CS376 Computer Vision
Lecture 12: Invariant Features

Qixing Huang
March 4th 2019
Roadmap of This Class

• Image Filters
  – Smoothing
  – Canny edge detector
  – Binary image analysis
  – Texture

• Grouping/Fitting/Segmentation
  – Hough transform/RANSAC
  – K-means
  – Graph-cut
Now: Multiple views

Matching, invariant features, stereo vision, instance recognition

Slide credit: Kristen Grauman
Important tool for multiple views: Local features

Multi-view matching relies on **local feature** correspondences.

How to detect *which local features* to match?
Local features: main components

1) Detection: Identify the interest points

2) Description: Extract vector feature descriptor surrounding each interest point.

3) Matching: Determine correspondence between descriptors in two views

\[
\mathbf{x}_1 = [x_1^{(1)}, \ldots, x_d^{(1)}] \\
\mathbf{x}_2 = [x_1^{(2)}, \ldots, x_d^{(2)}]
\]

Slide credit: Kristen Grauman
Local features: desired properties

• Invariance
  – Can be detected despite geometric and photometric transformations

• Saliency
  – Each feature has a distinctive description/Few potential matches on other images

• Compactness and efficiency
  – Only a few salient features from each image

• Locality
  – A feature occupies a relatively small area of the image; robust to clutter and occlusion
Goal: Invariance

• We want to detect (at least some of) the same points in both images.

No chance to find true matches!

• Yet we have to be able to run the detection procedure independently per image.

Slide credit: Kristen Grauman
Goal: descriptor distinctiveness

- We want to be able to reliably determine which point goes with which.

- Must provide some invariance to geometric and photometric differences between the two views.

Slide credit: Kristen Grauman
Local features: main components

1) Detection: Identify the interest points

2) Description: Extract vector feature descriptor surrounding each interest point.

3) Matching: Determine correspondence between descriptors in two views
• What points would you choose?
Detecting corners

Slide credit: Kristen Grauman
Detecting corners

- Compute “cornerness” response at every pixel.
Detecting corners

Slide credit: Kristen Grauman
Detecting local invariant features

• Detection of interest points
  – Harris corner detection
  – Scale invariant blob detection: LoG
• (Next lecture: description of local patches)
• Not all invariant features are corners --- there is a tradeoff between how many invariant features we detect from each image and the computational cost
Optical Flow Review
Translational model

- Invariance assumption

\[ I_1(x_1) = I_2(h(x_1)) = I_2(x_1 + u) \]

Make it continuous, like a video

\[ I(x(t), t) = I(x(t) + u(t), t + dt) \]

- Image brightness constant constraint:

Derivative computation:

\[ \nabla I(x(t), t)^T u + I_t(x(t), t) = 0 \]

where

\[ \nabla I(x, t) \doteq \left[ \frac{\partial I}{\partial x}(x, t) \right] \], \quad \text{and} \quad I_t(x, t) \doteq \frac{\partial I}{\partial t}(x, t). \]
Optical flow and the aperture problem

• Simplified notation
  \[ \nabla I^T u + I_t = 0 \]

• Eulerian view:
  – Fix our attention at a particular image location and compute the velocity of “particles flowing” through that pixel
  – \( u \) is called a optical flow

• Lagrangian view:
  – Fix our attention at a particular particle \( x(t) \)
  – This is called feature tracking
Aperture problem

- A single constraint does not uniquely specify the motion
  - We cannot differentiate diagonal motion and horizontal motion
Local constancy

• Motion is the same for all points in a window $W(x)$
• This is equivalent to assuming a purely translational deformation model:

$$h(x) = x + u(\lambda x) = x + u(\lambda x) \text{ for all } x \in W(x)$$

Optimization formulation

$$E_b(u) = \sum_{W(x,y)} (\nabla I^T(x, y, t) \cdot u(x, y) + I_t(x, y, t))^2$$

Least square solution

$$\begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} u + \begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix} = 0$$
M may be degenerate

- The intensity variation in a local image window varies only along one dimension or vanishes

\[ I_x = 0 \quad \text{and/or} \quad I_y = 0 \]
\[ M = \sum \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \]
What does this matrix reveal?

First, consider an axis-aligned corner:

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

This means dominant gradient directions align with x or y axis.

Look for locations where both \( \lambda \)'s are large.

If either \( \lambda \) is close to 0, then this is not corner-like.

What if we have a corner that is not aligned with the image axes?
What does this matrix reveal?

Since $M$ is symmetric, we have

$$M = X \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} X^T$$

$$Mx_i = \lambda_i x_i$$

The *eigenvalues* of $M$ reveal the amount of intensity change in the two principal orthogonal gradient directions in the window.
Corner response function

“edge”: \[ \lambda_1 \gg \lambda_2 \]
\[ \lambda_2 \gg \lambda_1 \]

“corner”: \[ \lambda_1 \text{ and } \lambda_2 \text{ are large,} \]
\[ \lambda_1 \sim \lambda_2; \]

“flat” region \[ \lambda_1 \text{ and } \lambda_2 \text{ are small,} \]

Cornerness score (other variants possible)

\[ f = \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2} \]
Harris corner detector

1) Compute $M$ matrix for each image window to get their *cornerness* scores.
2) Find points whose surrounding window gave large corner response ($f >$ threshold)
3) Take the points of local maxima, i.e., perform non-maximum suppression
Harris Detector: Steps
Harris Detector: Steps

Compute corner response $f$
Harris Detector: Steps

Find points with large corner response: $f > \text{threshold}$
Harris Detector: Steps

Take only the points of local maxima of $f$
Harris Detector: Steps
Properties of the Harris corner detector

• Rotation invariant? Yes

\[
M = X \begin{bmatrix}
\lambda_1 & 0 \\
0 & \lambda_2
\end{bmatrix} X^T
\]

• Scale invariant?
Properties of the Harris corner detector

- Rotation invariant? Yes

- Scale invariant? No

All points will be classified as edges.
Scale invariant interest points

How can we independently select interest points in each image, such that the detections are repeatable across different scales?
Automatic Scale Selection

How to find corresponding patch sizes, with only one image in hand?
Automatic scale selection

Intuition:
- Find scale that gives local maxima of some function $f$ in both position and scale.

![Diagram showing scale selection](image)
Automatic Scale Selection

- Function responses for increasing scale (scale signature)

\[ f(I_{i_1 \ldots i_m}(x, \sigma)) \]

\[ f(I_{i_1 \ldots i_m}(x', \sigma)) \]
Automatic Scale Selection

• Function responses for increasing scale (scale signature)

$$f(I_{i_1, i_m}(x, \sigma))$$

$$f(I_{i_1, i_m}(x', \sigma))$$
Automatic Scale Selection

- Function responses for increasing scale (scale signature)
Automatic Scale Selection

- Function responses for increasing scale (scale signature)

\[ f(I_{i_1...i_m}(x, \sigma)) \]

\[ f(I_{i_1...i_m}(x', \sigma)) \]
Automatic Scale Selection

- Function responses for increasing scale (scale signature)

\[ f(I_{i \ldots j} (x, \sigma)) \]

\[ f(I_{i \ldots j} (x', \sigma)) \]
Automatic Scale Selection

- Function responses for increasing scale (scale signature)
• What can be the “signature” function?
Blob detection in 2D

- Laplacian of Gaussian: Circularly symmetric operator for blob detection in 2D

\[ \nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2} \]
Blob detection in 2D: scale selection

- Laplacian-of-Gaussian = “blob” detector

\[ \nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2} \]
Blob detection in 2D

• We define the *characteristic scale* as the scale that produces peak of Laplacian response
Example

Original image at \( \frac{3}{4} \) the size

Slide credit: Kristen Grauman
Original image at 3/4 the size

Slide credit: Kristen Grauman
Scale invariant interest points

Interest points are local maxima in both position and scale.

\[ L_{xx}(\sigma) + L_{yy}(\sigma) \]

Squared filter response maps

\[ \Rightarrow \text{List of } (x, y, \sigma) \]

Slide credit: Kristen Grauman
Scale-space blob detector: Example

original image

scale-space maxima of \((\nabla^2_{\text{norm}} L)^2\)

Scale-space blob detector: Example

Image credit: Lana Lazebnik
Technical detail

• We can approximate the Laplacian with a difference of Gaussians; more efficient to implement.

\[ L = \sigma^2 \left( G_{xx}(x, y, \sigma) + G_{yy}(x, y, \sigma) \right) \]

(Laplacian)

\[ \text{DoG} = G(x, y, k\sigma) - G(x, y, \sigma) \]

(Difference of Gaussians)
Further Reading – Scale-Space
Summary

• Desirable properties for local features for correspondence
• Basic matching pipeline
• Interest point detection
  – Harris corner detector
  – Laplacian of Gaussian, automatic scale selection