Edge Detection

Ref: Forsyth+Ponce Ch. 7,8
Trucco+Verri Ch. 4
• Our goal is to extract a “line drawing” representation from an image

• Useful for recognition: edges contain shape information
  – invariance
Edge detection

\[ f(x) \]

\[ \frac{df(x)}{dx} \approx \text{edges} \]
Derivatives

• Edges are locations with high image gradient or derivative*
• Estimate derivative using finite difference

$$\frac{\partial}{\partial x} I(x_0, y_0) \approx I(x_0 + 1, y_0) - I(x_0, y_0)$$

• Problem?
Smoothing

- Reduce image noise by smoothing with a Gaussian

\[ J = G \ast I \]

\[ J(x, y) = \sum_{u,v} G(x - u, y - v)I(u, v) \]

\[ G(x, y) = \frac{1}{2\pi\sigma^2}e^{-\left(x^2+y^2\right)/2\sigma^2} \]
Convolution is Associative

• We compute derivative of smoothed image:

\[ J_x = \frac{\partial J}{\partial x} = K_{\partial/\partial x} \ast (G \ast I) \]

• Since convolution is associative:

\[ \frac{\partial J}{\partial x} = \frac{\partial G}{\partial x} \ast I \]
Separable Convolution

• Note that $G$ can be factored as

$$G(x, y) = \frac{1}{2\pi \sigma^2} \left( e^{-\frac{x^2}{2\sigma^2}} \right) \left( e^{-\frac{y^2}{2\sigma^2}} \right)$$

and computed as two 1-D convolutions
Edge orientation

• Would like gradients in all directions
• Approximate:
  – Compute smoothed derivatives in x,y directions
  – Edge strength
    \[ e_s(i, j) = \sqrt{J_x^2(i, j) + J_y^2(i, j)} \]
  – Edge normal
    \[ e_o(i, j) = \text{atan} \frac{J_y}{J_x} \]
Canny Edge Detection

• Compute edge strength and orientation at all pixels

• “Non-max suppression”
  – Reduce thick edge strength responses around true edges

• Link and threshold using “hysteresis”
  – Simple method of “contour completion”
Non-maximum suppression:
Select the single maximum point across the width of an edge.
Non-maximum suppression

At q, the value must be larger than values interpolated at p or r.
Examples:
Non-Maximum Suppression

Original image  Gradient magnitude  Non-maxima suppressed

Slide credit: Christopher Rasmussen
fine scale
high threshold
coarse scale, high threshold
coarse scale low threshold
Linking to the next edge point

Assume the marked point is an edge point.

Take the normal to the gradient at that point and use this to predict continuation points (either $r$ or $s$).
Edge Hysteresis

- **Hysteresis**: A lag or momentum factor
- **Idea**: Maintain two thresholds $k_{\text{high}}$ and $k_{\text{low}}$
  - Use $k_{\text{high}}$ to find strong edges to start edge chain
  - Use $k_{\text{low}}$ to find weak edges which continue edge chain
- **Typical ratio of thresholds** is roughly $k_{\text{high}} / k_{\text{low}} = 2$
Example: Canny Edge Detection

gap is gone

Original image

Strong edges only

Strong + connected weak edges

Weak edges

courtesy of G. Loy
Problem?

• Texture
  – Canny edge detection responds all over textured regions