CS4670/5670: Computer Vision
Kavita Bala

Lecture 3: Edge detection

From Sandlot Science
Announcements

• Find partners on piazza

• PA 1 will be out on Monday

• Quiz on Monday or Wednesday, beginning of class
Why edges?

- Humans are sensitive to edges
- Convert a 2D image into a set of curves
  - Extracts salient features of the scene, more compact
Origin of Edges

• Edges are caused by a variety of factors

- surface normal discontinuity
- depth discontinuity
- surface color discontinuity
- illumination discontinuity
Images as functions...

• Edges look like steep cliffs
Characterizing edges

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

Source: L. Lazebnik
Image derivatives

• How can we differentiate a digital image $F[x, y]$?
  – Option 1: reconstruct a continuous image, $f$, then compute the derivative
  – Option 2: take discrete derivative (finite difference)

$$\frac{\partial f}{\partial x}[x, y] \approx F[x + 1, y] - F[x, y]$$

How would you implement this as a linear filter?

\[
\begin{align*}
\frac{\partial f}{\partial x} & : H_x \\
\frac{\partial f}{\partial y} & : H_y
\end{align*}
\]

Source: S. Seitz
The gradient of an image: \( \nabla f = \left[ \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right] \)

The gradient points in the direction of most rapid increase in intensity.

\( \nabla f = [\frac{\partial f}{\partial x}, 0] \)

\( \nabla f = [0, \frac{\partial f}{\partial y}] \)

The edge strength is given by the gradient magnitude:

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

The gradient direction is given by:

\[ \theta = \tan^{-1} \left( \frac{\frac{\partial f}{\partial y}}{\frac{\partial f}{\partial x}} \right) \]

- how does this relate to the direction of the edge?

Source: Steve Seitz
Image gradient

Source: L. Lazebnik
Effects of noise

Noisy input image

\[ f(x) \]

\[ \frac{d}{dx} f(x) \]

Where is the edge?

Source: S. Seitz
Solution: smooth first

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

Source: S. Seitz
Image with Edge

Edge Location

Image + Noise

Derivatives detect edge \textit{and} noise

Smoothed derivative removes noise, but blurs edge
Associative property of convolution

- Differentiation is a convolution
- Convolution is associative:
  \[ \frac{d}{dx}(f \ast h) = f \ast \frac{d}{dx}h \]
- This saves us one operation:

![Graph](image)
2D edge detection filters

Gaussian

\[ h_\sigma(u, v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2\sigma^2}} \]

derivative of Gaussian \((x)\)

\[ \frac{\partial}{\partial x} h_\sigma(u, v) \]
\[ \nabla G_\sigma(x) = \left( \frac{\partial G_\sigma}{\partial x}, \frac{\partial G_\sigma}{\partial y} \right)(x) = [-x, -y] \exp \left( - \frac{x^2 + y^2}{2\sigma^2} \right) \]

\[ \nabla^2 G_\sigma(x) = \frac{1}{\sigma^3} \left( 2 - \frac{x^2 + y^2}{2\sigma^2} \right) \exp \left( - \frac{x^2 + y^2}{2\sigma^2} \right) \]
Derivative of Gaussian filter

x-direction

y-direction
FIGURE 5.3: The scale (i.e., $\sigma$) of the Gaussian used in a derivative of Gaussian filter has significant effects on the results. The three images show estimates of the derivative in the $x$ direction of an image of the head of a zebra obtained using a derivative of Gaussian filter with $\sigma$ one pixel, three pixels, and seven pixels (left to right). Note how images at a finer scale show some hair, the animal’s whiskers disappear at a medium scale, and the fine stripes at the top of the muzzle disappear at the coarser scale.
FIGURE 5.4: The gradient magnitude can be estimated by smoothing an image and then differentiating it. This is equivalent to convolving with the derivative of a smoothing kernel. The extent of the smoothing affects the gradient magnitude; in this figure, we show the gradient magnitude for the figure of a zebra at different scales. At the center, gradient magnitude estimated using the derivatives of a Gaussian with $\sigma = 1$ pixel; and on the right, gradient magnitude estimated using the derivatives of a Gaussian with $\sigma = 2$ pixel. Notice that large values of the gradient magnitude form thick trails.
The Sobel operator

- Common approximation of derivative of Gaussian
  - A mask (not a convolution kernel)

\[
\begin{array}{ccc}
\frac{1}{8} & \begin{array}{ccc}
-1 & 0 & 1 \\
-2 & 0 & 2 \\
-1 & 0 & 1 \\
\end{array} & \begin{array}{ccc}
\frac{1}{8} & \begin{array}{ccc}
1 & 2 & 1 \\
0 & 0 & 0 \\
-1 & -2 & -1 \\
\end{array} & \begin{array}{ccc}
1 & 2 & 1 \\
0 & 0 & 0 \\
-1 & -2 & -1 \\
\end{array}
\end{array}
\]

- The standard defn. of the Sobel operator omits the 1/8 term
  - doesn’t make a difference for edge detection
  - the 1/8 term **is** needed to get the right gradient magnitude
Sobel operator: example

Questions?
CS4670: Computer Vision

Image Resampling
This image is too big to fit on the screen. How can we generate a half-sized version?

Source: S. Seitz
Image sub-sampling

Throw away every other row and column to create a 1/2 size image - called image sub-sampling

Source: S. Seitz
Why does this look so crufty?
Image sub-sampling

Source: F. Durand
Even worse for synthetic images

Source: L. Zhang
What is aliasing?

- What if we “missed” things between the samples?
- Simple example: undersampling a sine wave
  - unsurprising result: information is lost
  - surprising result: indistinguishable from lower frequency
  - also was always indistinguishable from higher frequencies
  - aliasing: signals “traveling in disguise” as other frequencies
Wagon-wheel effect

Imagine a spoked wheel moving to the right (rotating clockwise). Mark wheel with dot so we can see what’s happening.

If camera shutter is only open for a fraction of a frame time (frame time = 1/30 sec. for video, 1/24 sec. for film):

Without dot, wheel appears to be rotating slowly backwards! (counterclockwise)

(See http://www.michaelbach.de/ot/mot_wagonWheel/index.html)

Source: L. Zhang
**Aliasing**

- Occurs when your sampling rate is not high enough to capture the amount of detail in your image
- Can give you the wrong signal/image—an alias
- To do sampling right, need to understand the structure of your signal/image
- Enter Monsieur Fourier...
- To avoid aliasing:
  - sampling rate ≥ 2 * max frequency in the image
    - said another way: ≥ two samples per cycle
  - This minimum sampling rate is called the **Nyquist rate**

Source: L. Zhang
Nyquist limit – 2D example

Good sampling

Bad sampling
Aliasing

• When downsampling by a factor of two
  – Original image has frequencies that are too high

• How can we fix this?
Gaussian pre-filtering

Solution: filter the image, *then* subsample
Subsampling with Gaussian pre-filtering

• Solution: filter the image, then subsample

Source: S. Seitz
Compare with...

1/2

1/4 (2x zoom)

1/8 (4x zoom)

Source: S. Seitz
Gaussian pre-filtering

- Solution: filter the image, *then* subsample
Gaussian pyramid

{blur subsample blur subsample ...}

\(F_0\)

\(F_0^*H\)

\(F_1\)

\(F_1^*H\)

\(F_2\)
Gaussian pyramids
[Burt and Adelson, 1983]

In computer graphics, a *mip map* [Williams, 1983]
A precursor to *wavelet transform*

Gaussian Pyramids have all sorts of applications in computer vision

Source: S. Seitz
Gaussian pyramids
[Burt and Adelson, 1983]

How much space does a Gaussian pyramid take compared to the original image?

Source: S. Seitz
Gaussian Pyramid
Questions?