# Edge Detection

Prof. M. Elif Karslıgil

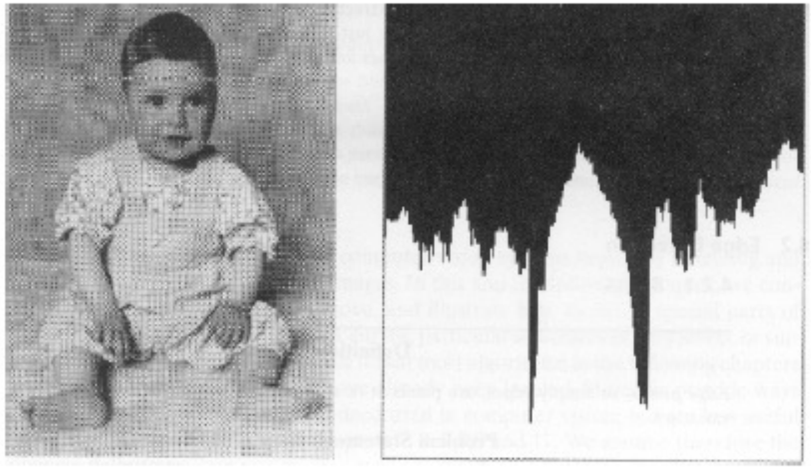
Yildiz Technical University

Computer Engineering Department

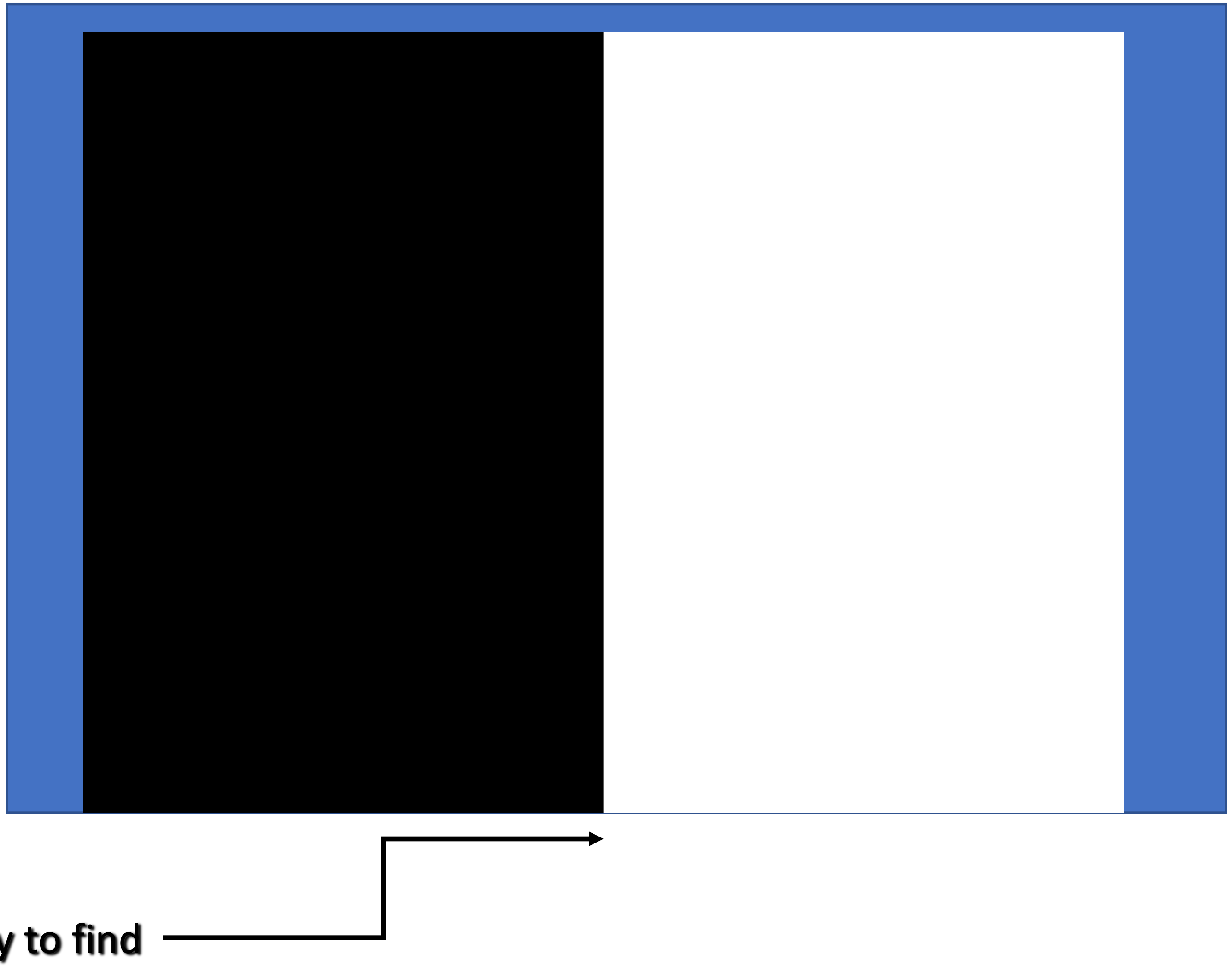
# What is an Edge ?

- Edges are significant local changes of intensity in an image.

- Edges typically occur on the boundary between two different regions in an image.



# What is an Edge?

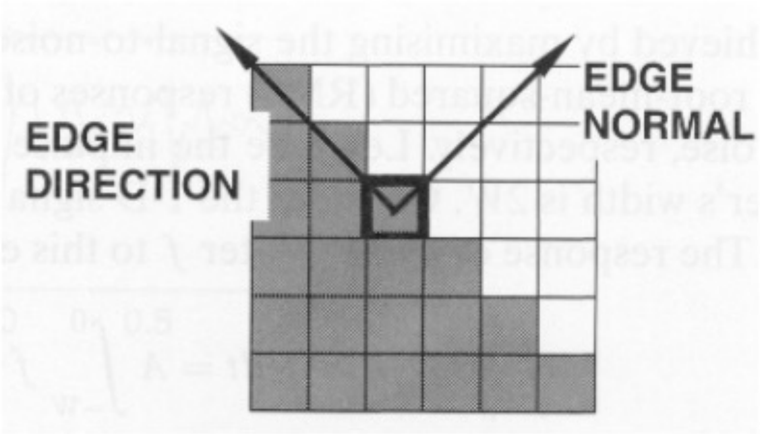


# Edge Detection

- **Goal:** Identify sudden changes (discontinuities) in an image
  - Intuitively, most semantic and shape information from the image can be encoded in the edges
  - More compact than pixels

# Edge Descriptors

Edge normal: unit vector in the direction of maximum intensity change.
Edge direction: unit vector to perpendicular to the edge normal.
Edge position or center: the image position at which the edge is located.
Edge strength: related to the local image contrast along the normal.

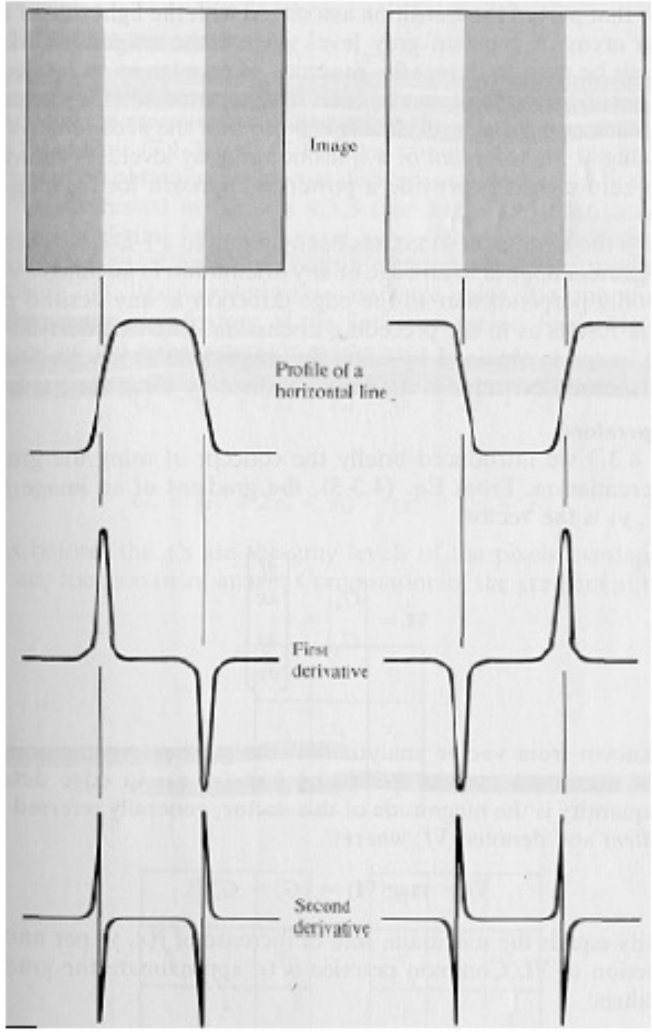


# Edge Detection using Derivatives

- Points which lie on an edge can be detected by:

    (1) detecting local maxima or minima of the first derivative

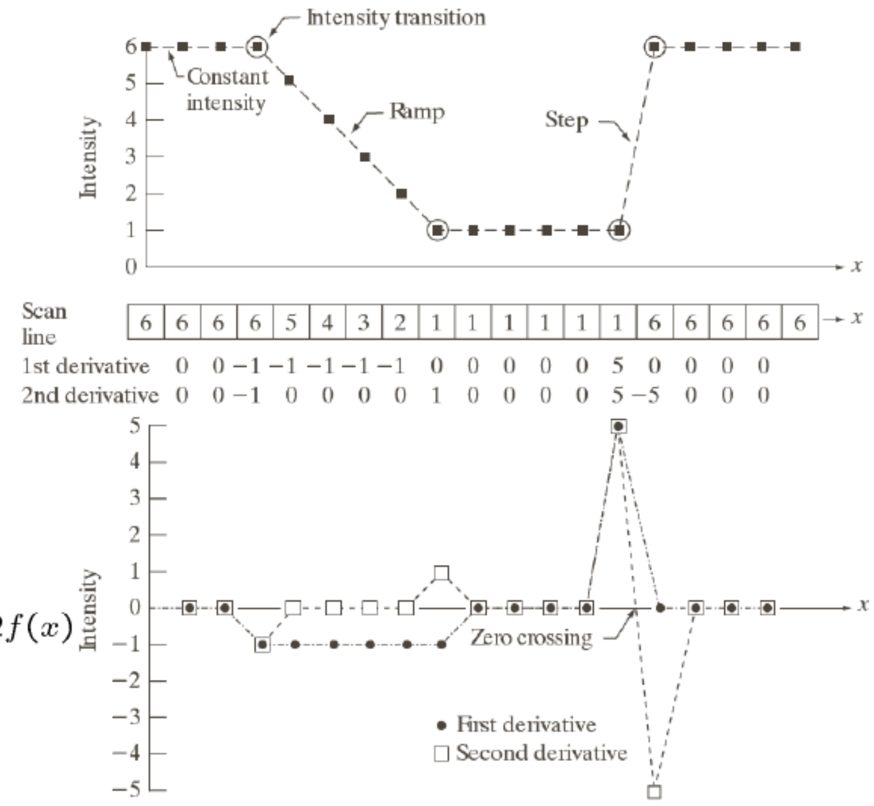
    (2) detecting the zero-crossing of the second derivative



# Image Derivative and Sharpening

$$
\begin{aligned}
f'(x) &= \frac{\partial f}{\partial x} \\
&= f(x+1) - f(x)
\end{aligned}
$$

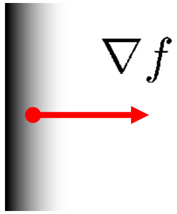
$$
\begin{aligned}
f''(x) &= \frac{\partial^2 f}{\partial x^2} \\
&= \frac{\partial f}{\partial x}(f'(x) - f'(x-1)) \\
&= f(x+1) + f(x-1) - 2f(x)
\end{aligned}
$$



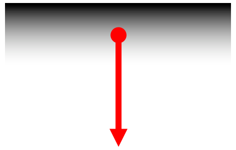
# Difference operators for 2D

- Contrast in the 2D picture function $f(x, y)$ can occur in any direction.
- From calculus, we know that the maximum change occurs along the direction of the *gradient*.
- The gradient of an image $f(x, y)$ at location $(x, y)$ is defined as the vector

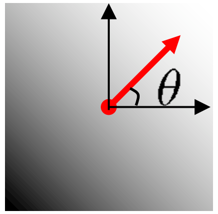
$$\nabla f = \left[ \frac{\partial f}{\partial x} \;\; \frac{\partial f}{\partial y} \right]^T .$$



$\nabla f = \left[\frac{\partial f}{\partial x}, 0\right]$



$\nabla f = \left[0, \frac{\partial f}{\partial y}\right]$




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

# Difference operators for 2D

- The *magnitude of the gradient*

$$|\nabla f| = \left( \left( \frac{\partial f}{\partial x} \right)^2 + \left( \frac{\partial f}{\partial y} \right)^2 \right)^{1/2}$$

gives the maximum rate of increase of $f(x, y)$ per unit distance in the direction of $\nabla f$.

- The *direction of the gradient*

$$\measuredangle(\nabla f) = \tan^{-1} \left( \frac{\partial f}{\partial y} \Big/ \frac{\partial f}{\partial x} \right)$$

represents the direction of this change with respect to the $x$-axis.

# The Prewitt Edge Detector

- Consider the arrangement of pixels about the pixel $(i, j)$:

$$
\begin{array}{ccc}
a_0 & a_1 & a_2 \\
a_7 & [i, j] & a_3 \\
a_6 & a_5 & a_4
\end{array}
$$

- The partial derivatives can be computed by:

$$
\begin{aligned}
M_x &= (a_2 + ca_3 + a_4) - (a_0 + ca_7 + a_6) \\
M_y &= (a_6 + ca_5 + a_4) - (a_0 + ca_1 + a_2)
\end{aligned}
$$

- The constant $c$ implies the emphasis given to pixels closer to the center of the mask.

- Setting $c = 1$, we get the Prewitt operator:

$$
M_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \qquad M_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}
$$

(*note*: $M_x$ and $M_y$ are approximations at $(i, j)$)

# The Sobel Edge Detector

- Setting $c = 2$, we get the Sobel operator:

$$M_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad M_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

(*note*: $M_x$ and $M_y$ are approximations at $(i, j)$)

# Difference operators for 2D

| | | |
|---|---|---|
| $z_1$ | $z_2$ | $z_3$ |
| $z_4$ | $z_5$ | $z_6$ |
| $z_7$ | $z_8$ | $z_9$ |

| | |
|---|---|
| −1 | 0 |
| 0 | 1 |

| | |
|---|---|
| 0 | −1 |
| 1 | 0 |

Roberts

| | | |
|---|---|---|
| −1 | −1 | −1 |
| 0 | 0 | 0 |
| 1 | 1 | 1 |

| | | |
|---|---|---|
| −1 | 0 | 1 |
| −1 | 0 | 1 |
| −1 | 0 | 1 |

Prewitt

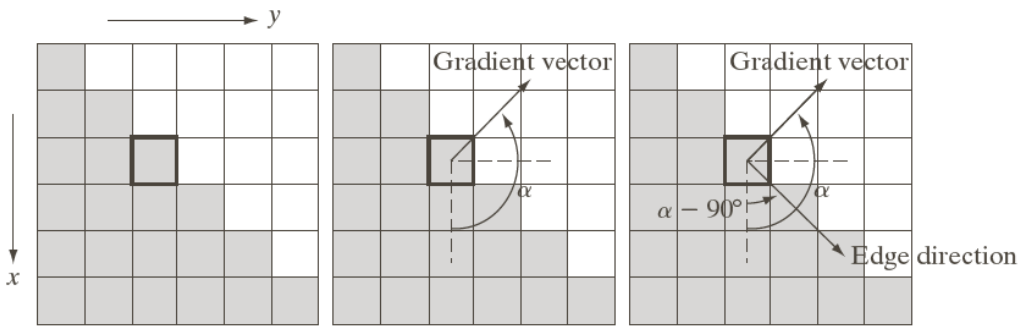
| | | |
|---|---|---|
| −1 | −2 | −1 |
| 0 | 0 | 0 |
| 1 | 2 | 1 |

| | | |
|---|---|---|
| −1 | 0 | 1 |
| −2 | 0 | 2 |
| −1 | 0 | 1 |

Sobel

a
b c
d e
f g

**FIGURE 10.14** A 3 × 3 region of an image (the $z$'s are intensity values) and various masks used to compute the gradient at the point labeled $z_5$.



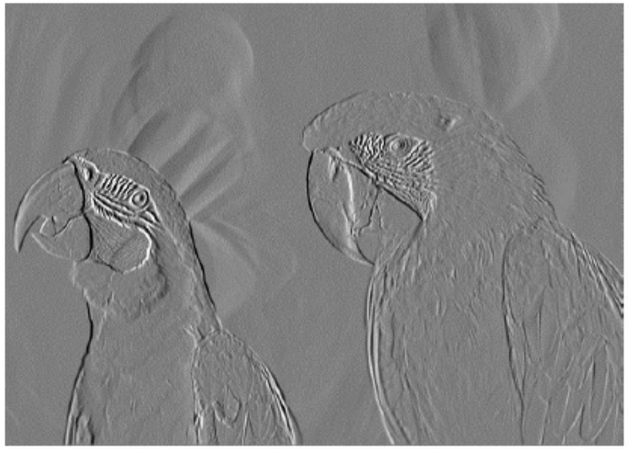

a b c

**FIGURE 10.12** Using the gradient to determine edge strength and direction at a point. Note that the edge is perpendicular to the direction of the gradient vector at the point where the gradient is computed. Each square in the figure represents one pixel.

Adapted from Gonzales and Woods

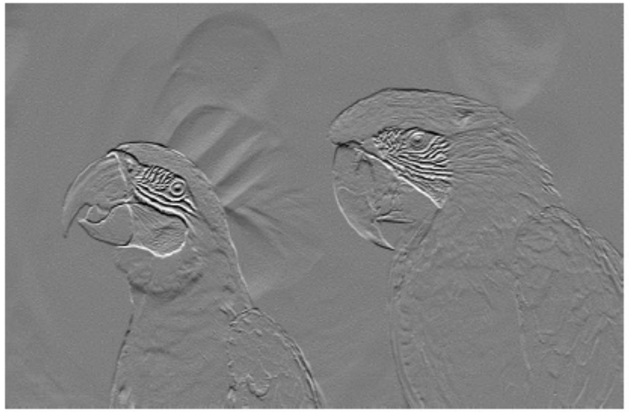
# Image Gradient

$f$



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

$||\nabla f||$



$\frac{\partial f}{\partial y}$

# Image gradient

$f$

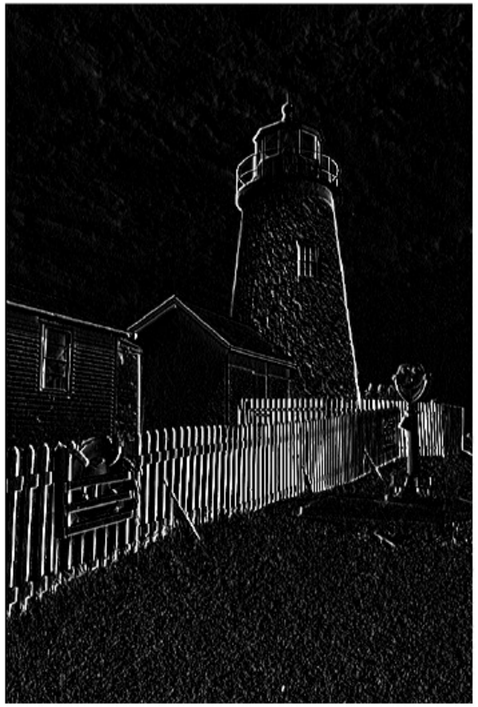
$\frac{\partial f}{\partial x}$

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

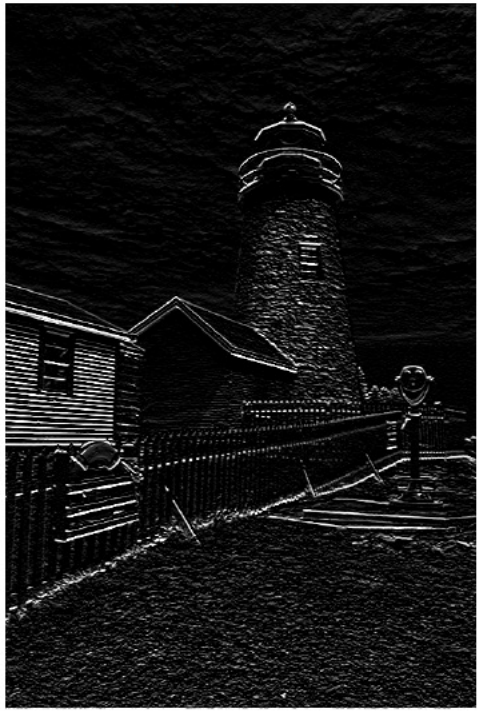
$\frac{\partial f}{\partial y}$

# Sobel Operator Example

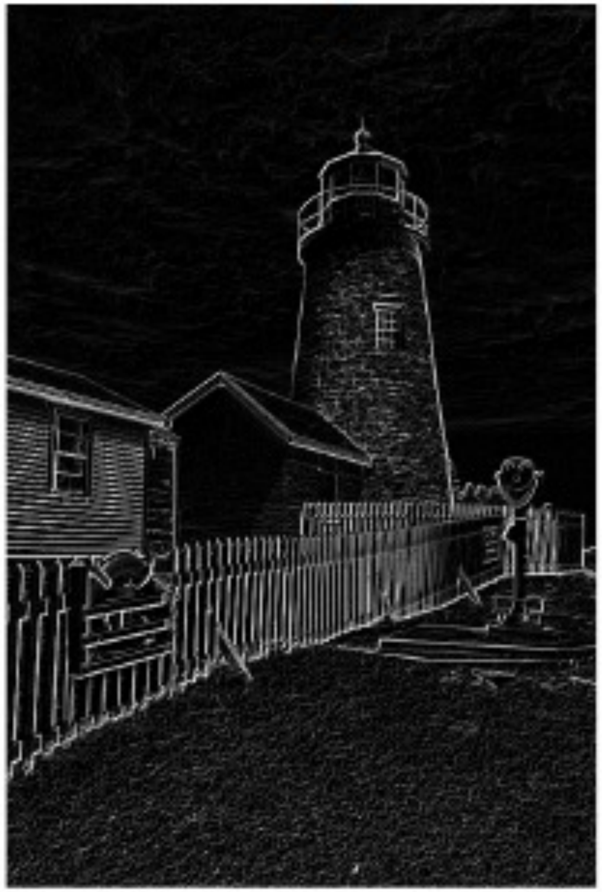
x-direction



y-direction



Gradient magnitude



# Edge Detection Examples

Roberts

Sobel

http://flickr.com/photos/reneemarie11/97326485
http://flickr.com/photos/valkyrie2112/2829322895

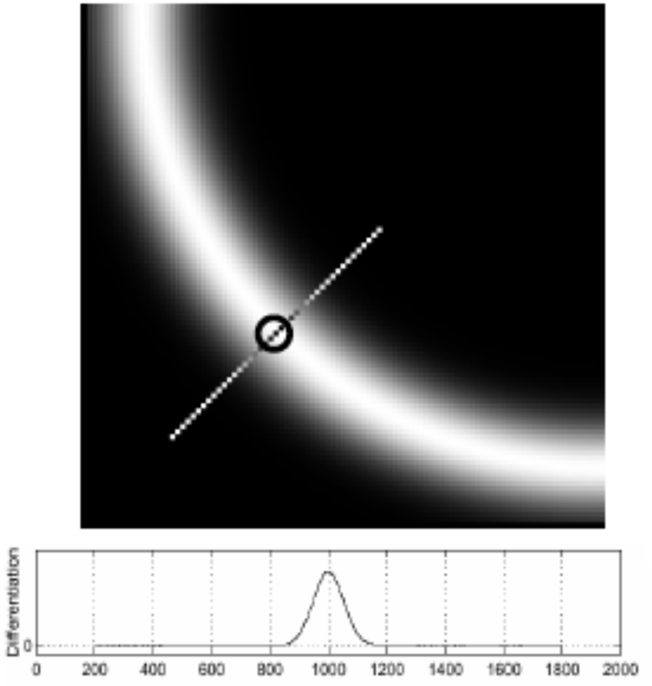
# Finding edges

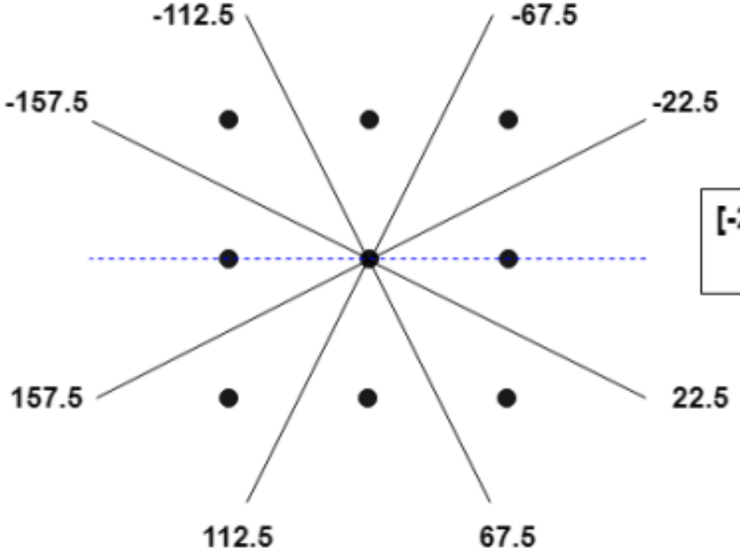

thresholding

# Canny Edge Detector

1. Filter image with derivative of Gaussian
2. Find magnitude and orientation of gradient
3. Non-maximum suppression
   - Thin multi-pixel wide “ridges” down to single pixel width
4. Linking of edge points
   - Hysteresis thresholding: use a higher threshold to start edge curves and a lower threshold to continue them

# Non maximum suppression (edge thinning)




For example

[-22.5°, 22.5°] or [-157.5°, 157.5°]
maps to 0°

- Check if pixel is local maximum along gradient direction
  - requires *interpolating* pixels p and r

For each pixel, the neighboring pixels are located in horizontal, vertical, and diagonal directions (0°, 45°, 90°, and 135°)

# Non maximum suppression

Gradient Magnitude

Non-max Suppression

# Canny Edge Detector

1. Filter image with derivative of Gaussian
2. Find magnitude and orientation of gradient
3. Non-maximum suppression
4. Linking and thresholding (hysteresis):
   - Define two thresholds: low and high
   - Use the high threshold to start edge curves and the low threshold to continue them

# Hysteresis Thresholding

Which edges are really edges and which are not ?

- Any edges with intensity greater than ‘High’ are the sure edges.
- Any edges with intensity less than ‘Low’ are sure to be non-edges.
- The edges between ‘High’ and ‘Low’ thresholds are classified as edges only if they are connected to a sure edge otherwise discarded.

# Hysteresis Thresholding

Non-max Suppression

Final Output

# Canny Edge Detector

original | Canny with $\sigma = 1$ | Canny with $\sigma = 2$

- The choice of $\sigma$ depends on desired behavior
  - large $\sigma$ detects "large-scale" edges
  - small $\sigma$ detects fine edges