Lecture 5: Gradients and Edge Detection
Reading: T&V Section 4.1 and 4.2

What Are Edges?
Simple answer: discontinuities in intensity.

Boundaries of objects

Boundaries of Material Properties

Boundaries of Lighting

Types of Edges (1D Profiles)
- Edges can be modeled according to their intensity profiles:
  - **Step edge:**
    - the image intensity abruptly changes from one value to one side of the discontinuity to a different value on the opposite side.
  - **Ramp edge:**
    - a step edge where the intensity change is not instantaneous but occurs over a finite distance.
Types of Edges (1D Profiles)

- Ridge edge:
  - the image intensity abruptly changes value but then returns to the starting value within some short distance
  - generated usually by lines

- Roof edge:
  - a ridge edge where the intensity change is not instantaneous but occurs over a finite distance
  - generated usually by the intersection of surfaces
Step/Ramp Edge Terminology

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

Important point: All of this information can be computed from the gradient vector field!!

Summary of Gradients

Gradient Vector: \( \nabla I = \left[ \frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right] \)

Magnitude: \( |\nabla I| = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2} \)

Orientation: \( \theta = \arctan2(\frac{\partial I}{\partial y}, \frac{\partial I}{\partial x}) \)

Simple Edge Detection Using Gradients

A simple edge detector using gradient magnitude

- Compute gradient vector at each pixel by convolving image with horizontal and vertical derivative filters
- Compute gradient magnitude at each pixel
- If magnitude at a pixel exceeds a threshold, report a possible edge point.

Compute Spatial Image Gradients

\[ \frac{I(x+1,y) - I(x-1,y)}{2} \]

\[ \frac{I(x,y+1) - I(x,y-1)}{2} \]

Partial derivative with respect to \( x \)

Partial derivative with respect to \( y \)

Replace with your favorite smoothing+derivative operator

Simple Edge Detection Using Gradients

A simple edge detector using gradient magnitude

- Compute gradient vector at each pixel by convolving image with horizontal and vertical derivative filters
- Compute gradient magnitude at each pixel
- If magnitude at a pixel exceeds a threshold, report a possible edge point.

Compute Gradient Magnitude

Magnitude of gradient \( \sqrt{I_x^2 + I_y^2} \)

Measures steepness of slope at each pixel (= edge contrast)
Simple Edge Detection Using Gradients

A simple edge detector using gradient magnitude

- Compute gradient vector at each pixel by convolving image with horizontal and vertical derivative filters
- Compute gradient magnitude at each pixel
- If magnitude at a pixel exceeds a threshold, report a possible edge point.

Threshold to Find Edge Pixels

- Example – cont.:

Edge Detection Using Gradient Magnitude

Edge thinning and linking

There is ALWAYS a tradeoff between smoothing and good edge localizations!

Issues to Address

How should we choose the threshold?

> 10  > 30  > 80

Smoothing + thresholding gives us a binary mask with “thick” edges

We want thin, one-pixel wide, connected contours
Canny Edge Detector

An important case study

Probably, the most used edge detection algorithm by C.V. practitioners

Experiments consistently show that it performs very well


Formal Design of an Optimal Edge Detector

- Edge detection involves 3 steps:
  - Noise smoothing
  - Edge enhancement
  - Edge localization
- J. Canny formalized these steps to design an *optimal* edge detector

Edge Model (1D)

- An ideal edge can be modeled as an step function:
  \[
  G(x) = \begin{cases} 
  0 & \text{if } x < 0 \\
  A & \text{if } x \leq 0 
  \end{cases}
  \]
- Additive, White Gaussian Noise
  - RMS noise amplitude/unit length \( n_o^2 \)

Performance Criteria (1)

- Good detection
  - The filter must have a stronger response at the edge location (\( x=0 \)) than to noise

Performance Criteria (2)

- Good Localization
  - The filter response must be maximum very close to \( x=0 \)

Performance Criteria (3)

- Low False Positives
  - There should be only one maximum in a reasonable neighborhood of \( x=0 \)
Canny Edge Detector

- Canny found a linear, continuous filter that maximized the three given criteria.
- There is no closed-form solution for the optimal filter.
- However, it looks VERY SIMILAR to the derivative of a Gaussian.

Recall: Practical Issues for Edge Detection

- Thinning and linking
- Choosing a magnitude threshold

Canny has good answers to all!

Thinning

- We want to mark points along curve where the magnitude is largest.
- We can do this by looking for a maximum along a 1D intensity slice normal to the curve (non-maximum supression).
- These points should form a one-pixel wide curve.

Which Threshold to Pick?

- Two thresholds applied to gradient magnitude
  - If the threshold is too high:
    - Very few (none) edges
    - High MISDETECTIONS, many gaps
  - If the threshold is too low:
    - Too many (all pixels) edges
    - High FALSE POSITIVES, many extra edges

SOLUTION: Hysteresis Thresholding

- Allows us to apply both! (e.g. a “fuzzy” threshold)
- Keep both a high threshold $H$ and a low threshold $L$.
- Any edges with strength $< L$ are discarded.
- Any edge with strength $> H$ are kept.
- An edge $P$ with strength between $L$ and $H$ is kept only if there is a path of edges with strength $> L$ connecting $P$ to an edge of strength $> H$.
- In practice, this thresholding is combined with edge linking to get connected contours
Complete Canny Algorithm

1. Compute x and y derivatives of image
   \[ I_x = G_x * I \quad I_y = G_y * I \]
2. Compute magnitude of gradient at every point
   \[ M(x, y) = |I_x|^2 + |I_y|^2 \]
3. Eliminate those pixels that are not local maxima of the magnitude in the direction of the gradient.
4. Hysteresis Thresholding
   - Select the pixels such that \( M > T_1 \) (high threshold)
   - Collect the pixels such that \( M > T_2 \) (low threshold) that are neighbors of already connected edge points.

See textbook for more details.