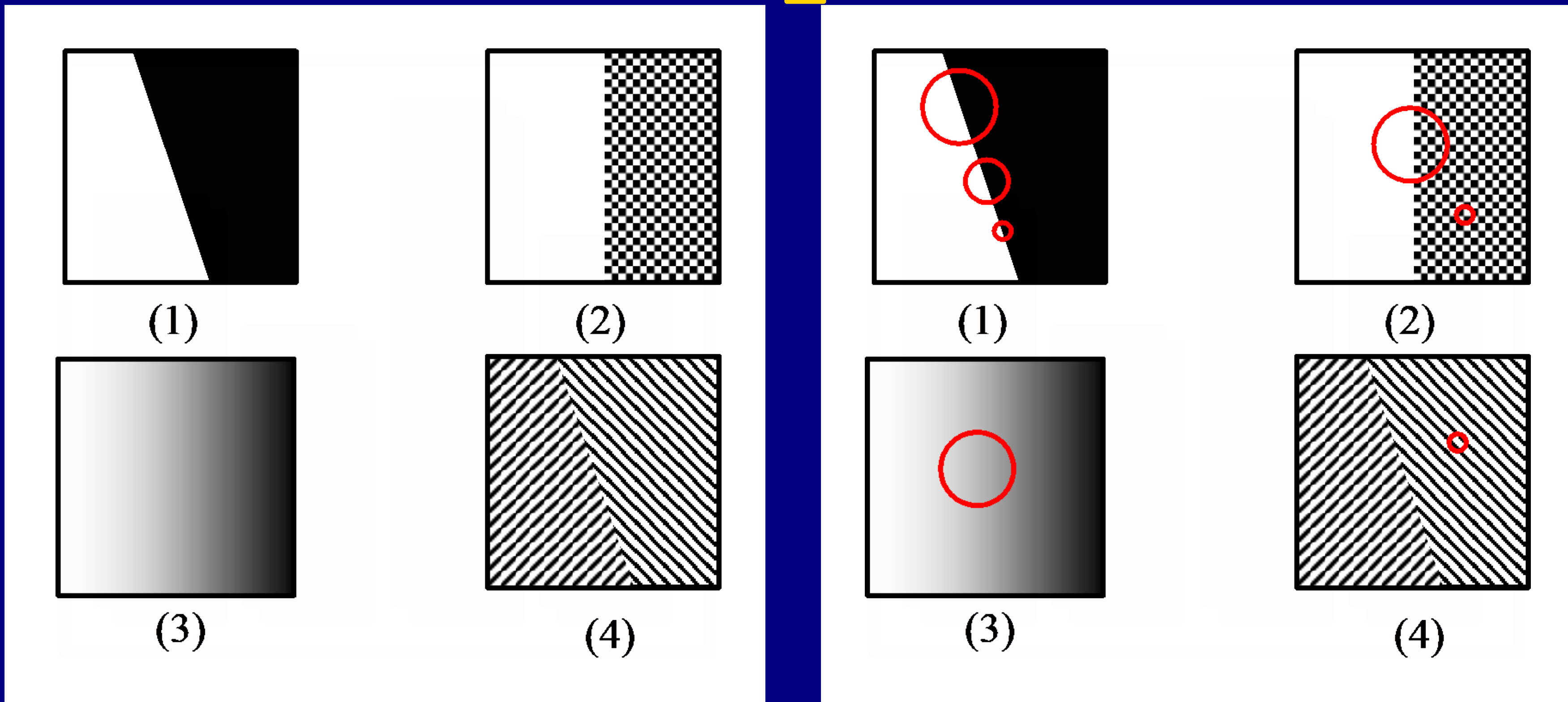
Image Processing and Visual Communications

Edge Detection

Zhou Wang

Dept. of Electrical and Computer Engineering
University of Waterloo

What is an Edge?



From Prof. Al Bovik

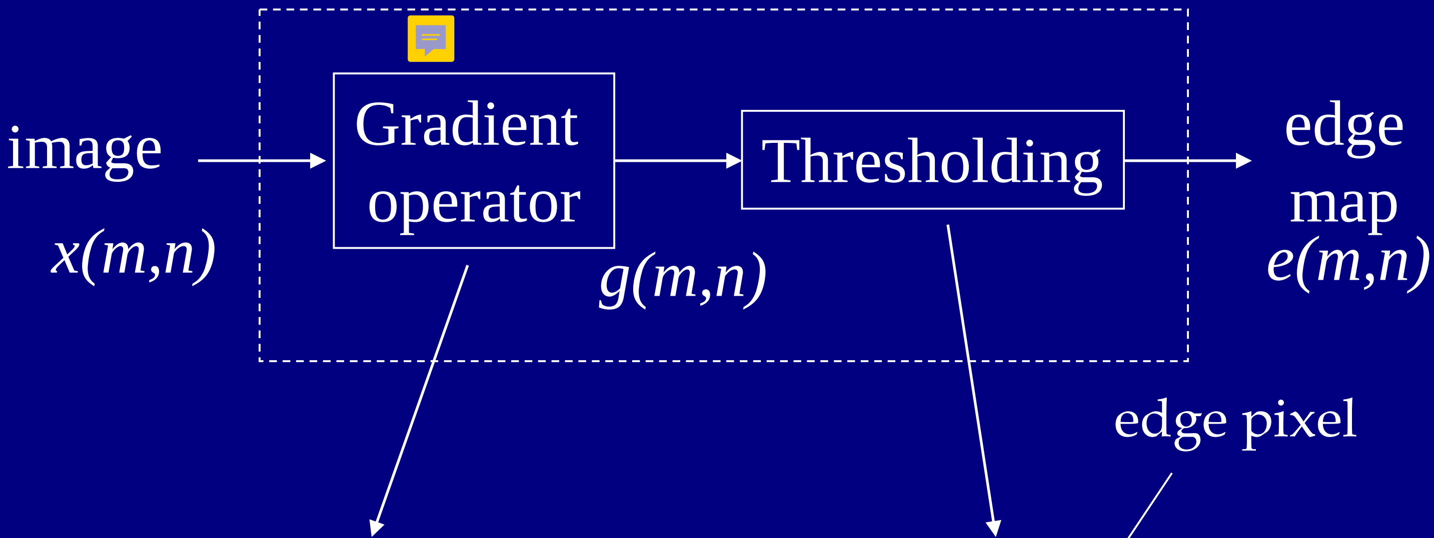
Edges: Sudden changes in **certain image properties** that extend along a contour

Edge map of an image is very **informative**

In general, the perception of edges changes with **scale**

Gradient-Based Methods

- **Motivation**
  - Detect sudden changes in image intensity
  - Gradient: sensitive to intensity changes



From Prof. Xin Li

Gradient: $$\nabla f = \left[\frac{\partial f}{\partial x} \quad \frac{\partial f}{\partial y} \right]^T$$

edge pixel threshold

$$e(m,n) = \begin{cases} 1 & |g(m,n)| > T \\ 0 & otherwise \end{cases}$$

non edge pixel

Gradient-Based Methods

- **Gradient Operators**

$$\begin{array}{ccc} & g_1 & g_2 \\ \text{Robert:} & \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} & \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \\ \text{Prewitt:} & \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} & \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \\ \text{Sobel:} & \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} & \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \end{array}$$

Local gradient vector:

$$g(m,n) = \begin{bmatrix} g_1(m,n) \\ g_2(m,n) \end{bmatrix}$$

Gradient magnitude:

$$|g(m,n)| = \sqrt{g_1^2(m,n) + g_2^2(m,n)}$$

Approximation:

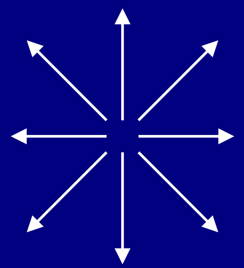
$$|g_1(m,n)| + |g_2(m,n)|$$

Gradient-Based Methods

- **Generalization: Compass Operator**

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

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



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

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

maximal magnitude:

$$q(m,n) = \max_k \{ | g_k(m,n) | \}$$

$\downarrow$

$\boxed{\text{Thresholding}}$

$\downarrow$

edge image

Gradient-Based Methods

original image

$\left| g_1(m,n) \right|$

$\left| g_2(m,n) \right|$

$\left| g_1(m,n) \right| + \left| g_2(m,n) \right|$

From [Gonzalez & Woods]

Examples

A 9x9 original image is given by

9	9	9	9	9	9	9	2	2
9	8	9	9	9	9	2	2	2
9	9	9	9	9	9	3	2	2
9	9	9	9	9	2	2	2	2
7	9	9	9	9	2	2	2	2
9	9	9	9	2	2	2	2	2
9	9	9	9	2	2	2	4	2
9	9	9	2	2	2	2	2	2
9	9	2	2	2	2	1	2	2

1) Use Robert gradient operator to find its edges

Use $|g(m,n)| = \sqrt{g_1^2(m,n) + g_2^2(m,n)}$ to estimate the gradient magnitude, and use $T = 5$ as the threshold for edge detection

2) Use Sobel gradient operator to find its edges

Use $|g(m,n)| = |g_1(m,n)| + |g_2(m,n)|$ to estimate the gradient magnitude, and use $T = 20$ as the threshold for edge detection

Examples

1) Use Robert gradient operator

g_1

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

original image

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

g_2

$g_1(m,n)$

1	0	0	0	0	7	7	0
0	-1	0	0	0	6	0	0
0	0	0	0	7	7	1	0
0	0	0	0	7	0	0	0
-2	0	0	7	7	0	0	0
0	0	0	7	0	0	-2	0
0	0	7	7	0	0	0	2
0	7	7	0	0	1	0	0

0	1	0	0	0	0	0	0
-1	0	0	0	0	-7	-1	0
0	0	0	0	0	1	0	0
2	0	0	0	-7	0	0	0
0	0	0	0	0	0	0	0
0	0	0	-7	0	0	0	-2
0	0	0	0	0	0	2	0
0	0	0	0	0	0	1	0

$g_2(m,n)$

$$|g(m,n)| = \sqrt{g_1^2(m,n) + g_2^2(m,n)}$$

1	1	0	0	0	7	7	0
1	1	0	0	0	9.2	1	0
0	0	0	0	7	7.1	1	0
2	0	0	0	9.9	0	0	0
2	0	0	7	7	0	0	0
0	0	0	9.9	0	0	2	2
0	0	7	7	0	0	2	2
0	7	7	0	0	1	1	0

edge map $e(m,n)$

0	0	0	0	0	1	1	0
0	0	0	0	0	1	0	0
0	0	0	0	1	1	0	0
0	0	0	0	1	0	0	0
0	0	0	1	1	0	0	0
0	0	0	1	0	0	0	0
0	0	1	1	0	0	0	0
0	1	1	0	0	0	0	0

thresholding $T = 5$

Note: 8x8 images (because of boundary problems)

Examples

2) Use Sobel gradient operator

original image

g_1

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

$g_1(m,n)$

0	2	0	0	-20	-28	-8
0	1	0	-7	-26	-21	-2
2	0	0	-21	-27	-7	-1
4	0	-7	-28	-21	0	0
2	0	-21	-28	-7	2	0
0	-7	-28	-21	0	4	0
-7	-21	-21	-7	-1	2	1

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

g_2

0	0	0	0	-6	-12	-6
2	1	0	-7	-14	-7	0
-2	0	0	-7	-15	-9	-1
0	0	-7	-14	-7	0	0
2	0	-7	-14	-7	2	4
0	-7	-14	-7	0	0	0
-7	-21	-21	-7	-1	-4	-5

$g_2(m,n)$

$$|g(m,n)| = |g_1(m,n)| + |g_2(m,n)|$$

0	2	0	0	26	40	14
2	2	0	14	40	28	2
4	0	0	28	42	16	2
4	0	14	42	28	0	0
4	0	28	42	14	4	4
0	14	42	28	0	4	0
14	42	42	14	2	6	6

thresholding $T = 20$

edge map $e(m,n)$

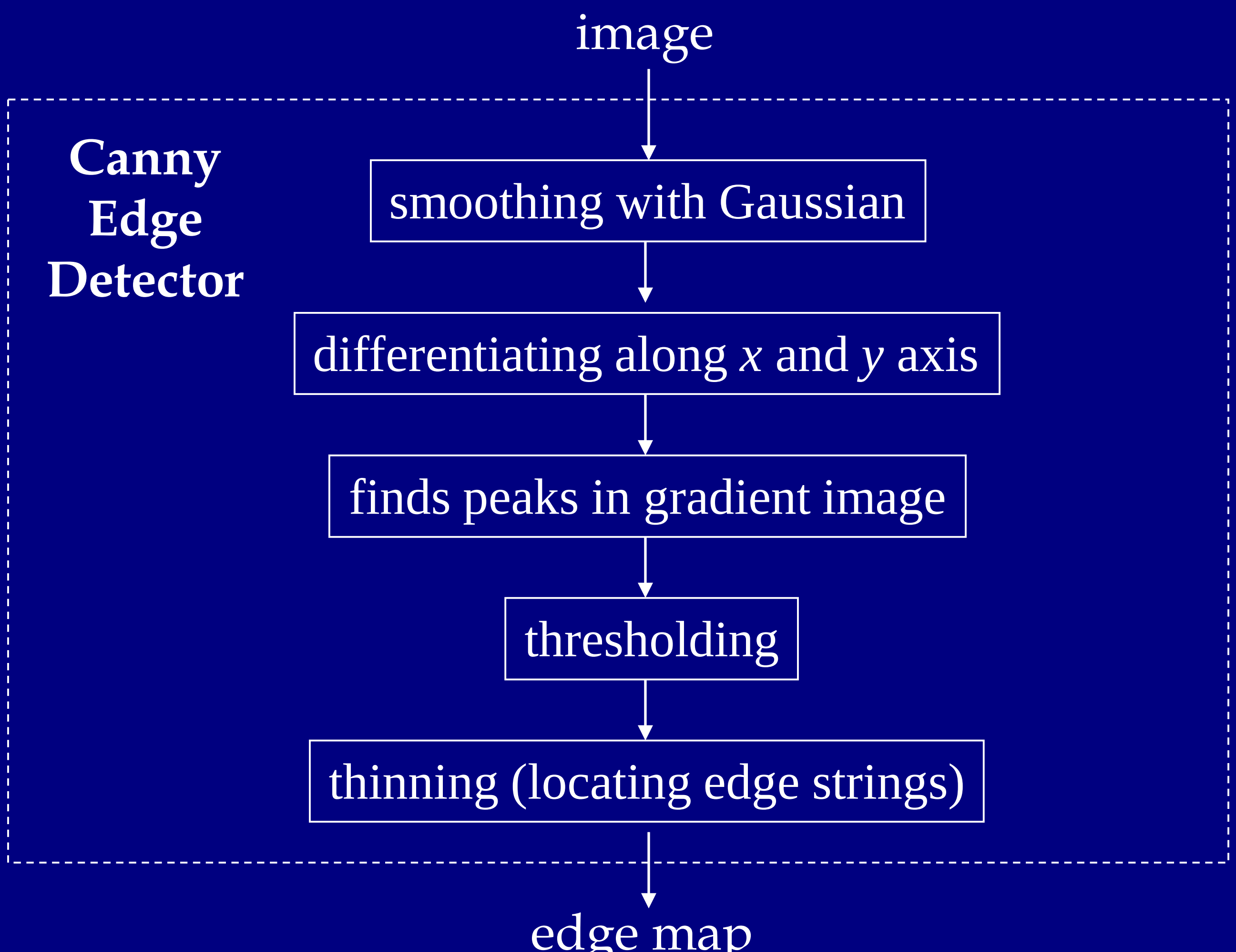
0	0	0	0	1	1	0
0	0	0	0	1	1	0
0	0	0	1	1	0	0
0	0	0	1	1	0	0
0	0	1	1	0	0	0
0	0	1	1	0	0	0
0	1	1	0	0	0	0

Note: 7x7 images (because of boundary problems)

Gradient-Based Methods: Canny Edge Detector

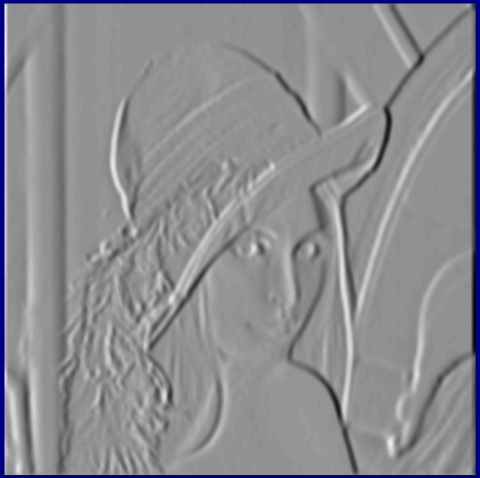
- **Problems with standard gradient edge detectors**

  noise-sensitive; thick edge; broken edge contour; hard to pick threshold



Gradient-Based Methods: Canny Edge Detector

original image



horizontally filtered

vertically filtered

gradient magnitude

after thresholding

after thinning

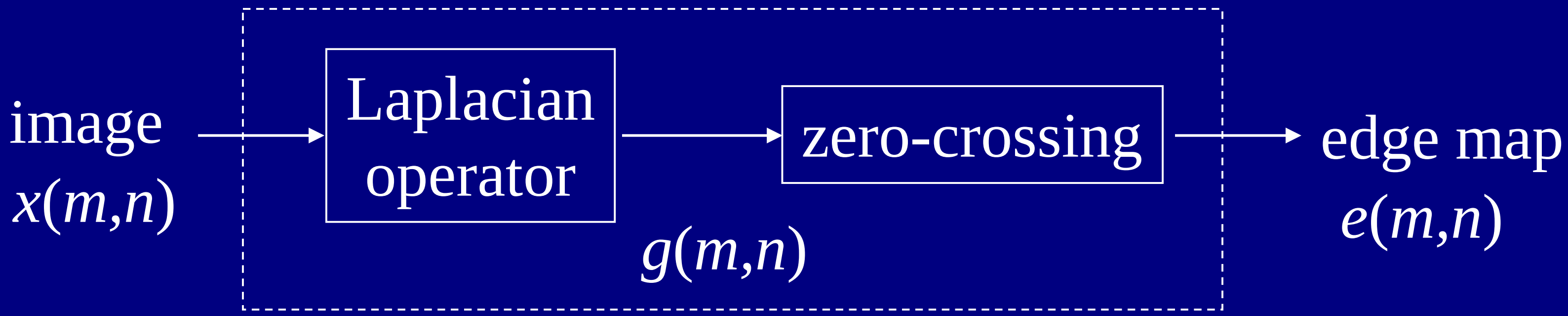
From Prof. Xin Li

Laplacian-Based Methods

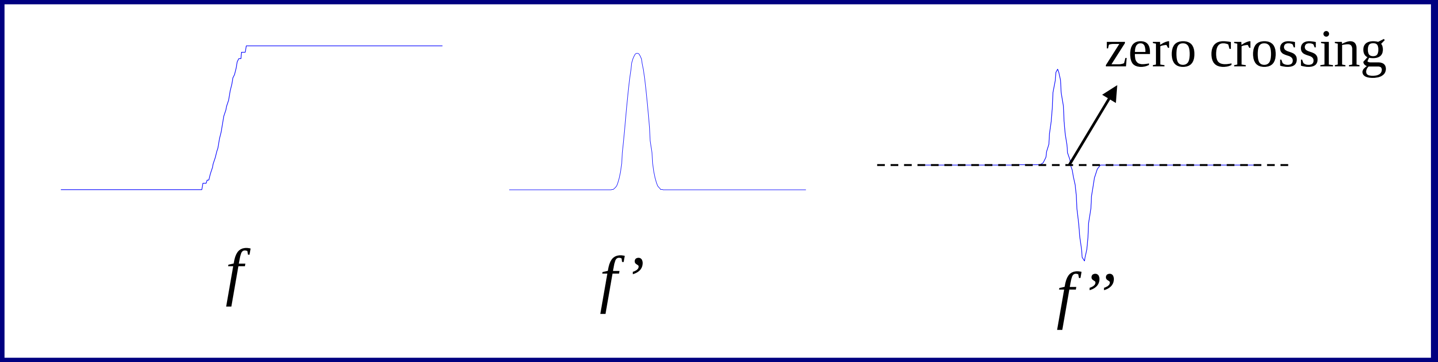
- **Laplacian Operator + Zero-Crossing**
  - Advantages: thin edges; closed edge contours

Laplacian:

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



1D illustration:



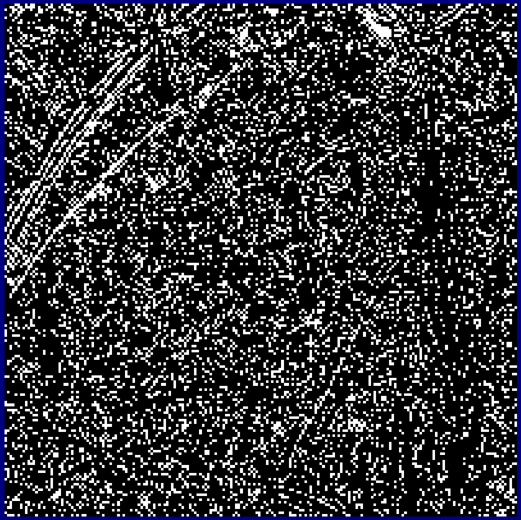
From Prof. Xin Li

Laplacian-Based Methods

Typical Laplacian operators:

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \qquad \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Laplacian + zero-crossing →



From Prof. Xin Li

- **Laplacian of Gaussian (LoG)**
  - Problem with standard Laplacian methods: Noise-sensitive
  - Idea of LoG: Smoothing with Guassian, followed by Laplacian

Laplacian-Based Methods

- **Laplacian of Gaussian (LoG)**

3D view

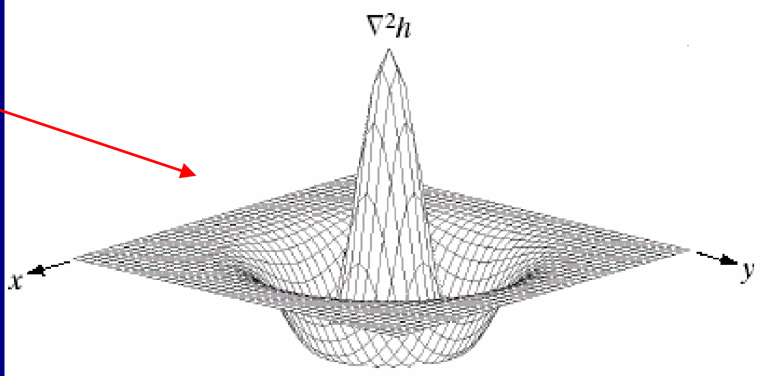
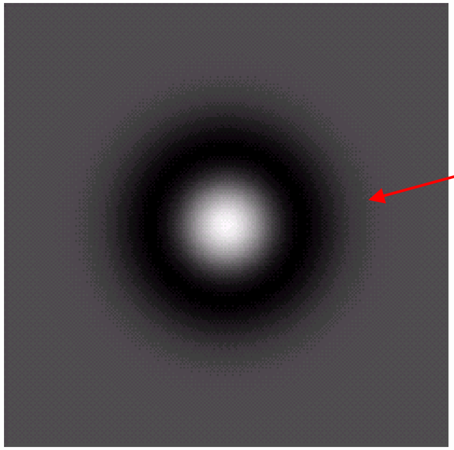
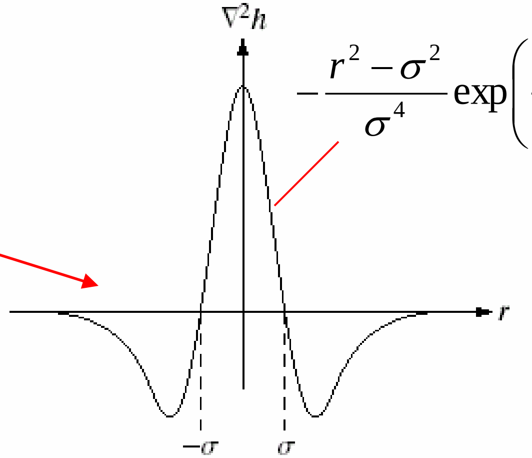


image view



cross section view



$$-\frac{r^2-\sigma^2}{\sigma^4}\exp\left(-\frac{r^2}{2\sigma^2}\right)$$

5x5 window approximation

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

From [Gonzalez & Woods]

Laplacian-Based Methods