ECE 4973: Lecture 12 Edge Detection

Samuel Cheng

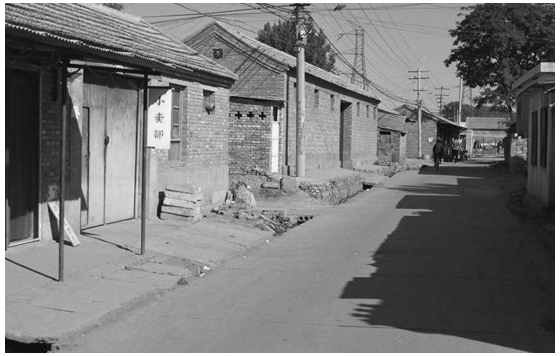
Slide credits: James Hays, Lana Lazebnik, Steve Seitz, David Forsyth, David Lowe, Fei-Fei Li, and Derek Hoiem

Think-Pair-Share

* = Convolution operator

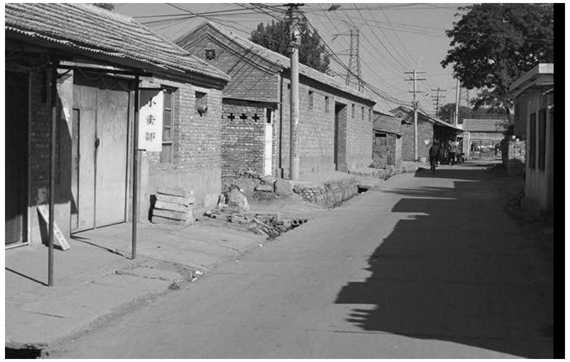
a) $\underline{G}$ = D * B
b) A = $\underline{B}$ * $\underline{C}$
c) F = D * $\underline{E}$
d) $\underline{I}$ = D * D

D

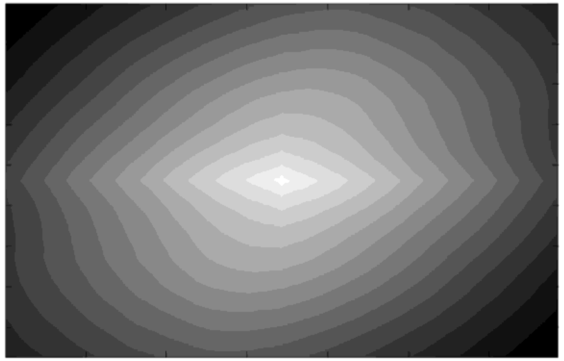


H

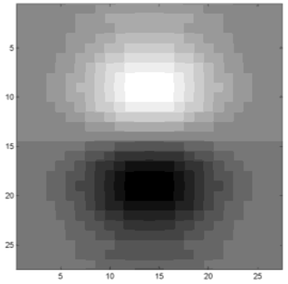
F



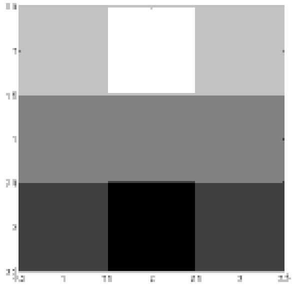
I



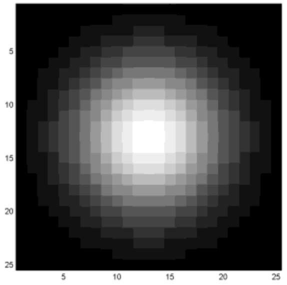
A



B

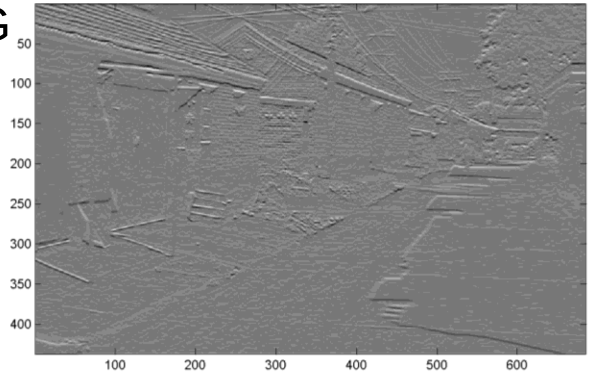


C



E

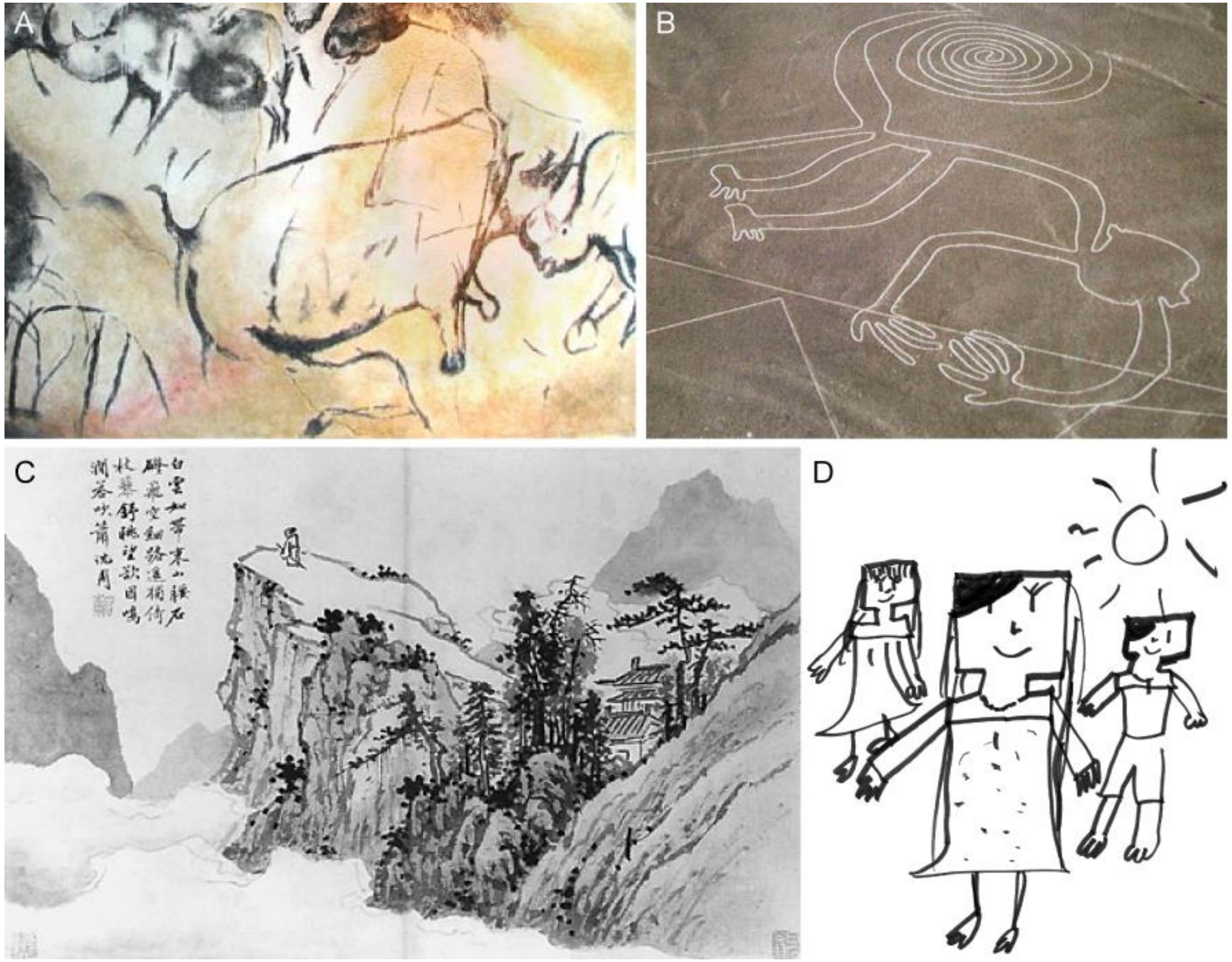
G



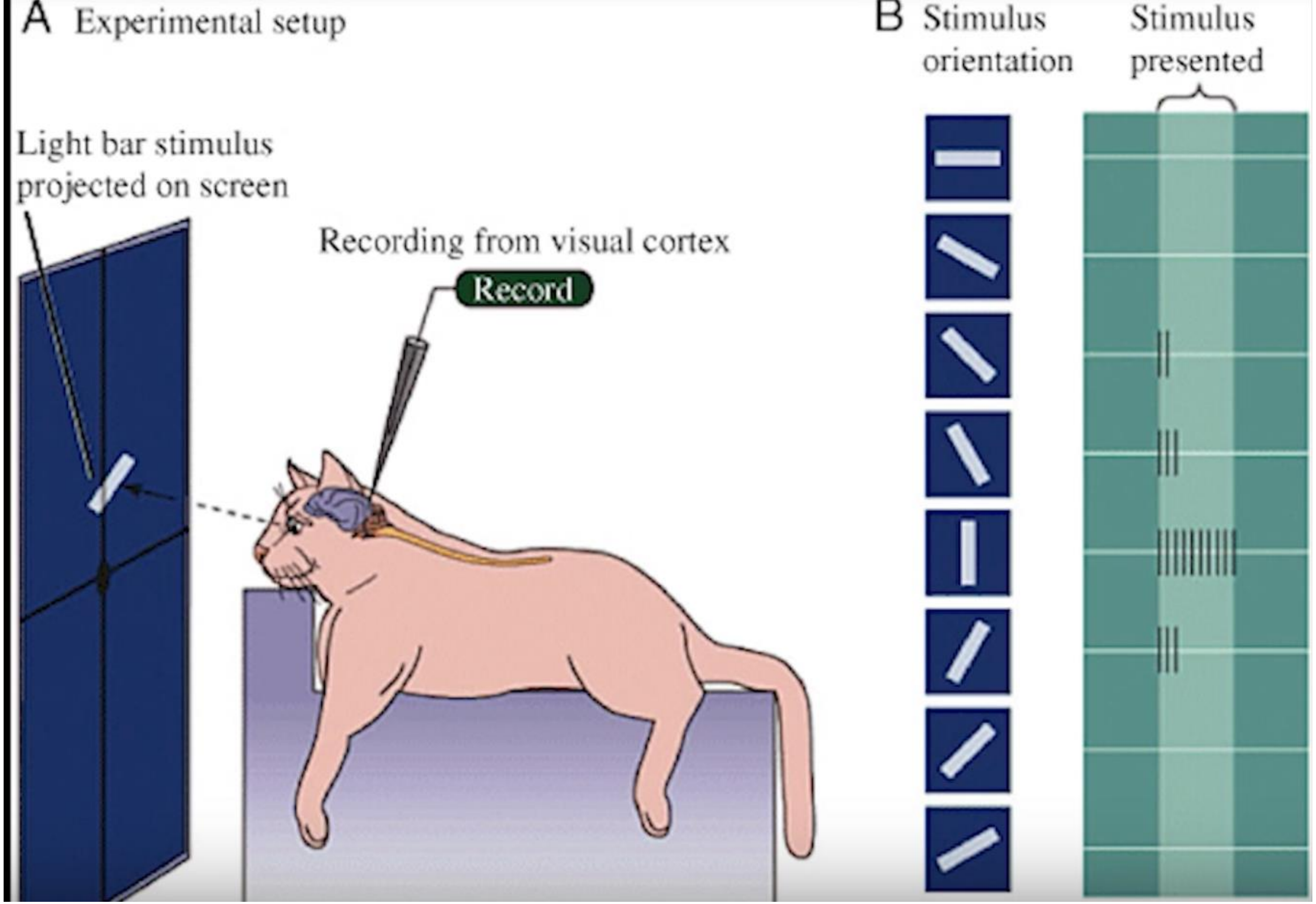
Hoiem

What we will learn today

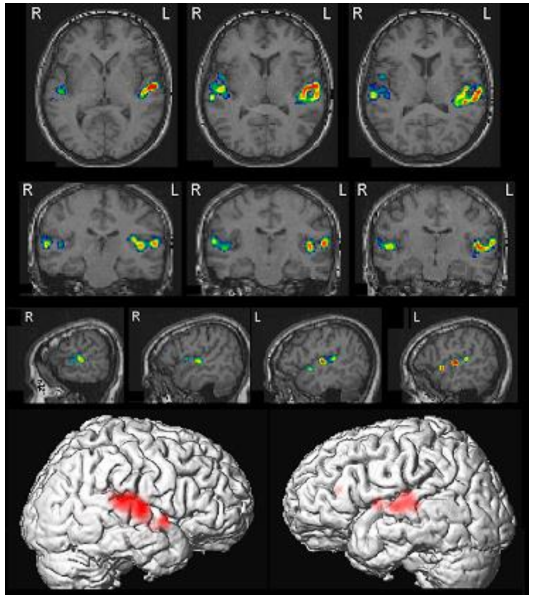
- Edge detection
- Image Gradients
- Derivative of Gaussian
- Sobel edge detector
- Canny edge detector

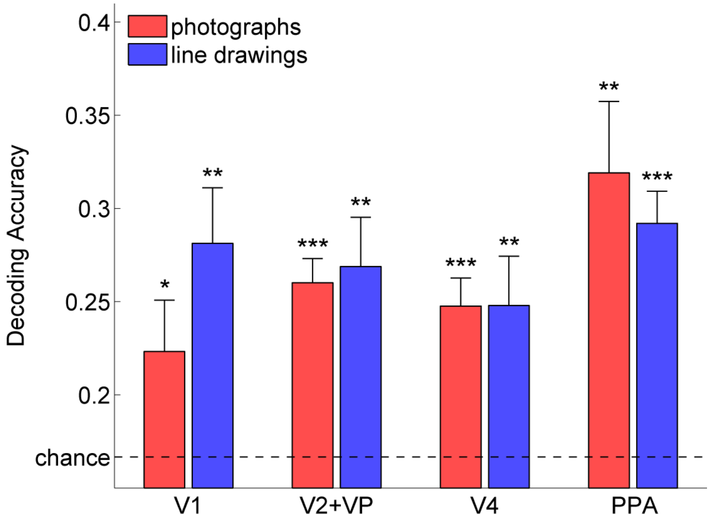


(A) Cave painting at Chauvet, France, about 30,000 B.C.;
(B) Aerial photograph of the picture of a monkey as part of the Nazca Lines geoglyphs, Peru, about 700 – 200 B.C.;
(C) Shen Zhou (1427-1509 A.D.): Poet on a mountain top, ink on paper, China;
(D) Line drawing by 7-year old I. Lleras (2010 A.D.).



Hubel & Wiesel, 1960s





Walther, Chai, Caddigan, Beck & Fei-Fei, *PNAS, 2011*

Elder – Are Edges Incomplete? 1999

Edge 'code':

- position,
- gradient magnitude,
- gradient direction,
- blur.

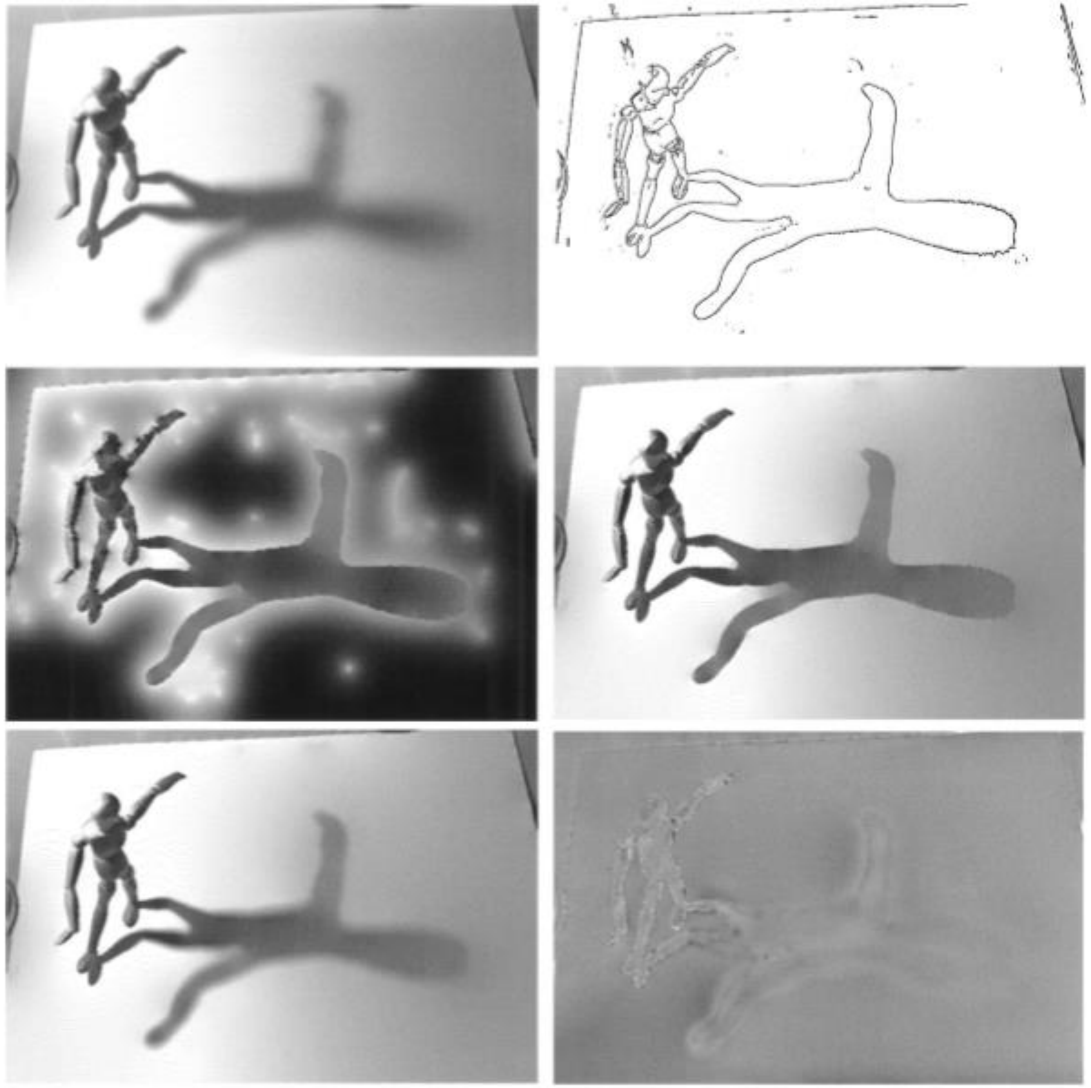
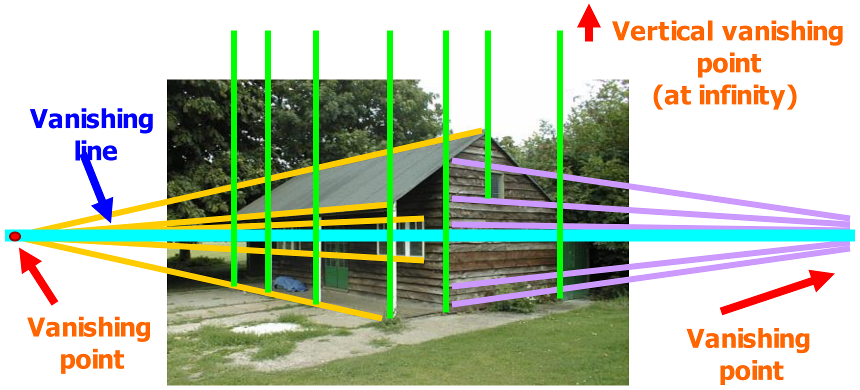


Figure 8. *Top left*: Original image. *Top right*: Detected edge locations. *Middle left*: Intermediate solution to the heat equation. *Middle right*: Reconstructed luminance function. *Bottom left*: Reblurred result. *Bottom right*: Error map (reblurred result—original). Bright indicates overestimation of intensity, dark indicates underestimation. Edge density is 1.7%. RMS error is 10.1 grey levels, with a 3.9 grey level DC component, and an estimated 1.6 grey levels due to noise removal.

Why do we care about edges?

- Extract information, preprocessing

- Recover geometry and viewpoint



Source: J. Hayes

Origins of edges

surface normal discontinuity

depth discontinuity

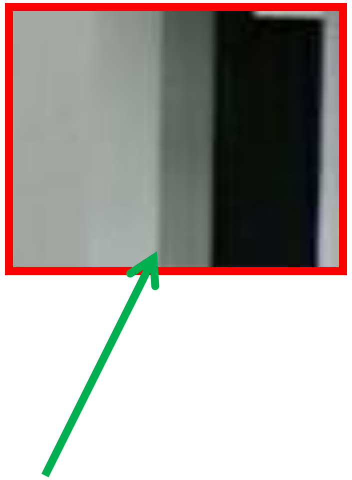
surface color discontinuity

illumination discontinuity

Source: D. Hoiem

Closeup of edges

Surface normal discontinuity



Source: D. Hoiem

Closeup of edges

Depth discontinuity

Source: D. Hoiem

Closeup of edges

Surface color discontinuity

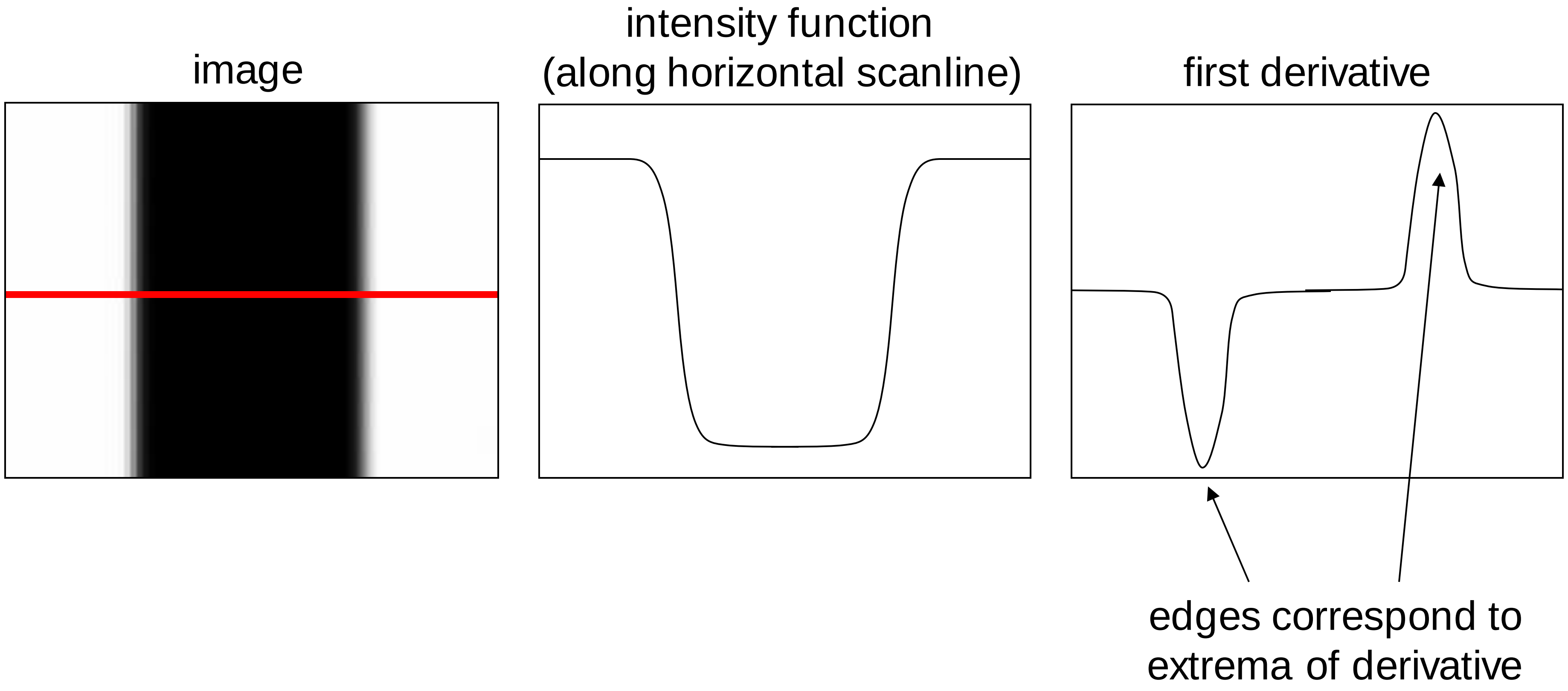
Source: D. Hoiem

What we will learn today

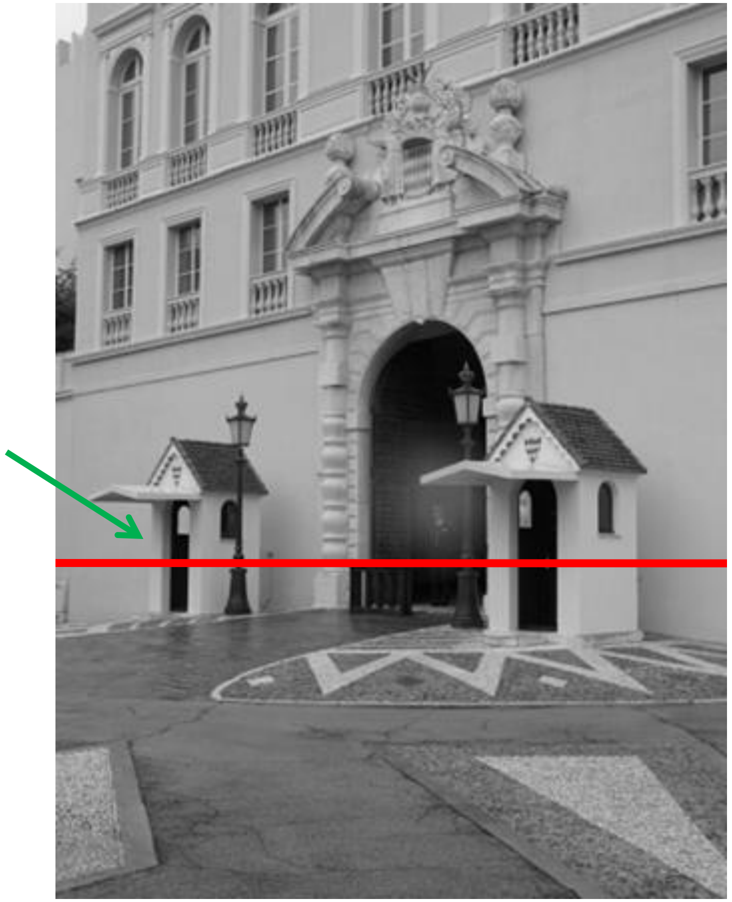
- Edge detection
- Image Gradients
- Derivative of Gaussian
- Sobel edge detector
- Canny edge detector

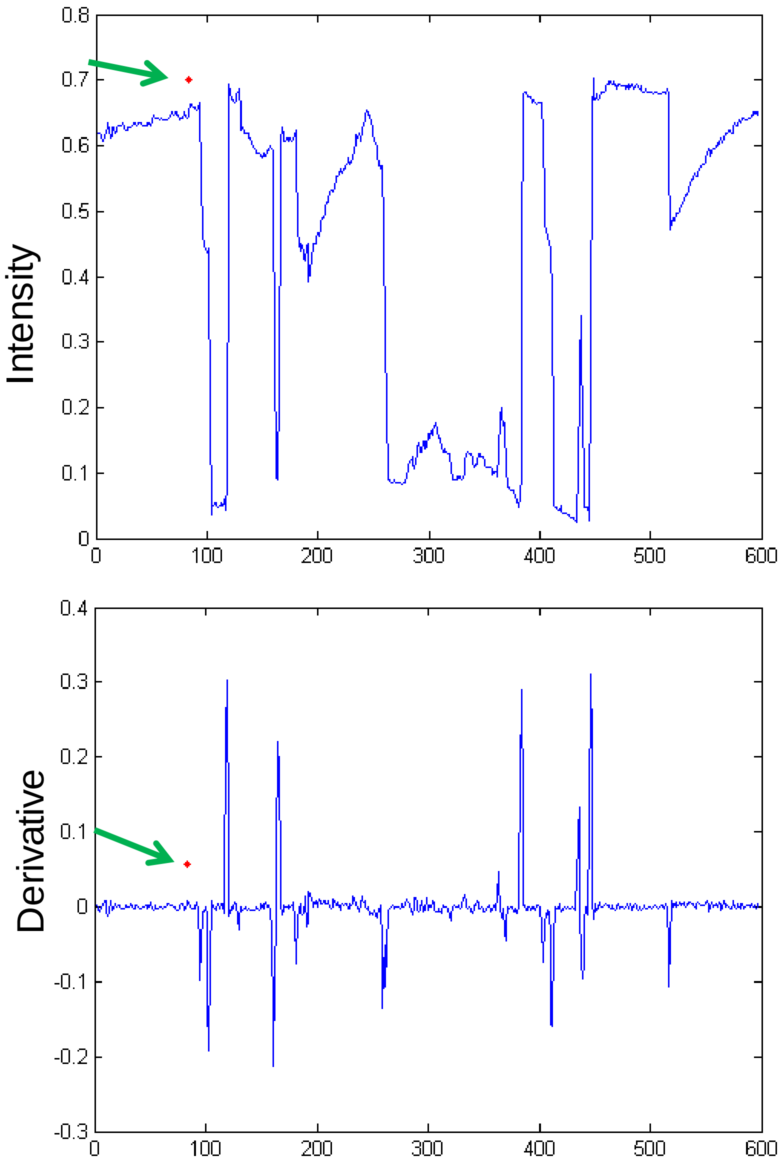
Characterizing edges

- An edge is a place of rapid change in the image intensity function



Intensity profile





Source: D. Hoiem

Types of Discrete derivative in 1D

Backward

$$[-1, 1, 0]$$

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

Forward

$$[0, -1, 1]$$

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

Central

$$\frac{[-1, 0, 1]}{2}$$

$$\frac{df}{dx} = \frac{f(x+1) - f(x-1)}{2} = f'(x)$$

1D discrete derivate example

$$[-1, 1, 0]$$

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

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

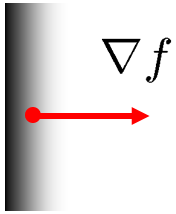
2D discrete derivative - example

$$I = \begin{pmatrix} 8 & 8 & 16 & 16 & 16 \\ 10 & 10 & 20 & 20 & 20 \\ 10 & 10 & 20 & 20 & 20 \\ 10 & 10 & 20 & 20 & 20 \\ 8 & 8 & 16 & 16 & 16 \end{pmatrix} \qquad [-1, 1, 0]$$

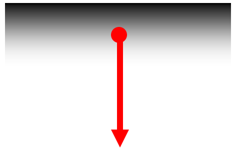
$$I_{\mathrm{x}} = \begin{pmatrix} 0 & 0 & 8 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 8 & 0 & 0 \end{pmatrix}$$

Image gradient

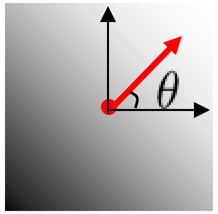
$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$



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



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



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

The gradient “angle” is given by $\theta = \tan^{-1}\left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x}\right)$

The *edge strength* is given by the gradient magnitude

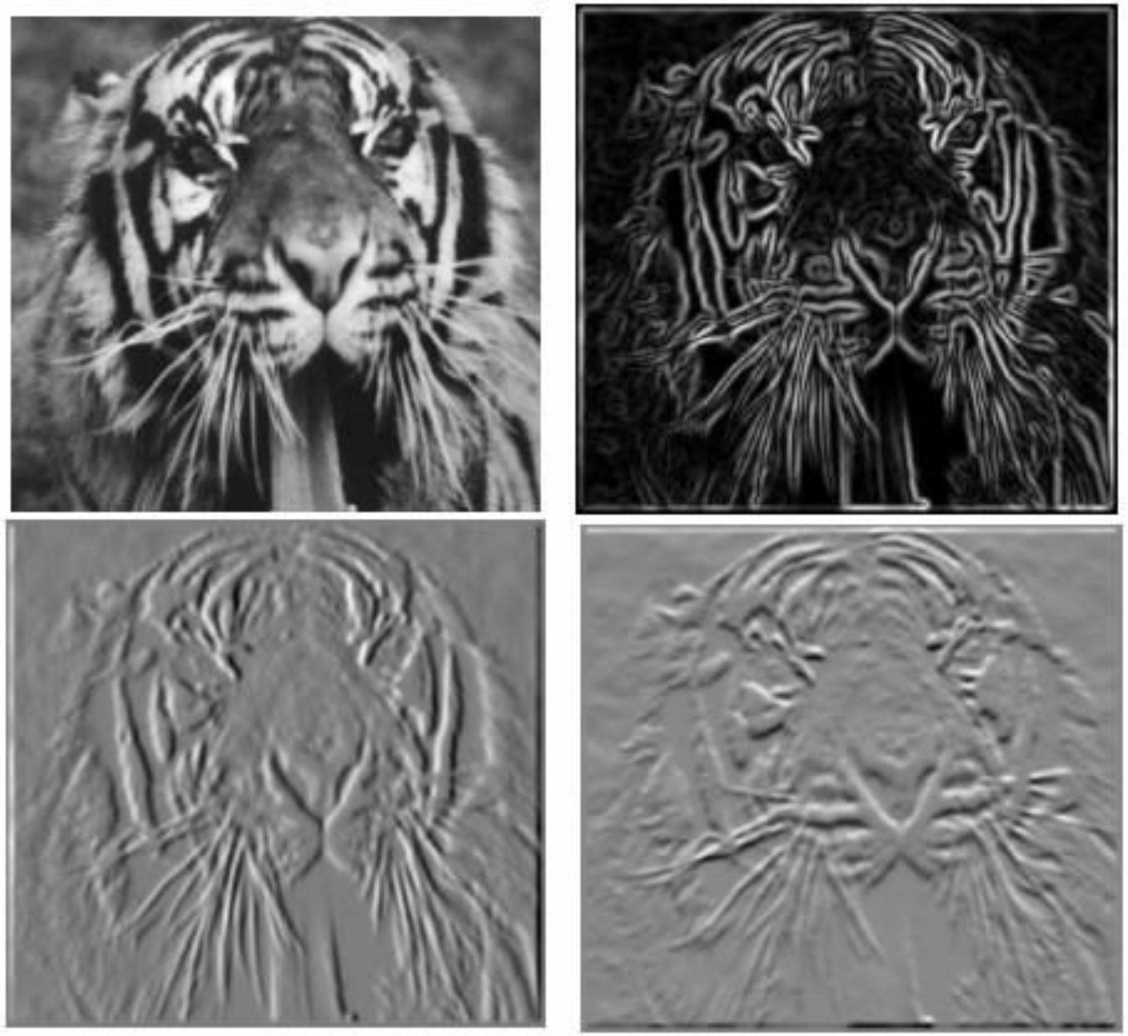
$$\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

Source: Steve Seitz

Finite differences: example

Original Image

Gradient magnitude



x-direction

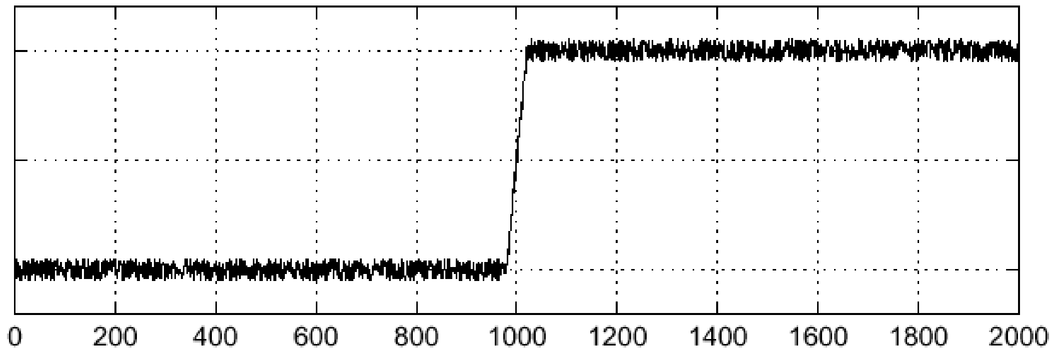
y-direction

- Which one is the gradient in the x-direction? How about y-direction?

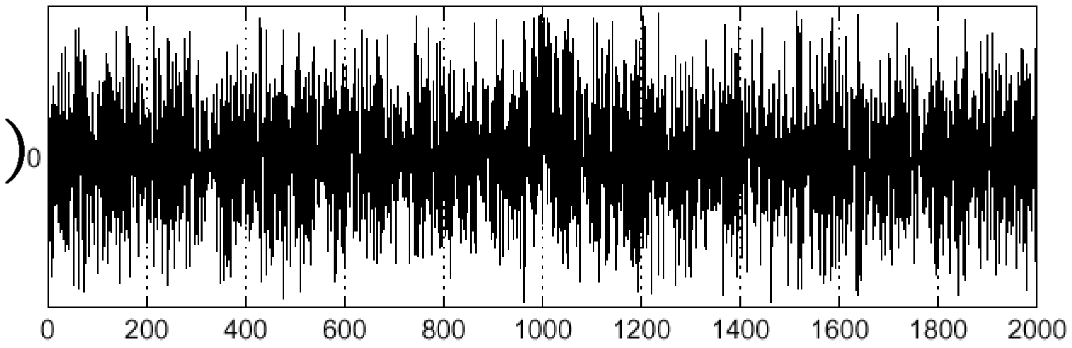
Effects of noise

- Consider a single row or column of the image
  - Plotting intensity as a function of position gives a signal

$f(x)$



$\frac{d}{dx}f(x)$



Where is the edge?

Source: S. Seitz

What we will learn today

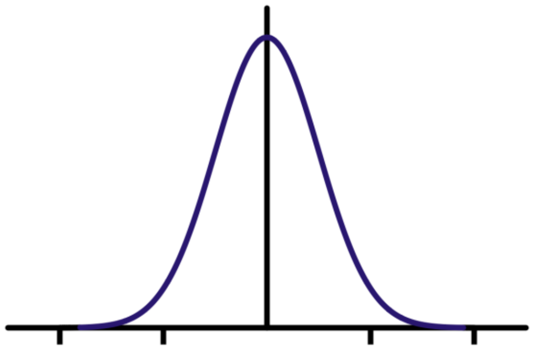
- Edge detection
- Image Gradients
- Derivative of Gaussian
- Sobel edge detector
- Canny edge detector

Smoothing with different filters

- Mean smoothing

$$\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \qquad \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$$

- Gaussian (smoothing * derivative)



$$\begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \qquad \begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$$

Slide credit: Steve Seitz

Discrete Approximation

- Prewitt edge detector

$$\mathbf{G}_x = \begin{bmatrix} +1 & 0 & -1 \\ +1 & 0 & -1 \\ +1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} +1 & 0 & -1 \end{bmatrix}$$

- Sobel edge detector

smoothing

differentiation

$$\mathbf{G}_x = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \begin{bmatrix} +1 & 0 & -1 \end{bmatrix}$$

Sobel Operation

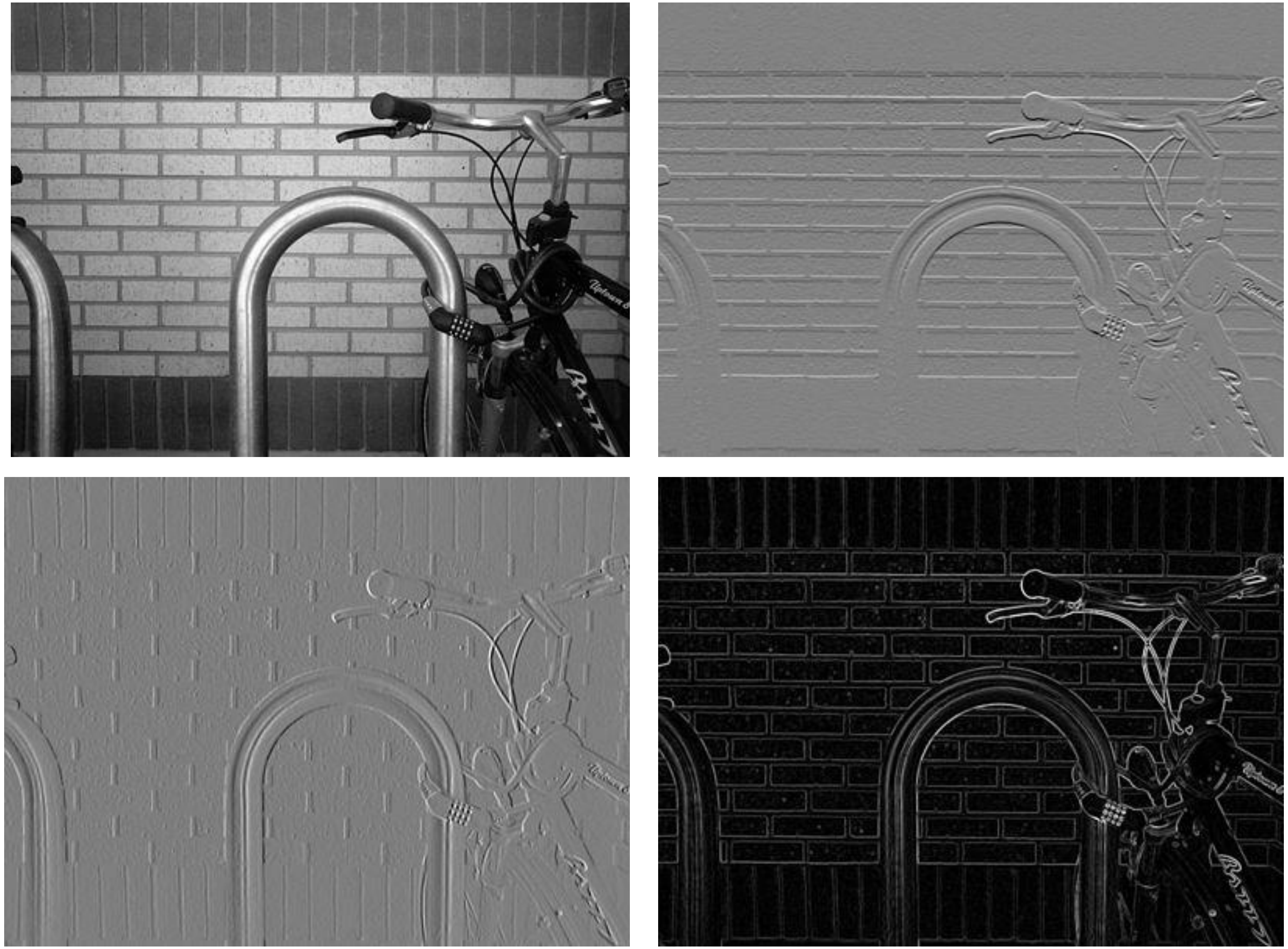
- Magnitude:

$$\mathbf{G} = \sqrt{{\mathbf{G}_x}^2 + {\mathbf{G}_y}^2}$$

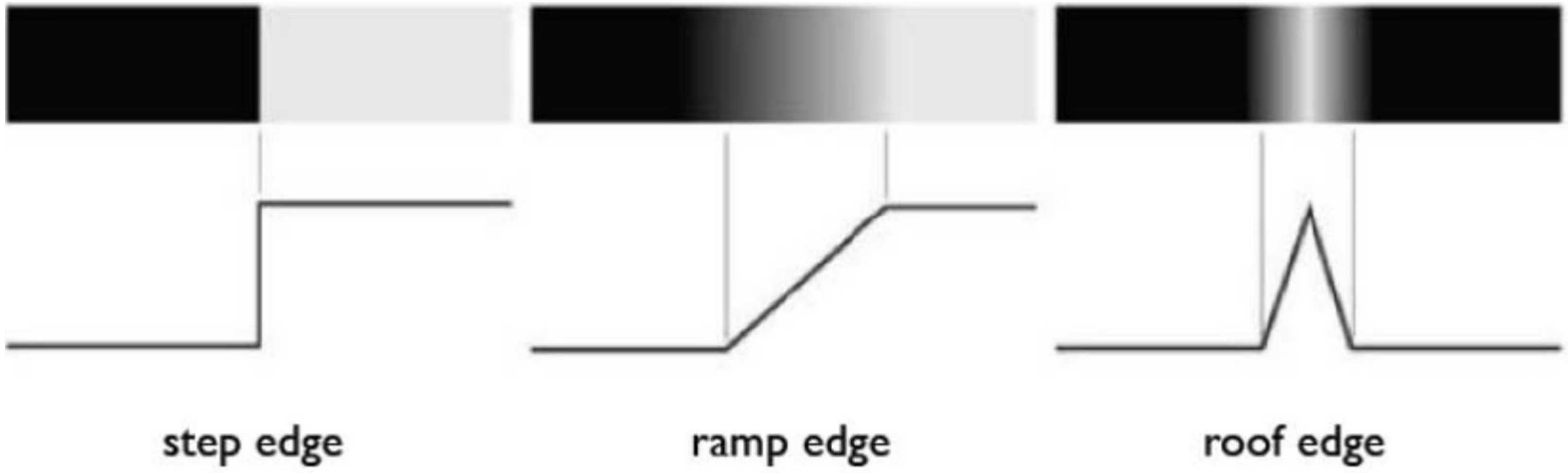
- Angle or direction of the gradient:

$$\mathbf{\Theta} = \operatorname{atan}\left(\frac{\mathbf{G}_y}{\mathbf{G}_x}\right)$$

Sobel Filter example

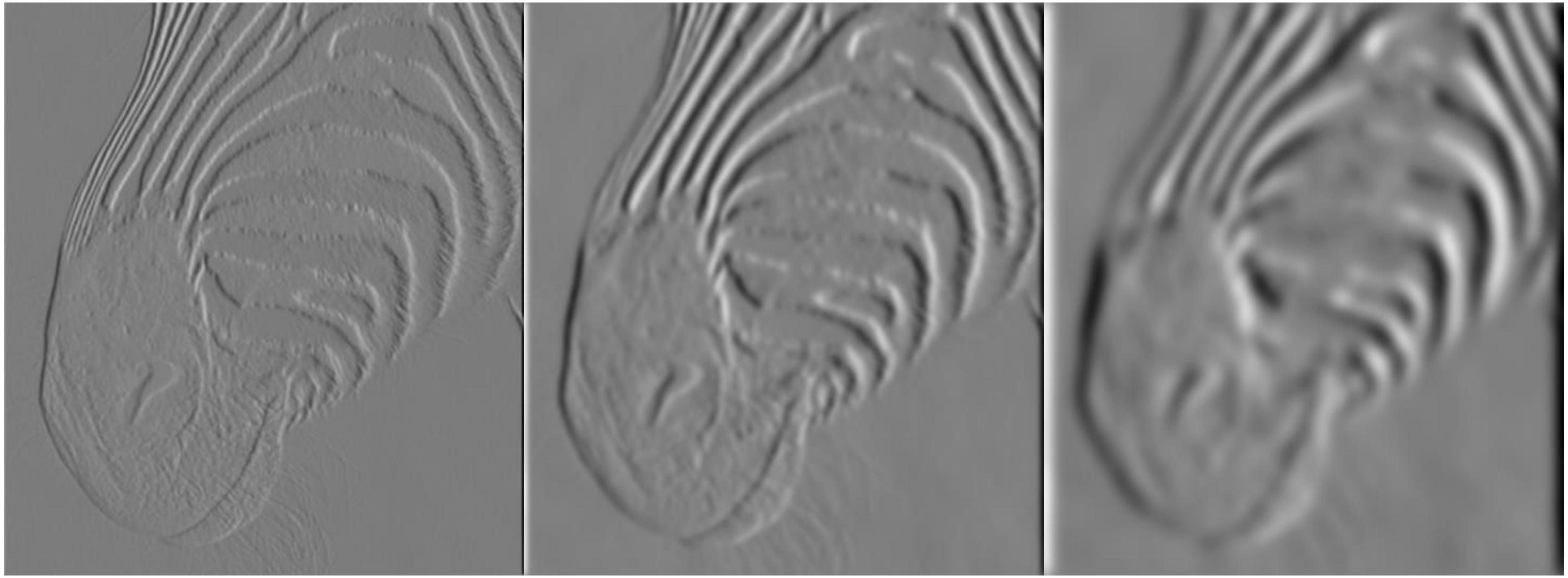


Sobel Filter Problems



- Poor Localization (Trigger response in multiple adjacent pixels)
- Thresholding value favors certain directions over others
  - Can miss oblique edges more than horizontal or vertical edges
  - False negatives

Tradeoff between smoothing at different scales



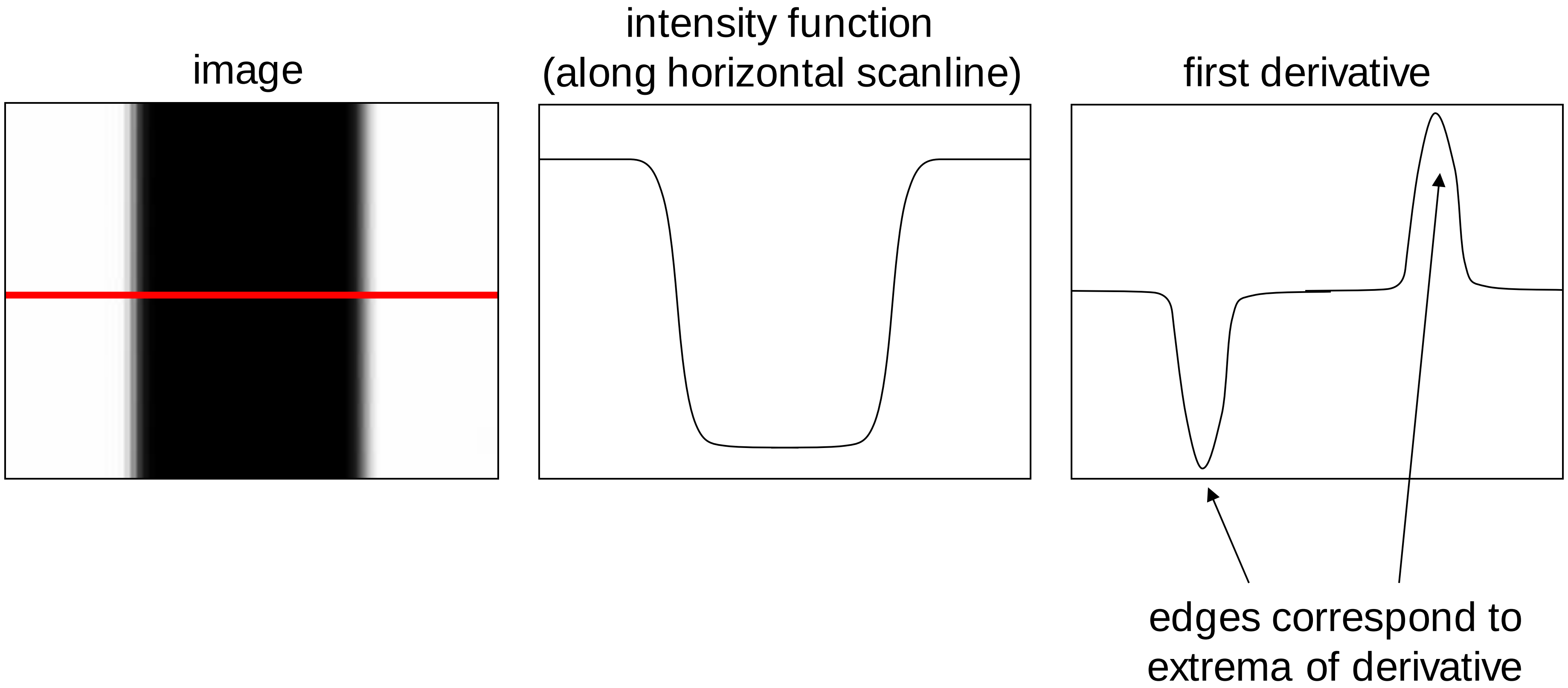
1 pixel | 3 pixels | 7 pixels

- Smoothed derivative removes noise, but blurs edge. Also finds edges at different “scales”.

Source: D. Forsyth

Summary: Characterizing edges

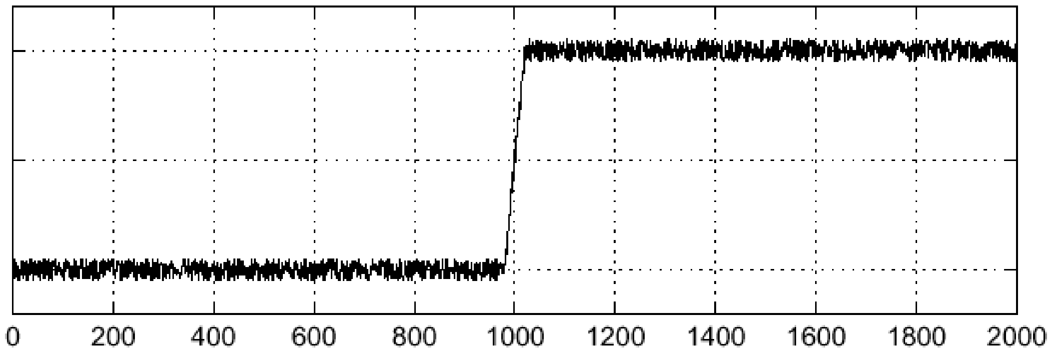
- An edge is a place of rapid change in the image intensity function



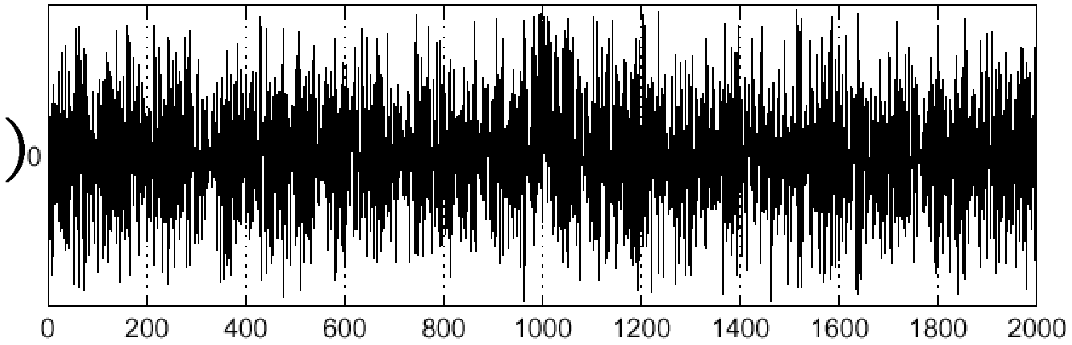
Effects of noise

- Consider a single row or column of the image
  - Plotting intensity as a function of position gives a signal

$f(x)$



$\frac{d}{dx}f(x)$



Where is the edge?

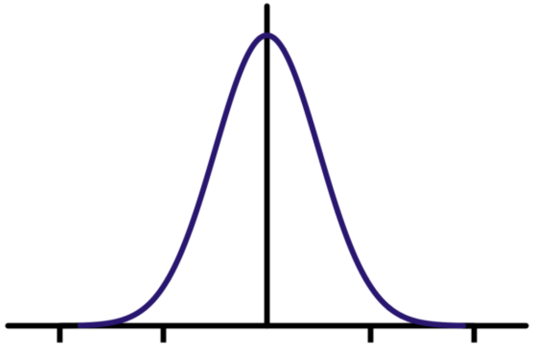
Source: S. Seitz

Smoothing with different filters

- Mean smoothing

$$\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \qquad \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$$

- Gaussian (smoothing * derivative)



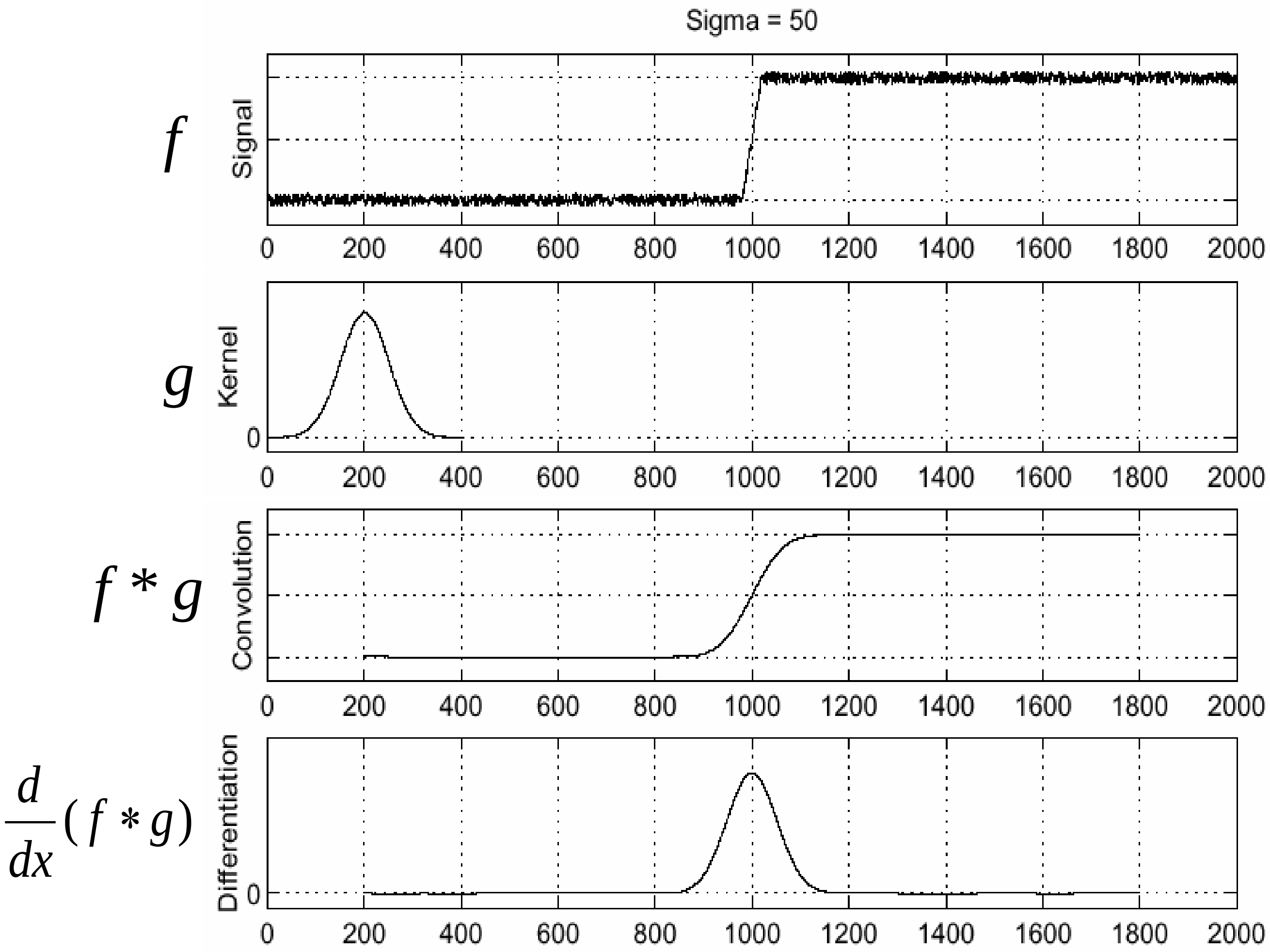
$$\begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \qquad \begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$$

Slide credit: Steve Seitz

What we will learn today

- Edge detection
- Image Gradients
- Derivative of Gaussian
- Sobel edge detector
- Canny edge detector

Smooth with Gaussian filter



- To find edges, look for peaks in $\frac{d}{dx}(f * g)$

Source: S. Seitz

Derivative theorem of convolution

- This theorem gives us a very useful property:

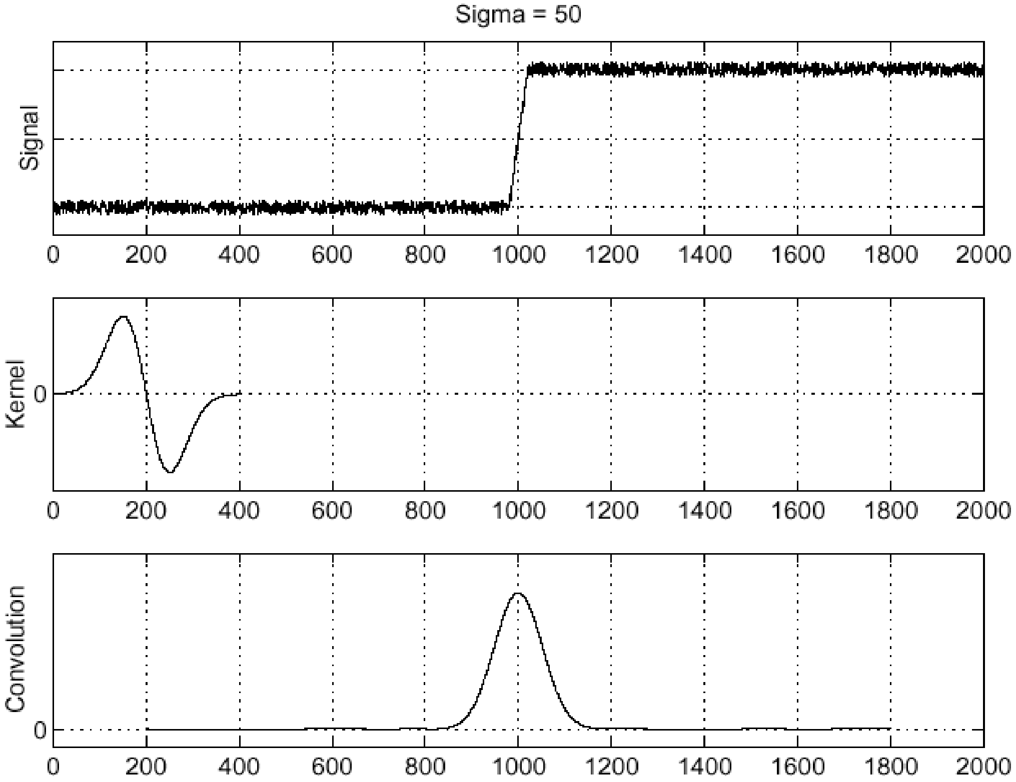
$$\frac{d}{dx}(f*g)=\frac{d}{dx}\int_{x'}f(x')g(x-x')dx'=\int_{x'}f(x')\frac{d}{dx}g(x-x')dx'=f*\frac{d}{dx}g$$

- This saves us one operation:

f

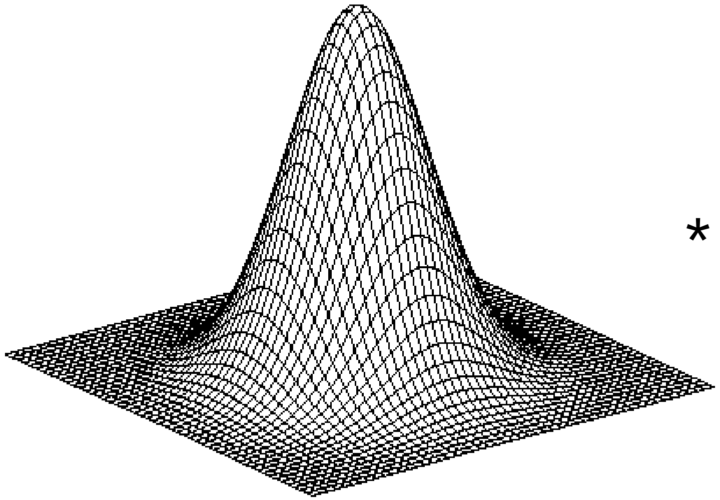
$\frac{d}{dx}g$

$f*\frac{d}{dx}g$



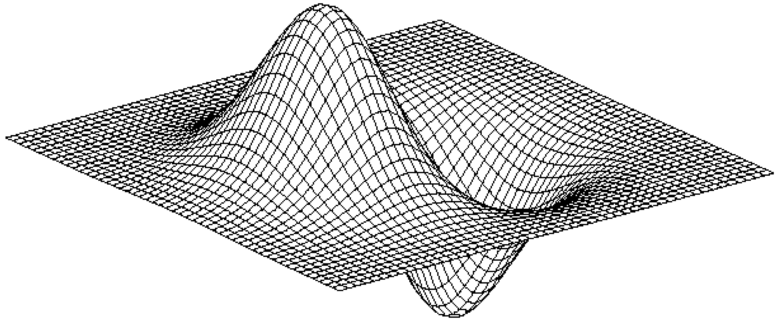
Source: S. Seitz

Derivative of Gaussian filter



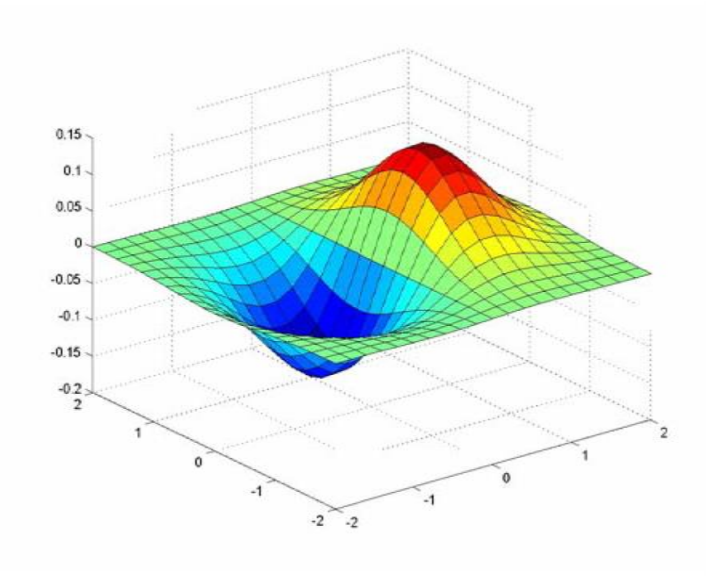
2D-gaussian

* [1   0   -1] =

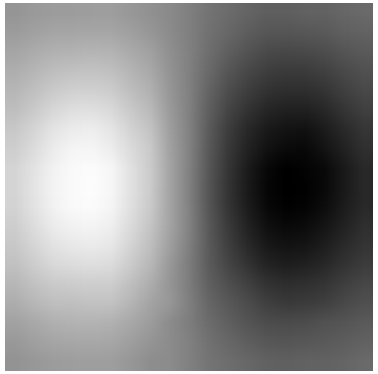


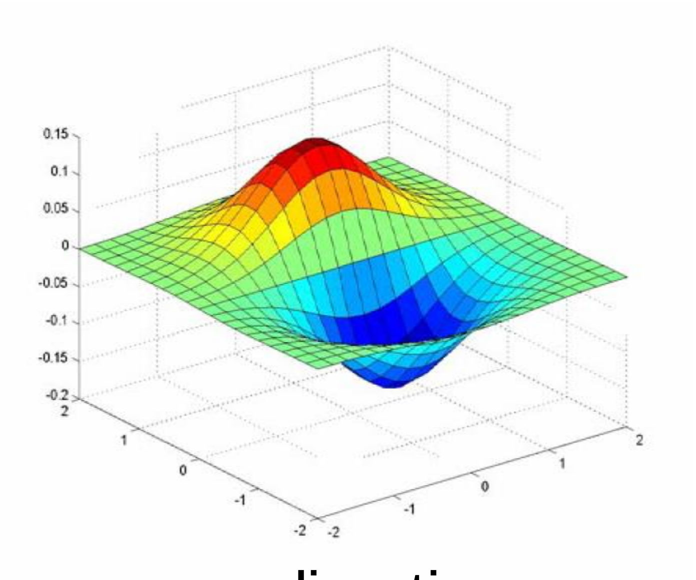
x - derivative

Derivative of Gaussian filter

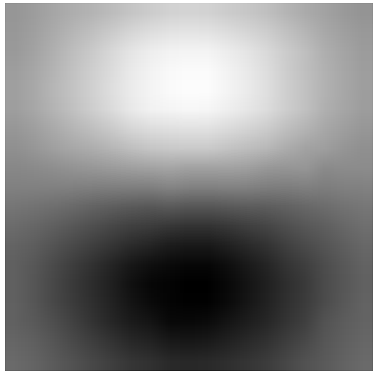


x-direction





y-direction



Derivative of Gaussian vs Sobel Operator

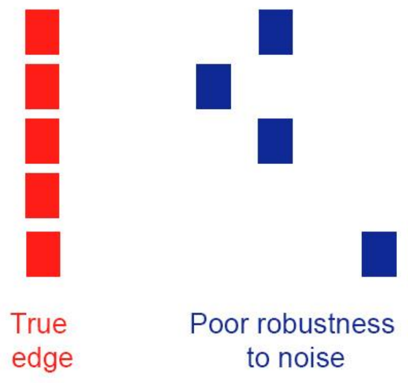
- Derivative of Gaussian smooth out with all near-by neighbors
- Sobel smooth out only with neighbors $\perp$ to the derivative's dir

$$\mathbf{G}_x = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} \qquad \mathbf{G}_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Derivative of Gaussian filter

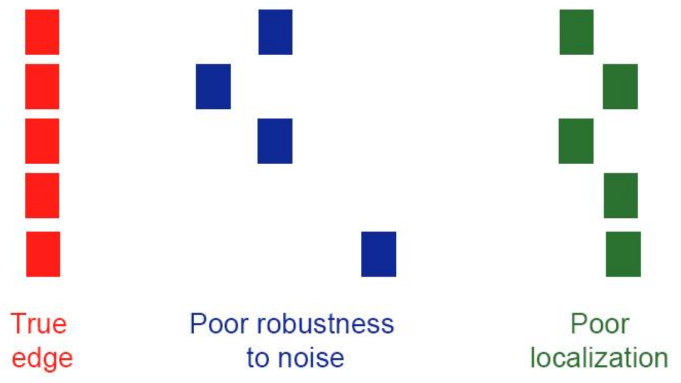
Designing an edge detector

- Criteria for an "optimal" edge detector:
  - **Accurate:** low false positives (detecting spurious edges caused by noise), and low false negatives (missing real edges)



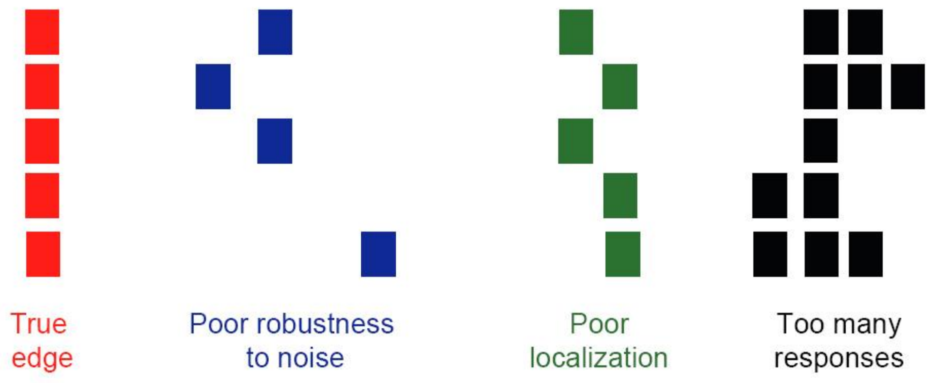
Designing an edge detector

- Criteria for an "optimal" edge detector:
  - **Accurate:** low false positives (detecting spurious edges caused by noise), and low false negatives (missing real edges)
  - **Good localization:** the edges detected must be as close as possible to the true edges



Designing an edge detector

- Criteria for an "optimal" edge detector:
  - **Accurate:** low false positives (detecting spurious edges caused by noise), and low false negatives (missing real edges)
  - **Good localization:** the edges detected must be as close as possible to the true edges
  - **Single response:** the detector must return one point only for each true edge point; that is, minimize the number of local maxima around the true edge



What we will learn today

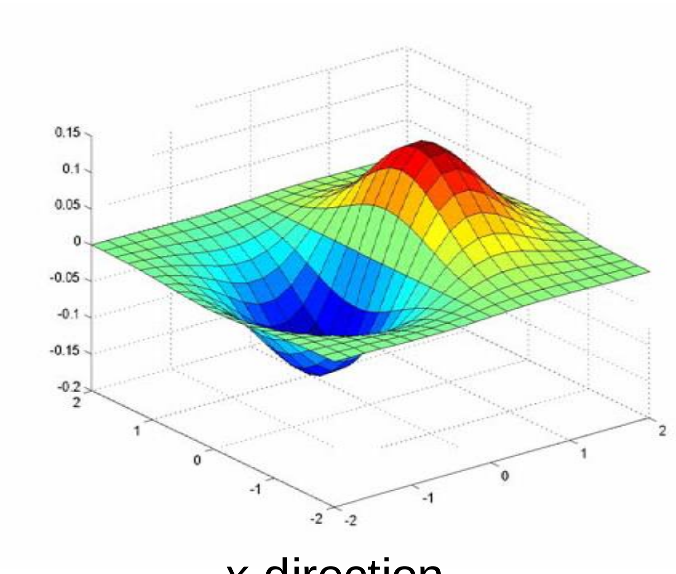
- Edge detection
- Image Gradients
- A simple edge detector
- Sobel edge detector
- Canny edge detector

Canny edge detector

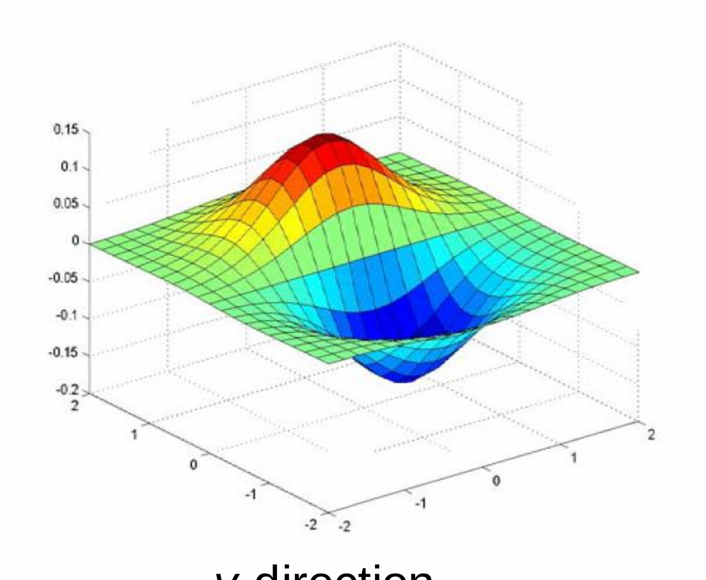
1. Filter image with *x*, *y* derivatives of Gaussian

Source: D. Lowe, L. Fei-Fei

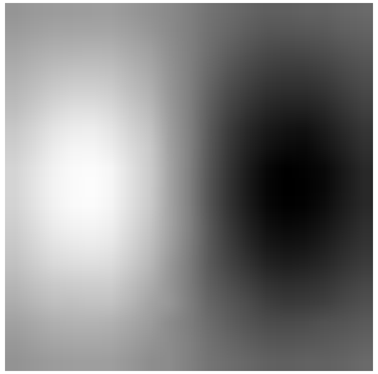
Derivative of Gaussian filter

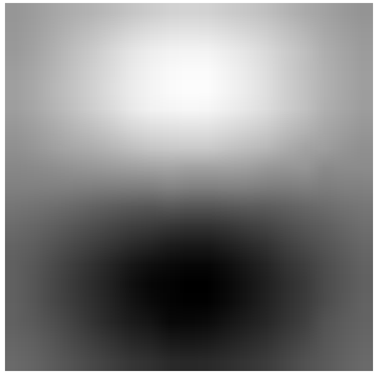


x-direction



y-direction

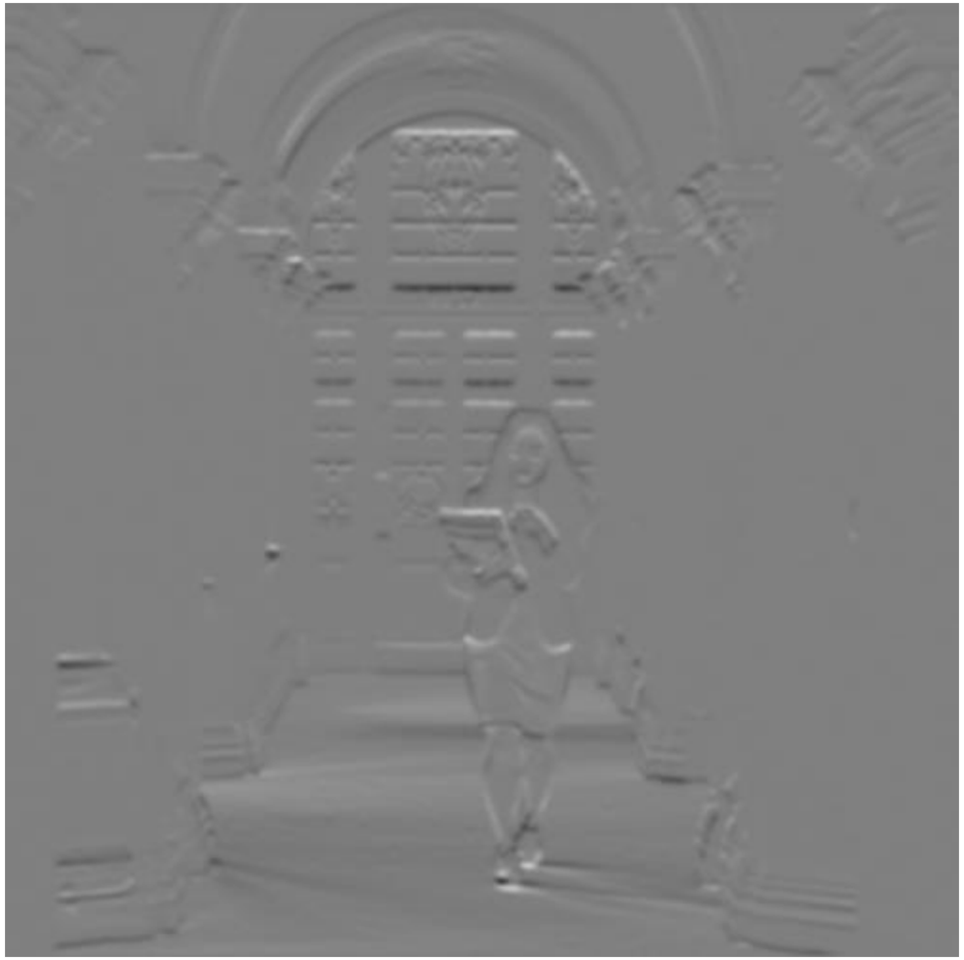




Compute Gradients

gx: X Derivative of Gaussian

gy: Y Derivative of Gaussian



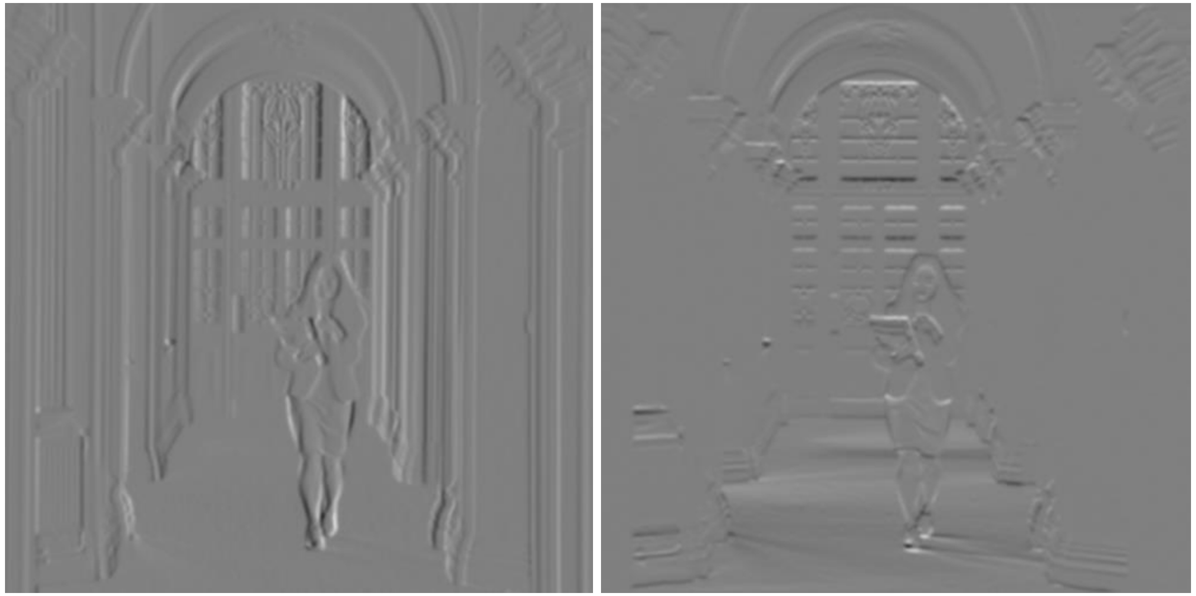
Canny edge detector

1. Filter image with *x*, *y* derivatives of Gaussian
2. Find magnitude and orientation of gradient

Source: D. Lowe, L. Fei-Fei

Compute Gradient Magnitude

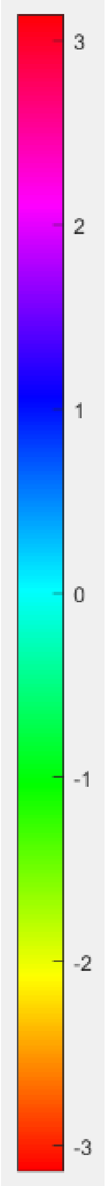
$\sqrt{gx^2 + gy^2}$ = gradient magnitude



(x4 for visualization)

Compute Gradient Orientation

- Threshold magnitude at minimum level
- Get orientation via $\theta = \text{atan}\left(\frac{gy}{gx}\right)$

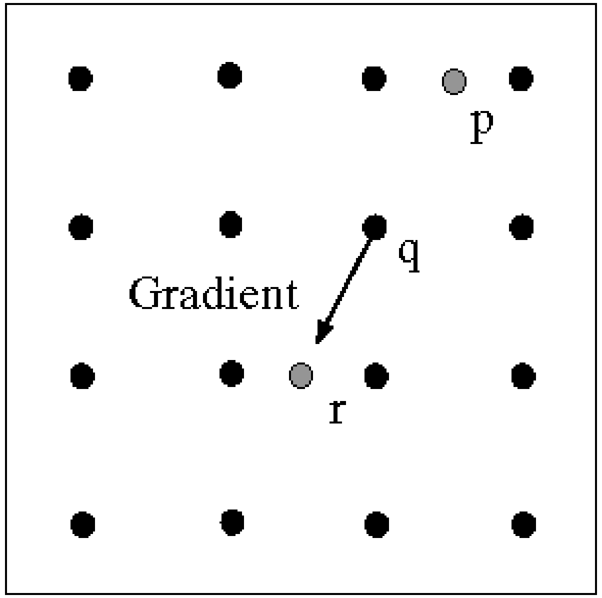


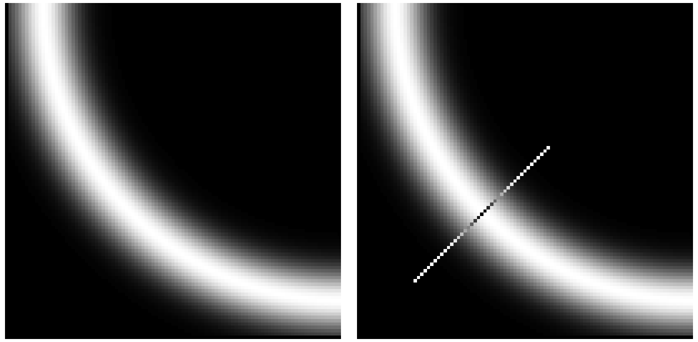
Canny edge detector

1. Filter image with *x*, *y* derivatives of Gaussian
2. Find magnitude and orientation of gradient
3. Non-maximum suppression:
   – Thin multi-pixel wide “ridges” to single pixel width

Source: D. Lowe, L. Fei-Fei

Non-maximum suppression for each orientation



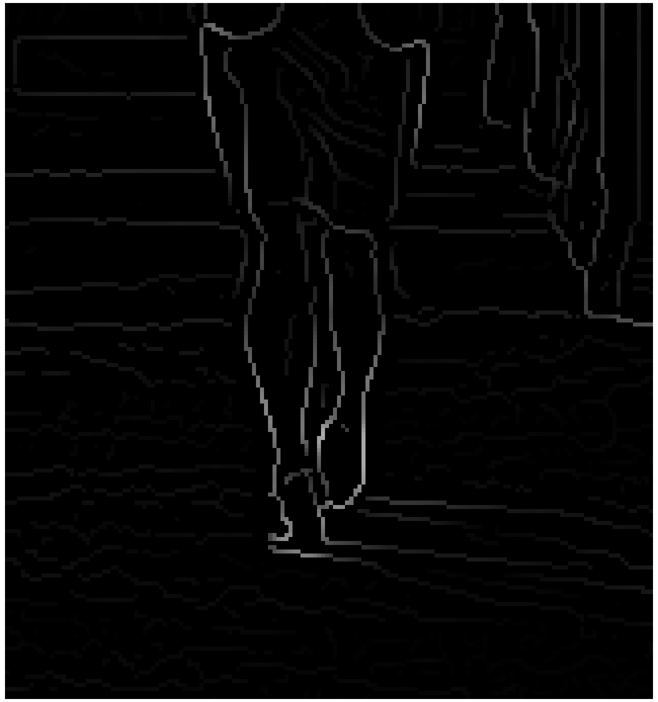


Source: D. Forsyth

Before Non-max Suppression

Gradient magnitude (x4 for visualization)

After non-max suppression



Gradient magnitude (x4 for visualization)

Canny edge detector

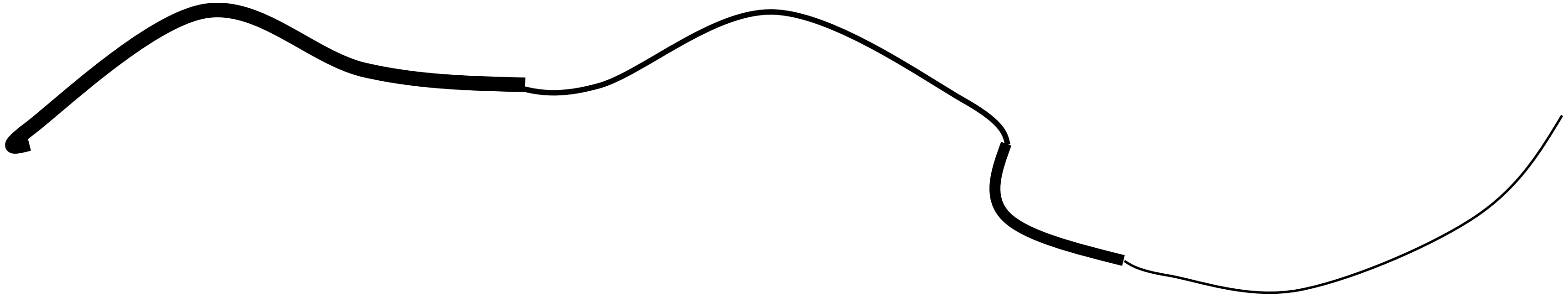
1. Filter image with *x*, *y* derivatives of Gaussian
2. Find magnitude and orientation of gradient
3. Non-maximum suppression:
   - Thin multi-pixel wide “ridges” to single pixel width
4. ‘Hysteresis’ Thresholding

Source: D. Lowe, L. Fei-Fei

‘Hysteresis’ (adaptive) thresholding

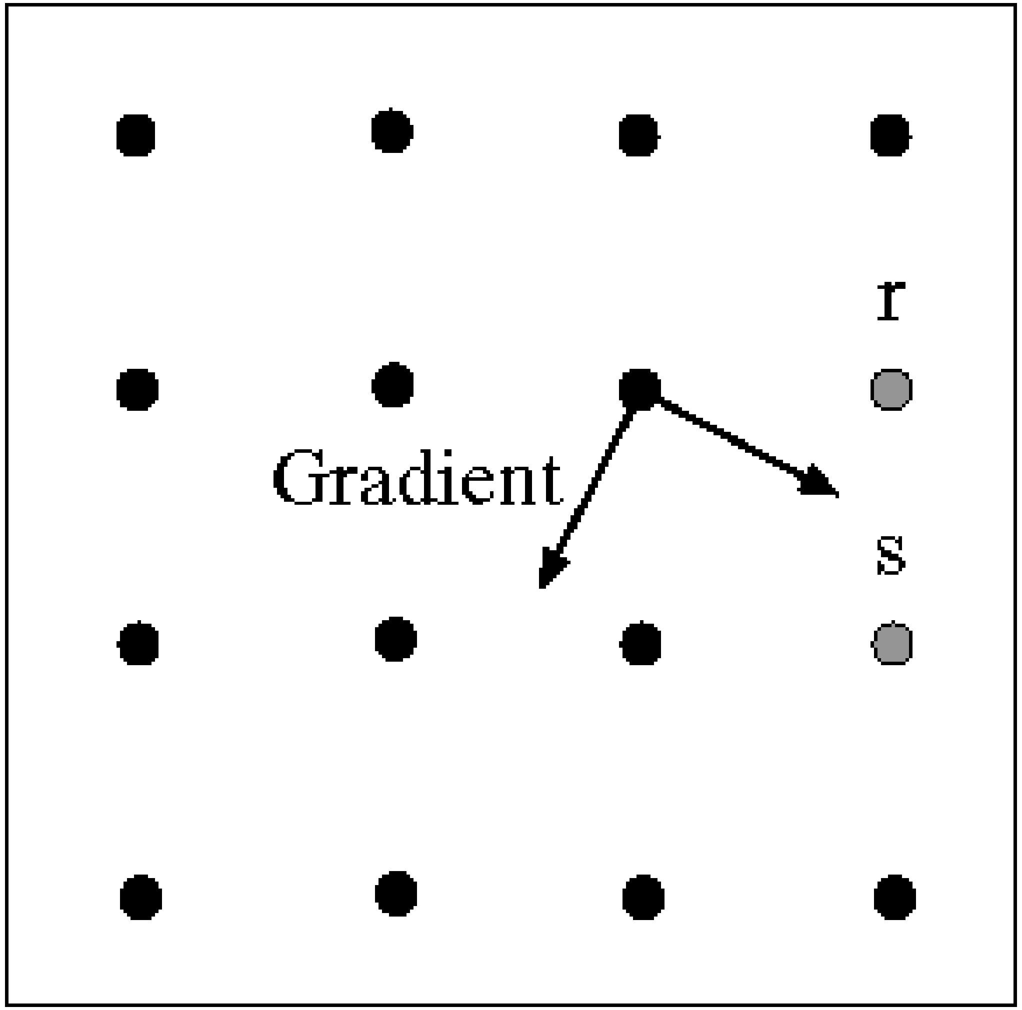
- Two thresholds – high and low
- Grad. mag. > high threshold? = strong edge
- Grad. mag. < low threshold? noise
- In between = weak edge

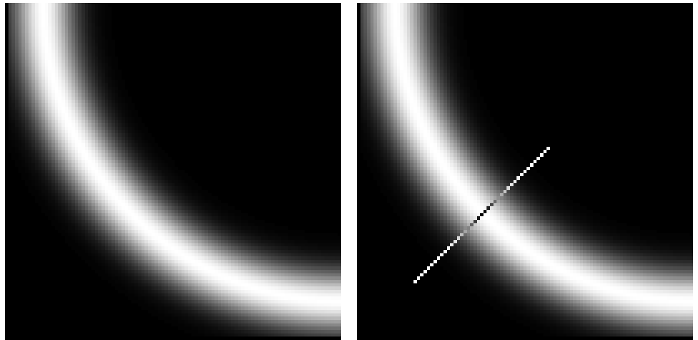
- ‘Follow’ edges starting from strong edge pixels
- Continue them into weak edges
    - Connected components (Szeliski 3.3.4)



Source: S. Seitz

Edge linking





Source: D. Forsyth

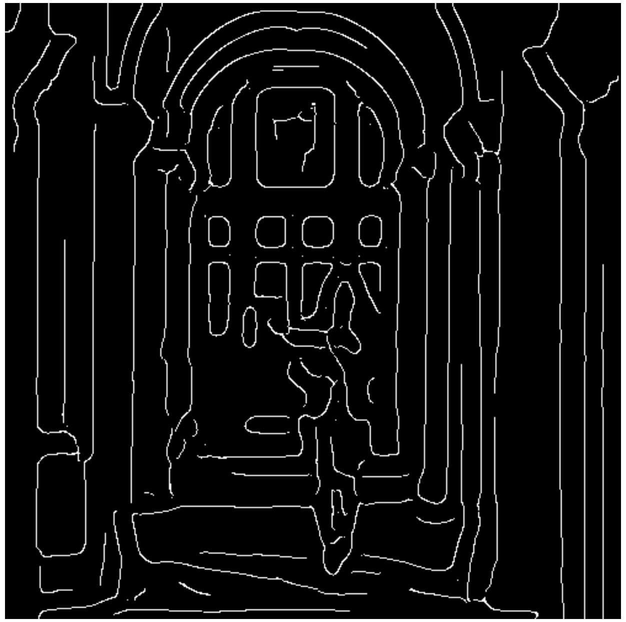
Final Canny Edges

$$\sigma = \sqrt{2}, t_{low} = 0.05, t_{high} = 0.1$$

Effect of σ (Gaussian kernel spread/size)

Original

$\sigma = \sqrt{2}$



$\sigma = 4\sqrt{2}$

The choice of σ depends on desired behavior

- large σ detects large scale edges
- small σ detects fine features

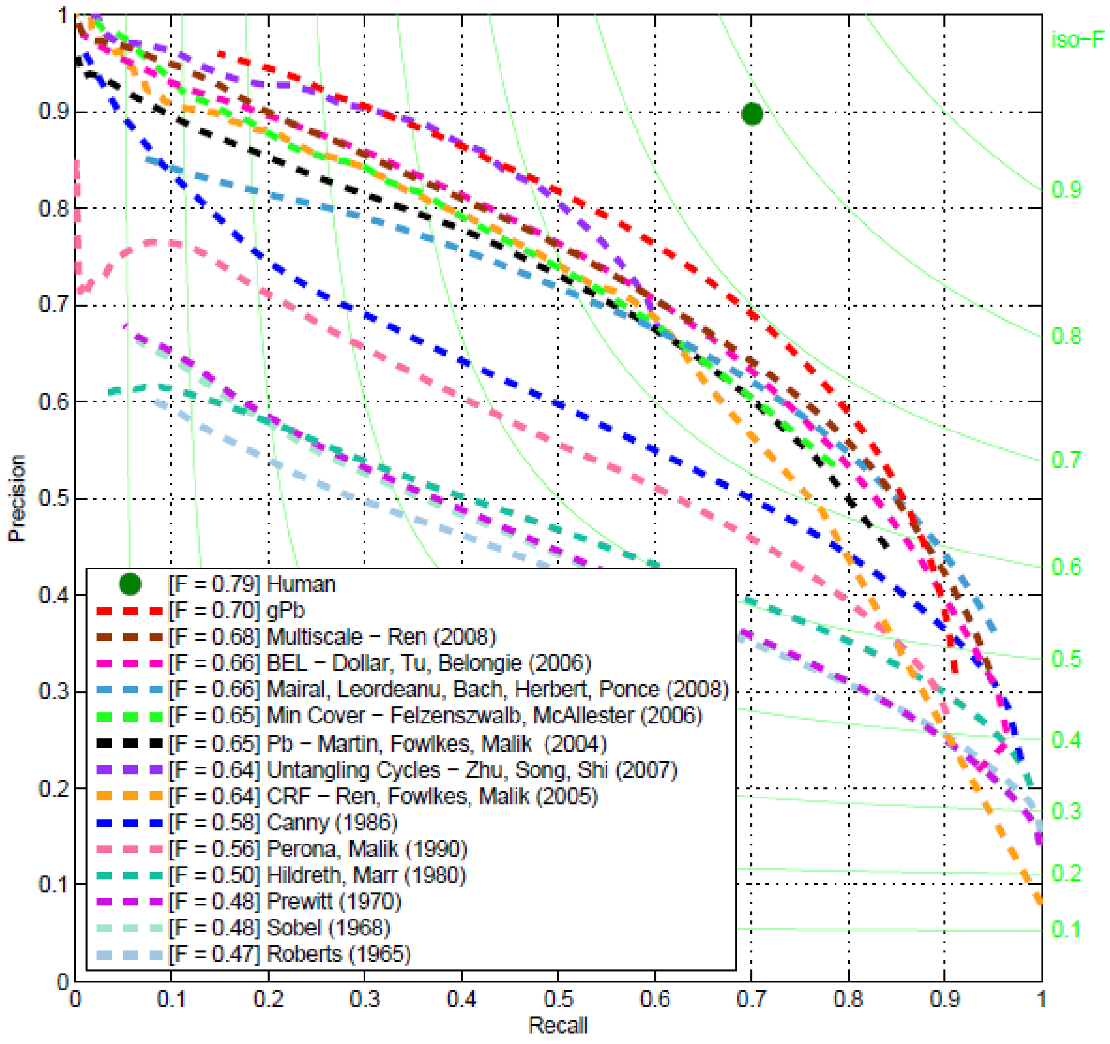
Source: S. Seitz

Canny edge detector

1. Filter image with *x*, *y* derivatives of Gaussian
2. Find magnitude and orientation of gradient
3. Non-maximum suppression:
   - Thin multi-pixel wide “ridges” to single pixel width
4. ‘Hysteresis’ Thresholding:
   - Define two thresholds: low and high
   - Use the high threshold to start edge curves and the low threshold to continue them
   - ‘Follow’ edges starting from strong edge pixels
     - Connected components (Szeliski 3.3.4)

Source: D. Lowe, L. Fei-Fei

45 years of boundary detection



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

Summary

- Edge can be approximated with image gradient
  - Magnitude: edge strength
  - Direction: Orientation
- Filtering is needed to reduce image noise
  - Can combine with derivative to form a single filter
  - Sober/Derivative of Gaussian
- Canny edge detector
  - Derivative of Gaussian
  - Non-maximum suppression
  - Adaptive thresholding