CAP 5415 Computer Vision
Fall 2005

Dr. Alper Yilmaz
Univ. of Central Florida
www.cs.ucf.edu/courses/cap5415/fall2005
Office: CSB 250

Recap (Edge Detection)

- Prewitt and Sobel edge detectors
  - Compute derivatives
    - In x and y directions
  - Find gradient magnitude
  - Threshold gradient magnitude
- Difference between Prewitt and Sobel is the derivative filters

Prewitt Edge Detector

Prewitt's edges in x direction

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

Prewitt's edges in y direction

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

Alper Yilmaz, Fall 2005 UCF

Sobel Edge Detector

Sobel's edges in x direction

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

Sobel's edges in y direction

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

Alper Yilmaz, Fall 2005 UCF

Edge Detection Continued

- Marr Hildreth Edge Detector
- Canny Edge Detector

Marr Hildreth Edge Detector

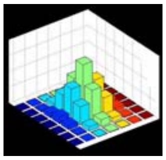
- Smooth image by Gaussian filter ➔ S
- Apply Laplacian to S
  - Used in mechanics, electromagnetics, wave theory, quantum mechanics and Laplace equation
- Find zero crossings
  - Scan along each row, record an edge point at the location of zero-crossing.
  - Repeat above step along each column

Marr Hildreth Edge Detector

- Gaussian smoothing

$$\overbrace{S}^{\text{smoothed image}} = \overbrace{g}^{\text{Gaussian filter}} * \overbrace{I}^{\text{image}}$$

$$g = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2+y^2}{2\sigma^2}}$$



- Find Laplacian

$$\Delta^2 S = \overbrace{\frac{\partial^2}{\partial x^2} S}^{\text{second order derivative in } x} + \overbrace{\frac{\partial^2}{\partial y^2} S}^{\text{second order derivative in } y}$$

- ∇ is used for gradient (derivative)
- Δ is used for Laplacian

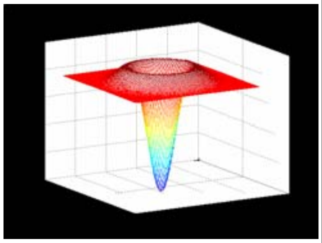
Marr Hildreth Edge Detector

- Deriving the Laplacian of Gaussian (LoG)

$$\Delta^2 S = \Delta^2 (g * I) = (\Delta^2 g) * I$$

Homework

$$\Delta^2 g = -\frac{1}{\sqrt{2\pi}\sigma^3}\left(2 - \frac{x^2+y^2}{\sigma^2}\right) e^{-\frac{x^2+y^2}{2\sigma^2}}$$



LoG Filter

$$\Delta^2 G_\sigma = -\frac{1}{\sqrt{2\pi}\sigma^3}\left(2-\frac{x^2+y^2}{\sigma^2}\right)e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Y

0.0008	0.0066	0.0215	0.031	0.0215	0.0066	0.0008	
0.0066	0.0438	0.0982	0.108	0.0982	0.0438	0.0066	
0.0215	0.0982	0	-0.242	0	0.0982	0.0215	
0.031	0.108	-0.242	-0.7979	-0.242	0.108	0.031	X
0.0215	0.0982	0	-0.242	0	0.0982	0.0215	
0.0066	0.0438	0.0982	0.108	0.0982	0.0438	0.0066	
0.0008	0.0066	0.0215	0.031	0.0215	0.0066	0.0008	

Finding Zero Crossings

- Four cases of zero-crossings :
  - {+,-}
  - {+,0,-}
  - {-,+}
  - {-,0,+}
- Slope of zero-crossing {a, -b} is |a+b|.
- To mark an edge
  - compute slope of zero-crossing
  - Apply a threshold to slope

On the Separability of LoG

- Similar to separability of Gaussian filter
  - Two-dimensional Gaussian can be separated into 2 one-dimensional Gaussians

$$h(x,y) = I(x,y) * g(x,y)$$ n^2 multiplications

$$h(x,y) = \left(I(x,y) * g_1(x)\right) * g_2(y)$$ $2n$ multiplications

$$g_1 = [.011 \quad .13 \quad .6 \quad 1 \quad .6 \quad .13 \quad .011]$$

$$g_2 = \begin{bmatrix} .011 \\ .13 \\ .6 \\ 1 \\ .6 \\ .13 \\ .011 \end{bmatrix}$$

Alper Yilmaz, Fall 2005 UCF

On the Separability of LoG

$$\Delta^2 S = \Delta^2 (g * I) = \left(\Delta^2 g\right) * I = I * \left(\Delta^2 g\right)$$

Requires n^2 multiplications

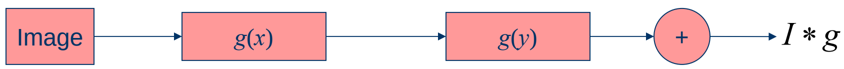
$$\Delta^2 S = \left(I * g_{xx}(x)\right) * g(x) + \left(I * g_{yy}(y)\right) * g(y)$$

Requires $4n$ multiplications

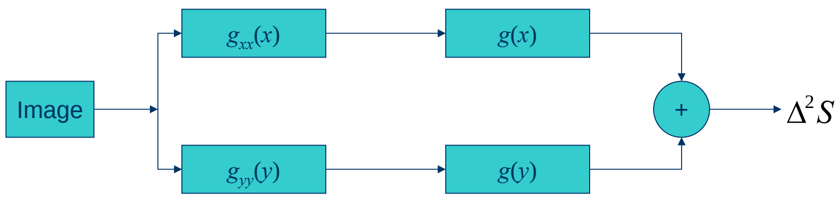
Alper Yilmaz, Fall 2005 UCF

Seperability

Gaussian Filtering



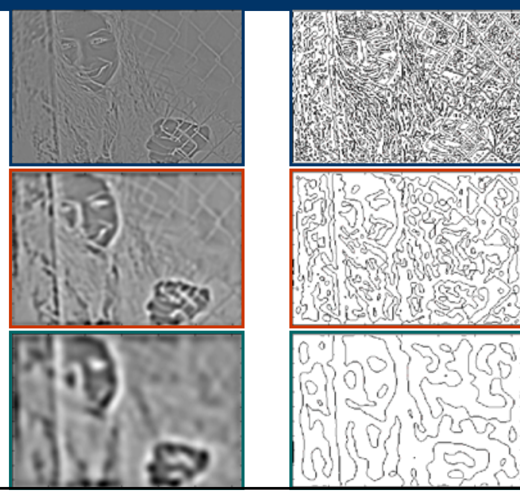
Laplacian of Gaussian Filtering



Example

I | $I * (\Delta^2 g)$ | Zero crossings of $\Delta^2 S$

Example



$\sigma = 1$

$\sigma = 3$

$\sigma = 6$

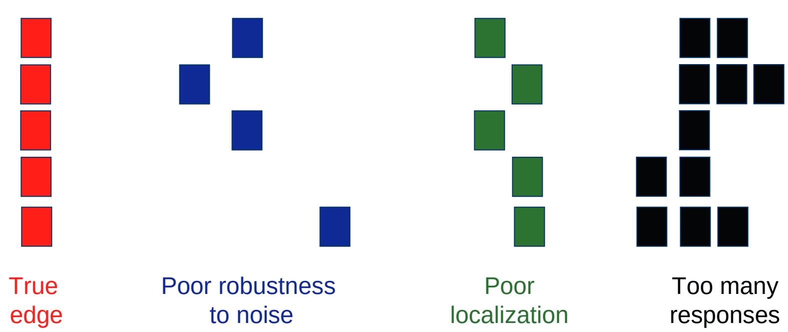
Algorithm

- Compute LoG
  - Use 2D filter $\Delta^2 g(x, y)$
  - Use 4 1D filters $g(x),\ g_{xx}(x),\ g(y),\ g_{yy}(y)$
- Find zero-crossings from each row
- Find slope of zero-crossings
- Apply threshold to slope and mark edges

Quality of an Edge

- Robust to noise
- Localization
- Too many or too less responses

Quality of an Edge



Alper Yilmaz, Fall 2005 UCF

Canny Edge Detector

- Criterion 1: Good Detection: The optimal detector must minimize the probability of false positives as well as false negatives.
- Criterion 2: Good Localization: The edges detected must be as close as possible to the true edges.
- Single Response Constraint: The detector must return one point only for each edge point.

Canny Edge Detector Steps

1. Smooth image with Gaussian filter
2. Compute derivative of filtered image
3. Find magnitude and orientation of gradient
4. Apply “Non-maximum Suppression”
5. Apply “Hysteresis Threshold”

Canny Edge Detector
First Two Steps

- Smoothing

$$S = I * g(x,y) = g(x,y) * I \qquad g(x,y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

- Derivative

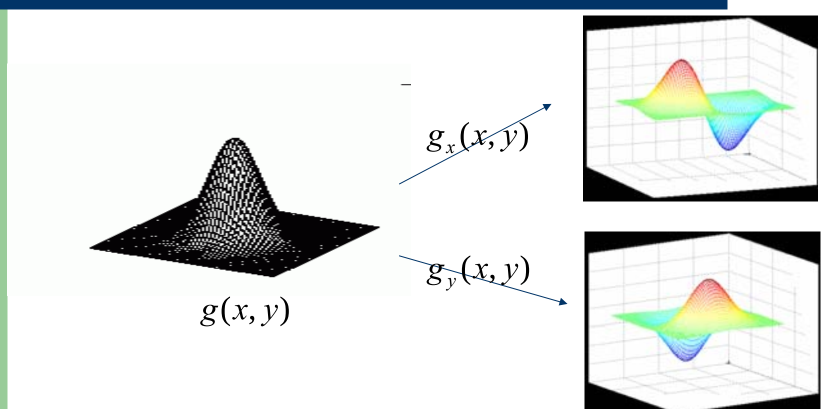
$$\nabla S = \nabla(g * I) = (\nabla g) * I$$

$$\nabla S = \begin{bmatrix} g_x \\ g_y \end{bmatrix} * I = \begin{bmatrix} g_x * I \\ g_y * I \end{bmatrix}$$

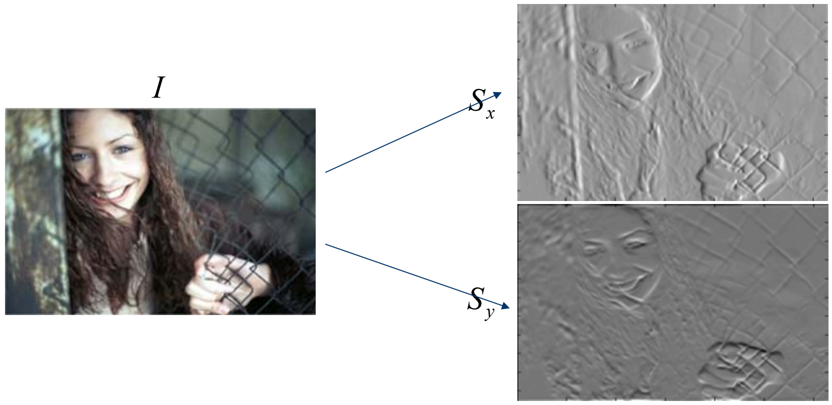
Homework

$$\nabla g = \begin{bmatrix} \frac{\partial g}{\partial x} \\ \frac{\partial g}{\partial y} \end{bmatrix} = \begin{bmatrix} g_x \\ g_y \end{bmatrix}$$

Canny Edge Detector
Derivative of Gaussian



Canny Edge Detector
First Two Steps



Canny Edge Detector
Third Step

- Gradient magnitude and gradient direction

(S_x, S_y) Gradient Vector

magnitude $= \sqrt{(S_x^2 + S_y^2)}$

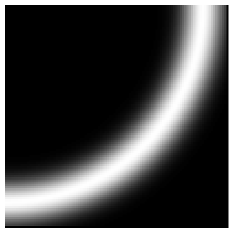
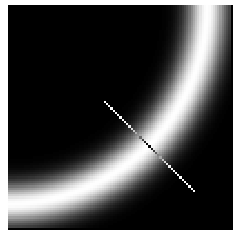
direction $= \theta = \tan^{-1} \frac{S_y}{S_x}$

image

gradient magnitude

Canny Edge Detector
Fourth Step

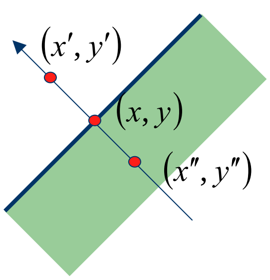
- Non maximum suppression

We wish to mark points along the curve where the **magnitude is biggest**. We can do this by looking for a maximum along a slice normal to the curve (non-maximum suppression). These points should form a curve. There are then two algorithmic issues: at which point is the maximum, and where is the next one?

Canny Edge Detector
Non-Maximum Suppression

- Suppress the pixels in $|\nabla S|$ which are not local maximum



$$M(x,y) = \begin{cases} |\nabla S|(x,y) & \text{if } |\Delta S|(x,y) > |\Delta S|(x',y') \\ & \& \ |\Delta S|(x,y) > |\Delta S|(x'',y'') \\ 0 & \text{otherwise} \end{cases}$$

x' and **x**'' are the neighbors of **x** along normal direction to an edge

Canny Edge Detector
Quantization of Normal Directions

$$\tan\theta = \frac{S_y}{S_x}$$

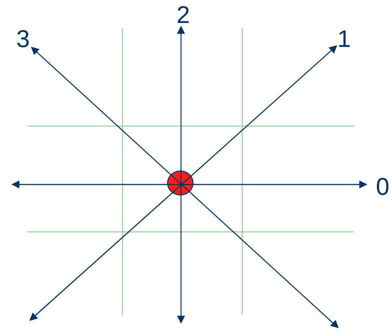
Quantizations:

0: $-0.4142 < \tan\theta \leq 0.4142$

1: $0.4142 < \tan\theta < 2.4142$

2: $|\tan\theta| \geq 2.4142$

3: $-2.4142 < \tan\theta \leq -0.4142$



Alper Yilmaz, Fall 2005 UCF

Canny Edge Detector
Non-Maximum Suppression

$$|\Delta S| = \sqrt{S_x^2 + S_y^2}$$

M

For visualization

$M \geq Threshold = 25$

Alper Yilmaz, Fall 2005 UCF

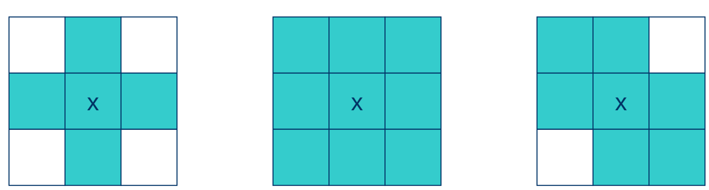
Canny Edge Detector
Hysteresis Thresholding

- If the gradient at a pixel is
  - above "**High**", declare it an '**edge pixel**'
  - below "**Low**", declare it a "**non-edge-pixel**"
  - **between** "low" and "high"
    - Consider its neighbors iteratively then declare it an "edge pixel" if it is **connected** to an 'edge pixel' **directly** or via pixels **between** "low" and "high".

Alper Yilmaz, Fall 2005 UCF

Canny Edge Detector
Hysteresis Thresholding

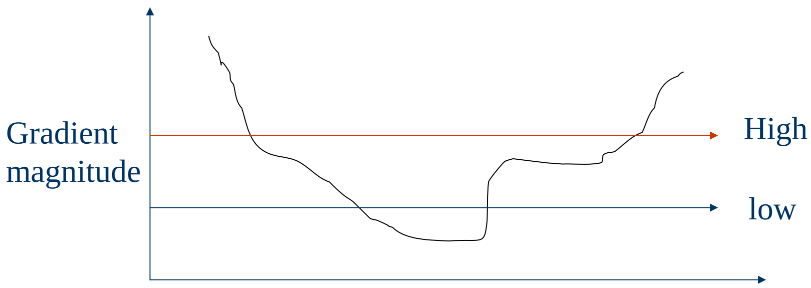
- Connectedness



4 connected | 8 connected | 6 connected

Alper Yilmaz, Fall 2005 UCF

Canny Edge Detector
Hysteresis Thresholding



Alper Yilmaz, Fall 2005 UCF

Canny Edge Detector
Hysteresis Thresholding

- Scan the image from left to right, top-bottom.
  - The gradient magnitude at a pixel is above a high threshold declare that as an edge point
  - Then recursively consider the *neighbors* of this pixel.
    - If the gradient magnitude is above the low threshold declare that as an edge pixel.

Alper Yilmaz, Fall 2005 UCF

Canny Edge Detector
Hysteresis Thresholding

M

regular

$M \geq 25$

Hysteresis

$High = 35$

$Low = 15$

Homework

1. Derive Laplacian of Gaussian
2. Drive gradient of 2D Gaussian
3. Show that gradient magnitude is rotation invariant.

$$g = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

- Due date 28 September 2005

Programming Project 1

- Implement Canny Edge Detector
- Deliverables
  - Short report, problems, results, program code, etc.
  - Step by step outputs of images
  - Program code
    - Program should ask for a PGM image from user
    - Ask for the threshold value, sigma of Gaussian
    - Write out or display the image
- Due Date 17 October 2005

Suggested Reading

- Chapter 4, Emanuele Trucco, Alessandro Verri, "Introductory Techniques for 3-D Computer Vision"
- Chapter 2, Mubarak Shah, “Fundamentals of Computer Vision”