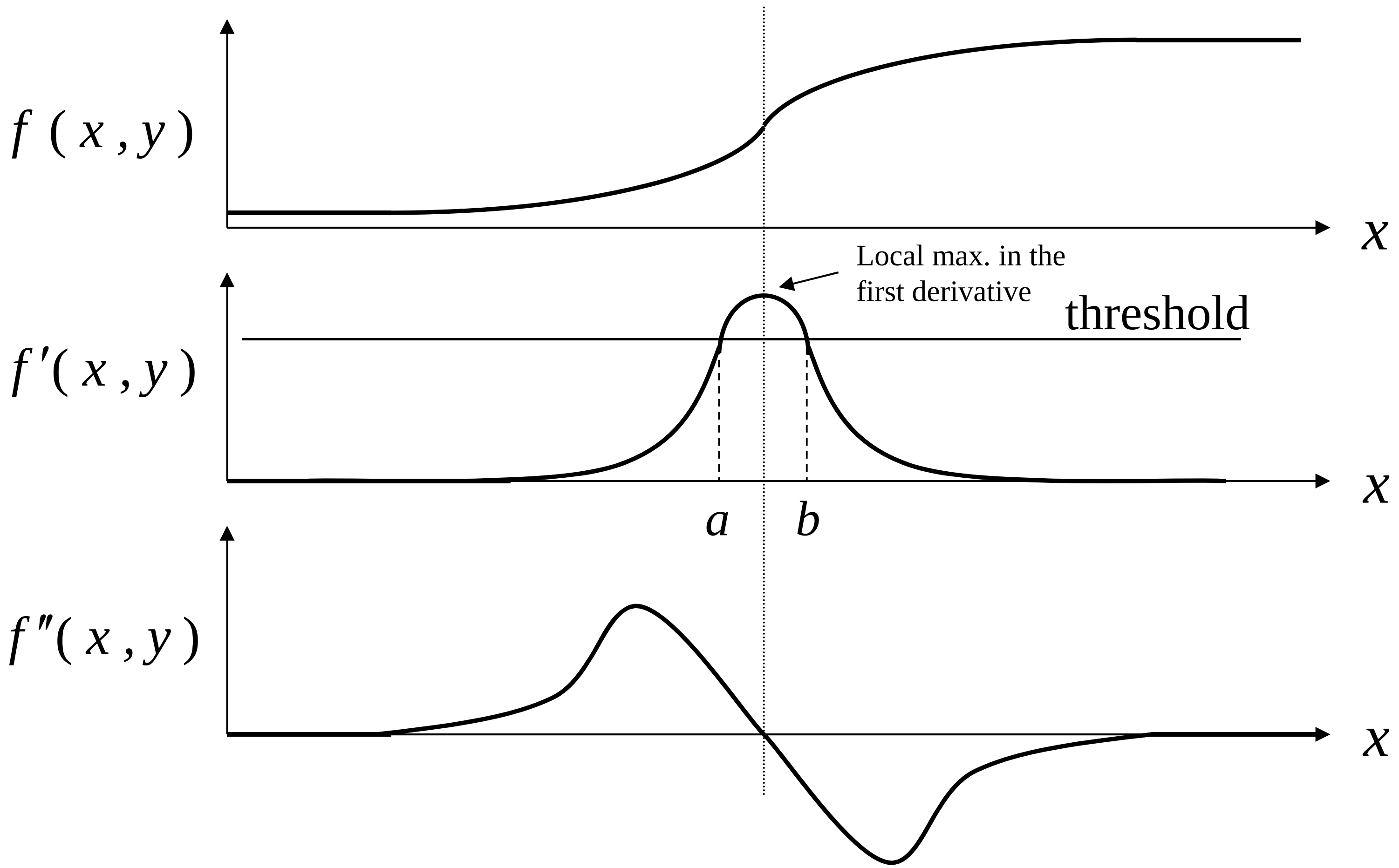
Canny Edge Detector (11)

$\sigma = 2$

Thresholding (high=130, low=20)

Thresholding (high=100, low=15)

Second Derivative Operators (1)



Second Derivative Operators (2)

- Thresholding gradient images
  - too many edge points $\longrightarrow$ finding only the points that have local maxima in gradient values and considering them as edge points

Second Derivative Operators (3)

- Peak in the first derivative

  = zero crossing in the second derivative (subpixel accuracy)

- Laplacian operator
  - not used frequently in machine vision (sensitive to noise)

    Gaussian filtering+second derivative

Second Derivative Operators (4)

$$\text{Laplacian}: \nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$$

$$G(x,y) = -\nabla^2 \left\{ f(x,y) \right\}$$

$\Rightarrow \nabla^2 \left\{ f(x,y) \right\}$ is zero if $f(x,y)$ is constrint or changing linearly in amplitude

$$\frac{\partial^2 f}{\partial x^2} \cong \frac{\partial G_x}{\partial x} = \frac{\partial \left[f(i,j) - f(i,j-1) \right]}{\partial x}$$

$$= f(i,j+1) - 2f(i,j) + f(i,j-1)$$

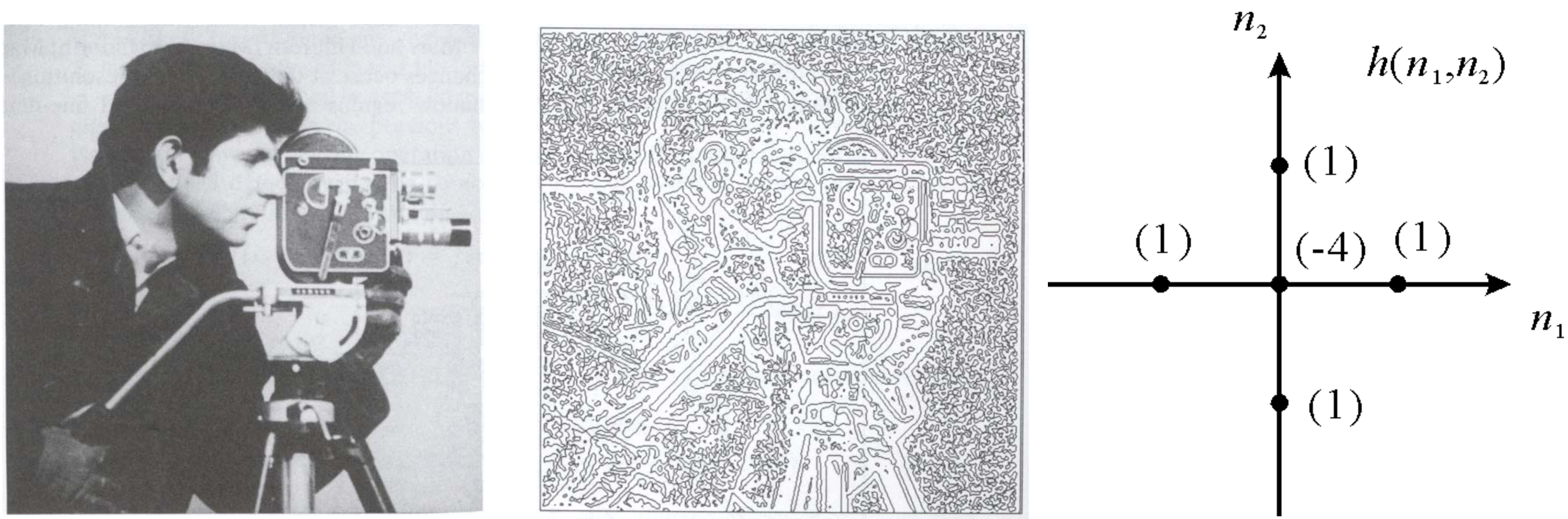
$$\frac{\partial^2 f}{\partial y^2} = f(i+1,j) - 2f(i,j) + f(i-1,j)$$

$$H_1 = \begin{array}{|c|c|c|} \hline 0 & -1 & 0 \\ \hline -1 & 4 & -1 \\ \hline 0 & -1 & 0 \\ \hline \end{array} \qquad H_2 = \begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline -1 & 8 & -1 \\ \hline -1 & -1 & -1 \\ \hline \end{array}$$

4 – neighbor Laplacia   8-neighbor Laplacian

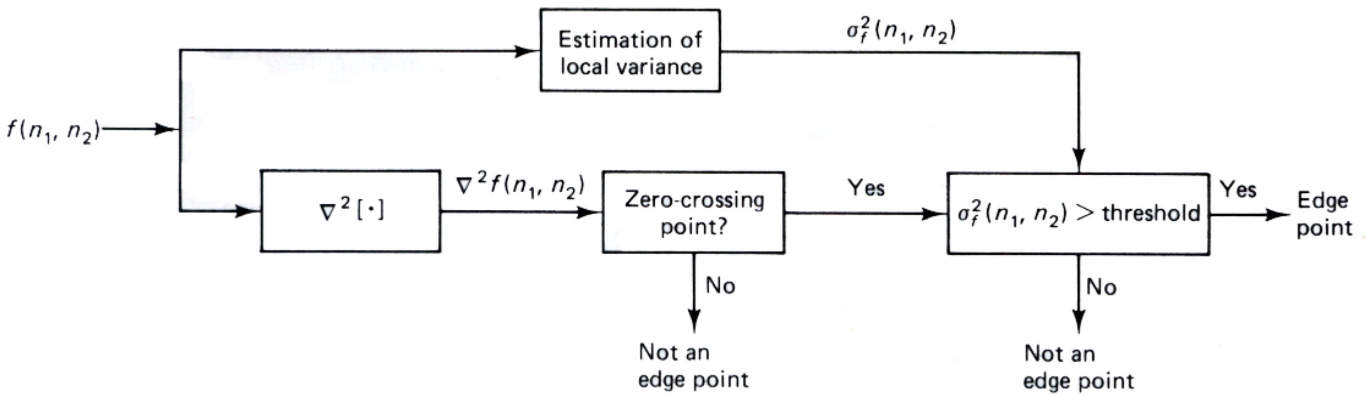
Second Derivative Operators (5)

- Laplacian –based techniques generate many “false” edges, which typically appear in the regions when the local variance of the image is small
  ⇨sensitive to the input noise



(a) Image, (b) Result of Laplacian operator in (c).

Second Derivative Operators (6)



(a)

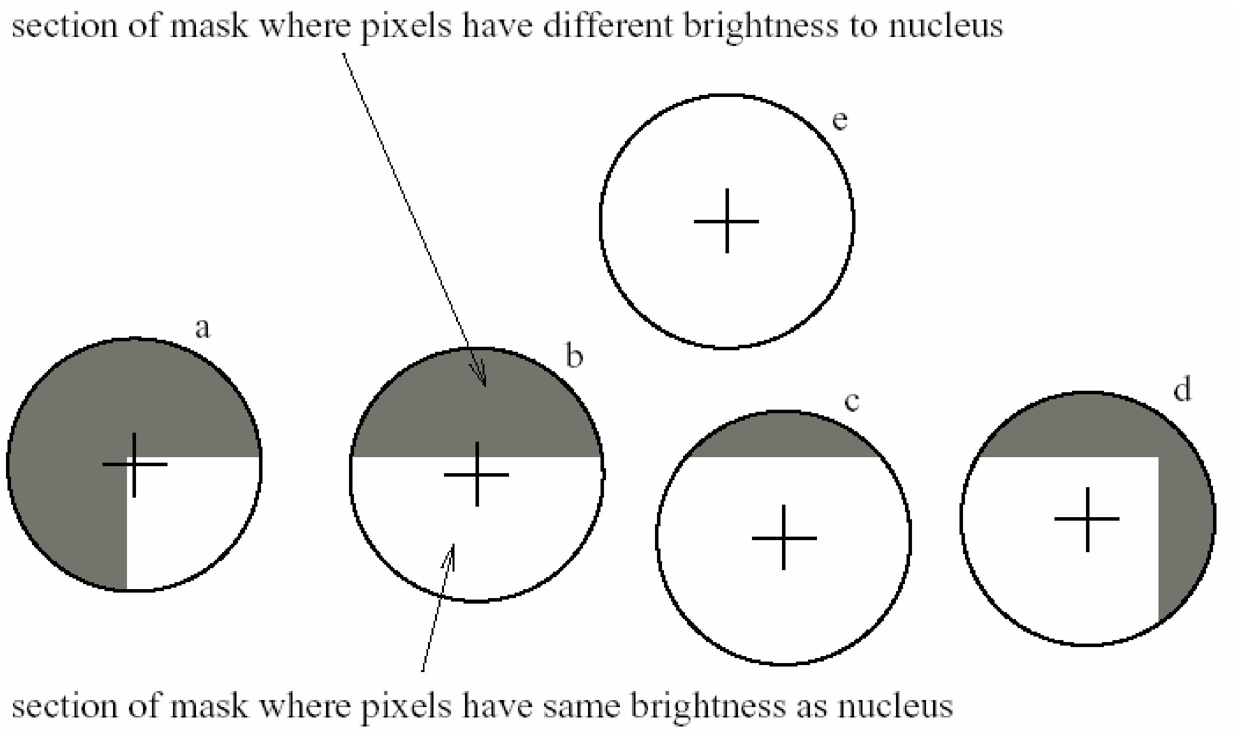
(b)

(c)

(a) Laplacian-based edge detection system that does not produce many false edge contours.
(b) Image of 512x 512 pixels.
(c) Edge map obtained by applying the system in (a) to the image in (b)

SUSAN Edge Detector (1)

- Smallest Univalue Segment Assimilating Nucleus



SUSAN Edge Detector (2)

- Smallest Univalue Segment Assimilating Nucleus

$$c(r,r_0)=\begin{cases}1 & if\ |I(r)-I(r_0)|\le t\\ 0 & if\ |I(r)-I(r_0)|> t\end{cases}$$

$$c(r,r_0)=e^{-(\frac{I(r)-I(r_0)}{t})^6}$$

$$n(r_0)=\sum_r c(r,r_0)$$

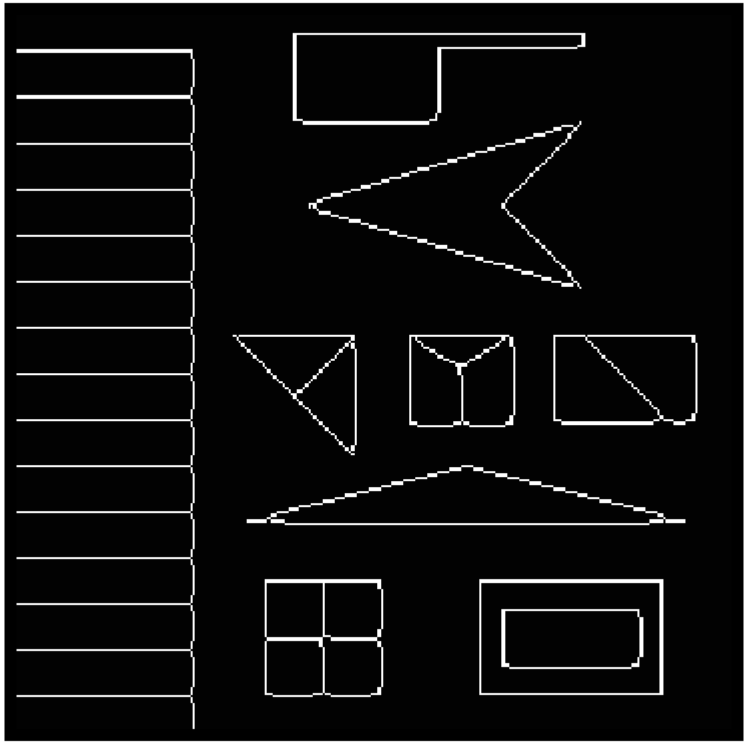
$$R(r)=\begin{cases}g-n(r) & if\ \ n(r)<g\\ 0 & otherwise\end{cases}$$

SUSAN Edge Detector (3)

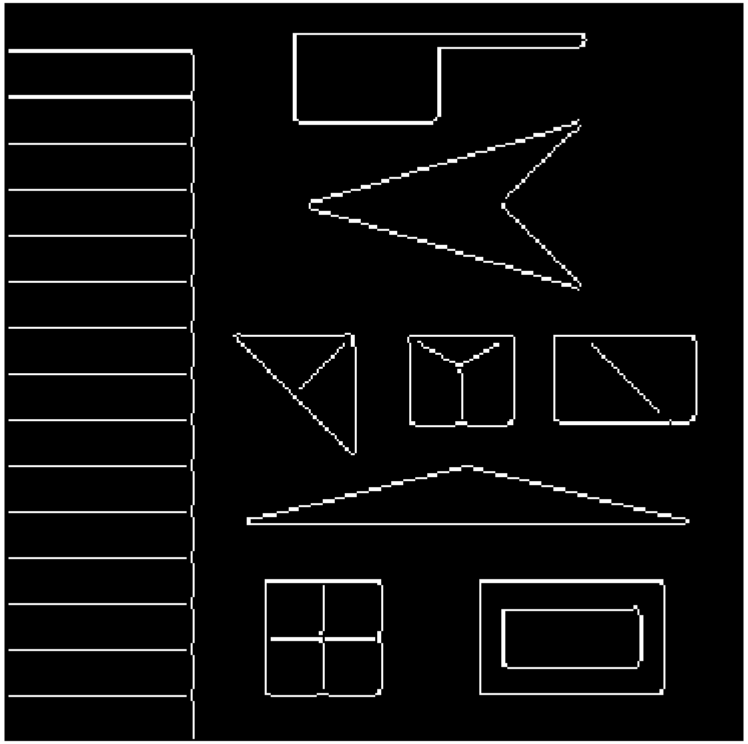
- Test image

SUSAN Edge Detector (4)

- Some Results



SUSAN edge detector



Canny edge detector