CS5670: Computer Vision
Feature invariance
Reading

- Szeliski: 4.1
Announcements

• Project 1 code due tonight at 11:59pm
• Project 1 artifact due Wednesday, 2/10, at 11:59pm
• Quiz 1 in class this Wednesday, 2/10 (first 10 minutes of class)
  – Closed book / closed note
• Project 2 (Feature Detection & Matching) will be out next week
  – To be done in groups of 2
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

\[ x_1 = [x_1^{(1)}, \ldots, x_d^{(1)}] \]

\[ x_2 = [x_1^{(2)}, \ldots, x_d^{(2)}] \]
Harris features (in red)
Image transformations

• Geometric

Rotation

Scale

• Photometric
  Intensity change
Invariance and equivariance

- We want corner locations to be *invariant* to photometric transformations and *equivariant* to geometric transformations
  
  - **Invariance**: image is transformed and corner locations do not change
  
  - **Equivariance**: if we have two transformed versions of the same image, features should be detected in corresponding locations

- (Sometimes “invariant” and “equivariant” are both referred to as “invariant”)
- (Sometimes “equivariant” is called “covariant”)
Harris detector invariance properties:

image translation

- Derivatives and window function are equivariant

Corner location is equivariant w.r.t. translation
Harris detector invariance properties: image rotation

Second moment ellipse rotates but its shape (i.e. eigenvalues) remains the same

Corner location is equivariant w.r.t. image rotation
Harris detector invariance properties:
Affine intensity change

- Only derivatives are used → invariance to intensity shift $I \rightarrow I + b$
- Intensity scaling: $I \rightarrow a \ I$

*Partially invariant to affine intensity change*
Harris detector invariance properties: scaling

Neither invariant nor equivariant to scaling
Scale invariant detection

Suppose you’re looking for corners

Key idea: find scale that gives local maximum of $f$
- in both position and scale
- One definition of $f$: the Harris operator
Automatic scale selection

Function responses for increasing scale:
Scale trace (signature)

Lindeberg et al., 1996
Automatic scale selection

Function responses for increasing scale
Scale trace (signature)
Automatic scale selection

Function responses for increasing scale
Scale trace (signature)
Automatic scale selection

Function responses for increasing scale

Scale-trace (signature)
Automatic scale selection

Function responses for increasing scale

Scale trace (signature)
Automatic scale selection

Function responses for increasing scale
Scale trace (signature)
Automatic scale selection

Function responses for increasing scale
Scale trace (signature)

\[ f(l_{i...m}(x, \sigma)) \]

\[ f(l_{i...m}(x', \sigma')) \]
Automatic scale selection

Normalize: rescale to fixed size
Implementation

• Instead of computing $f$ for larger and larger windows, we can implement using a fixed window size with a Gaussian pyramid

(sometimes need to create in-between levels, e.g. a ¾-size image)
Feature extraction: Corners and blobs
Another common definition of $f$

- The *Laplacian of Gaussian (LoG)*

\[
\nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}
\]

(very similar to a Difference of Gaussians (DoG) – i.e. a Gaussian minus a slightly smaller Gaussian)
Laplacian of Gaussian

- “Blob” detector

- Find maxima and minima of LoG operator in space and scale
Scale selection

- At what scale does the Laplacian achieve a maximum response for a binary circle of radius $r$?
Characteristic scale

- We define the characteristic scale as the scale that produces peak of Laplacian response.

Find local maxima in 3D position-scale space

\[ L_{xx}(\sigma) + L_{yy}(\sigma) \rightarrow \sigma^2 \]

\[ \Rightarrow \text{List of } (x, y, s) \]
Scale-space blob detector: Example
Scale-space blob detector: Example

\[ \text{sigma} = 11.9912 \]
Scale-space blob detector: Example
Scale Invariant Detection

- Functions for determining scale

Kernels:

$$\nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}$$

(Laplacian)

$$DoG = G(x, y, k\sigma) - G(x, y, \sigma)$$

(Difference of Gaussians)

where Gaussian

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

Note: The LoG and DoG operators are both rotation equivariant
Questions?
Feature descriptors

We know how to detect good points
Next question: How to match them?

Answer: Come up with a descriptor for each point, find similar descriptors between the two images