EECS 442 – Computer vision

Detectors part I

• Edge feature detectors
• Corner feature detectors

Reading:  [FP] Chapters: 8,9

Some slides of this lectures are courtesy of prof  F. Li, prof  S. Lazebnik, and various other lecturers
Goal:
Identify interesting regions from the images (edges, corners, blobs…)

Descriptors

Matching / Indexing / Recognition
e.g. SIFT
Linear filtering

• Convolution:

$$(f * g)[m, n] = \sum_{k,l} f[k, l] g[m-k, n-l]$$

• Smoothing

• Differentiation
Smoothing with a Gaussian

- Weight contributions of neighboring pixels by nearness

\[
G_\sigma = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}
\]

Slide credit: Christopher Rasmussen
Smoothing with a Gaussian
Differentiation and convolution

\[ \frac{\partial f}{\partial x} \approx \frac{f(x_{n+1}, y) - f(x_n, y)}{\Delta x} \]

<table>
<thead>
<tr>
<th>Original Image</th>
<th>2D Kernel</th>
<th>2D Kernel</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>-2 -1 0 1 2</td>
<td>-2 -1 0 1 2</td>
</tr>
<tr>
<td>E</td>
<td>-2 -1 0 1 2</td>
<td>-2 -1 0 1 2</td>
</tr>
<tr>
<td>R</td>
<td>-2 -1 0 1 2</td>
<td>-2 -1 0 1 2</td>
</tr>
</tbody>
</table>

Rudimentary edge detector!
Edge detection
What causes an edge?

Identifies sudden changes in an image

- Depth discontinuity
- Surface orientation discontinuity
- Reflectance discontinuity (i.e., change in surface material properties)
- Illumination discontinuity (e.g., shadow)
Edge Detection

• Criteria for **optimal edge detection** (Canny 86):
  
  – **Good detection:**
    • minimize the probability of false positives (detecting spurious edges caused by noise),
    • false negatives (missing real edges)
  
  – **Good localization:**
    • edges must be detected as close as possible to the true edges.
  
  – **Single response constraint:**
    • minimize the number of local maxima around the true edge (i.e. detector must return single point for each true edge point)
Edge Detection

• Examples:

True edge

Poor robustness to noise

Poor localization

Too many responses
Designing an edge detector

• **Edge**: a location with high gradient (thus, use derivatives!)

• Need two derivatives, in x and y direction.

• Need **smoothing** to reduce noise prior to taking derivative
\( f \ast g \)

\( \frac{d}{dx} (f \ast g) \)

Source: S. Seitz
Edge by Derivative of Gaussian

• We can use derivative of Gaussian filters

Why?

• Gaussian filter is needed for smoothing the image
• Differentiation can be modeled by a convolution
• Convolution is associative:

\[ D \ast (G \ast I) = (D \ast G) \ast I \]
Edge by Derivative of Gaussian

\[ \frac{\partial}{\partial x} G_\sigma \]

\[ \frac{\partial}{\partial y} G_\sigma \]
Canny Edge Detection

- Most widely used edge detector in computer vision.
- First derivative of the Gaussian closely approximates the operator that optimizes the product of signal-to-noise ratio and localization.
- Analysis based on "step-edges" corrupted by "additive Gaussian noise".
Canny Edge Detection

Steps:

1. Gaussian smoothing
2. & Derivative = Derivative of Gaussian
3. Find magnitude and orientation of gradient
4. Extract edge points: ‘Non-maximum suppression’
5. Linking and thresholding ‘Hysteresis’:

   • Matlab: `edge(I, 'canny')`
Canny Edge Detector
First 2 Steps

• Smoothing

\[ I' = g(x, y) * I \]

\[ g(x, y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2 + y^2}{2\sigma^2}} \]

• Derivative

\[ S = \nabla(g * I) = (\nabla g) * I = \]

\[ \begin{bmatrix} g_x \\ g_y \end{bmatrix} * I = \begin{bmatrix} g_x * I \\ g_y * I \end{bmatrix} \]

\[ \nabla g = \begin{bmatrix} \frac{\partial g}{\partial x} \\ \frac{\partial g}{\partial y} \end{bmatrix} = \begin{bmatrix} g_x \\ g_y \end{bmatrix} \]
Canny Edge Detector
Derivative of Gaussian
Canny Edge Detector
First 2 Steps

\[ S = \nabla(g * I) = (\nabla g) * I \]

\[ S = \begin{bmatrix} S_x & S_y \end{bmatrix} = \text{gradient vector} \]
Canny Edge Detector
Third Step

• magnitude and direction of \( S = \begin{bmatrix} S_x & S_y \end{bmatrix} \)

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

direction = \( \theta = \tan^{-1} \frac{S_y}{S_x} \)
Canny Edge Detector - Fourth Step

• Non maximum suppression

• Slice gradient magnitude along the gradient direction
• Mark the point along the slide where the magnitude is max
Non-maximum suppression

Gradient
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).
Examples: Non-Maximum Suppression

Original image  Gradient magnitude  Non-maxima suppressed

Slide credit: Christopher Rasmussen
Canny Edge Detector
Step 5: Thresholding

- Set a threshold $T$ to suppress gradients with magnitude $< T$
high threshold (strong edges)

low threshold (weak edges)
Canny Edge Detector
Step 5: Hysteresis Thresholding

- **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 along the edge chain
- Typical ratio of thresholds is roughly
  \[ \frac{k_{\text{high}}}{k_{\text{low}}} = 2 \]
hysteresis threshold
Effect of $\sigma$ (Gaussian kernel spread/size)

- The choice of $\sigma$ depends on desired behavior
  - large $\sigma$ detects large scale edges
  - small $\sigma$ detects fine features

Source: S. Seitz
Demo

http://www.cs.washington.edu/research/imagedatabase/demo/edge/
Other edge detectors:

- Sobel
- Canny-Deriche
- Differential
Extract useful building blocks: Corners
Extract useful building blocks: blobs
• **Repeatability**
  – The same feature can be found in several images despite geometric and photometric transformations

• **Saliency**
  – Each feature is found at an “interesting” region of the image

• **Locality**
  – A feature occupies a “relatively small” area of the image;
Repeatability

Illumination invariance

Scale invariance

Pose invariance
- Rotation
- Affine
• Saliency

• Locality
Harris corner detector

Harris Detector: Basic Idea

“flat” region: no change in all directions

“edge”: no change along the edge direction

“corner”: significant change in all directions
Harris Detector: Mathematics

Change of intensity for the shift \([u, v]\):

\[
E(u, v) = \sum_{x,y} w(x, y) \left[ I(x + u, y + v) - I(x, y) \right]^2
\]

Window function

Shifted intensity

Intensity

Window function \(w(x, y) =

1\text{ in window, } 0\text{ outside}

or

Gaussian
Harris Detector: Mathematics

For small shifts \([u, v]\) we have a bilinear approximation:

\[
E(u, v) \approx [u, v] \ M \begin{bmatrix} u \\ v \end{bmatrix}
\]

where \(M\) is a 2×2 matrix computed from image derivatives:

\[
M = \sum_{x, y} \ w(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix}
\]
Second-moment matrix

Sum over a small region around the hypothetical corner

$$M = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix}$$

Matrix is symmetric

Gradient with respect to x, times gradient with respect to y

$$\left( g_x \ast I \right) \left( g_y \ast I \right)$$

Slide credit: David Jacobs
Second-moment matrix

First, consider case where dominant gradient directions aligned with x or y

\[
M = \begin{bmatrix}
\sum I_x^2 & \sum I_x I_y \\
\sum I_x I_y & \sum I_y^2 \\
\end{bmatrix} = \begin{bmatrix}
\lambda_1 & 0 \\
0 & \lambda_2 \\
\end{bmatrix}
\]
First, consider case where dominant gradient directions aligned with x or y

\[
M = \begin{bmatrix}
\sum I_x^2 & \sum I_x I_y \\
\sum I_x I_y & \sum I_y^2
\end{bmatrix} = \begin{bmatrix}
\lambda_1 & 0 \\
0 & \lambda_2
\end{bmatrix}
\]

If either \( \lambda \) is close to 0, then this is an edge
Second-moment matrix

First, consider case where dominant gradient directions aligned with x or y

\[ M = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \]

If both \( \lambda \)s are close to 0, then this is a flat region
Second-moment matrix

First, consider case where dominant gradient directions aligned with x or y

\[
M = \begin{bmatrix}
\sum I_x^2 & \sum I_x I_y \\
\sum I_x I_y & \sum I_y^2 \\
\end{bmatrix} = U^{-1} \begin{bmatrix}
\lambda_1 & 0 \\
0 & \lambda_2 \\
\end{bmatrix} U
\]

Lambda 1, 2 are the eigenvalues of C
Classification of image points using eigenvalues of $M$:

- $\lambda_1$ and $\lambda_2$ are small; $E$ is almost constant in all directions
- $\lambda_1$ and $\lambda_2$ are large, $\lambda_1 \sim \lambda_2$; $E$ increases in all directions
- $\lambda_1 \gg \lambda_2$
Harris Detector: Mathematics

Measure of corner response:

\[ R = \det M - k \left( \text{trace } M \right)^2 \]

\[ \det M = \lambda_1 \lambda_2 \]

\[ \text{trace } M = \lambda_1 + \lambda_2 \]

\( (k \text{ – empirical constant, } k = 0.04-0.06) \)
Harris Detector: Mathematics

- $R$ depends only on eigenvalues of $M$
- $R$ is large for a corner
- $R$ is negative with large magnitude for an edge
- $|R|$ is small for a flat region
Harris Detector: Algorithm

- Filter image with Gaussian to reduce noise
- Compute magnitude of the x and y gradients at each pixel
- Construct $M$ in a window around each pixel (Harris uses a Gaussian window)
- Compute $\lambda$s of $M$
- Compute $R = \det M - k (\text{trace } M)^2$
- If $R > T$ a corner is detected; retain point of local maxima
Harris Detector: Workflow
Harris Detector: Workflow

Compute corner response $R$
Harris Detector: Workflow
Find points with large corner response: $R > \text{threshold}$
Harris Detector: Workflow

Take only the points of local maxima of $R$
Harris Detector: Workflow
Harris Detector: Some Properties

- Rotation invariance

Corner response $R$ is invariant to image rotation

$$C = U^{-1} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} U \quad \rightarrow \quad R = R(\lambda_1, \lambda_2) \text{ doesn’t change!}$$
Harris Detector: Some Properties

• But: non-invariant to *image scale*!

All points will be classified as *edges*

Corner!
Harris Detector: Some Properties

- Partial invariance to affine intensity change
  - Invariance to intensity shift $I \rightarrow I + b$
    (only derivatives are used)
  - Intensity scale: $I \rightarrow a \cdot I$

---

Threshold

$x$ (image coordinate)

---

$x$ (image coordinate)
Next lecture:

Detectors part ii
Descriptors