Computer Vision - Lecture 11

Local Features (cont’d)


Bastian Leibe
RWTH Aachen
http://www.vision.rwth-aachen.de
leibe@vision.rwth-aachen.de
Course Outline

- Image Processing Basics
- Segmentation & Grouping
- Object Recognition
- Object Categorization I
  - Sliding Window based Object Detection
- Local Features & Matching
  - Local Features - Detection and Description
  - Recognition with Local Features
- Object Categorization II
  - Part based Approaches
- 3D Reconstruction
- Motion and Tracking
Recap: Local Feature Matching Outline

1. Find a set of distinctive keypoints
2. Define a region around each keypoint
3. Extract and normalize the region content
4. Compute a local descriptor from the normalized region
5. Match local descriptors

\[ d(f_A, f_B) < T \]
Recap: Requirements for Local Features

- Problem 1:
  - Detect the same point \textit{independently} in both images

- Problem 2:
  - For each point correctly recognize the corresponding one

We need a reliable and distinctive descriptor!

We need a repeatable detector!
Recap: Harris Detector [Harris88]

- Compute second moment matrix (autocorrelation matrix)

\[
M(\sigma_I, \sigma_D) = g(\sigma_I) \begin{bmatrix}
I_x^2(\sigma_D) & I_x I_y(\sigma_D) \\
I_x I_y(\sigma_D) & I_y^2(\sigma_D)
\end{bmatrix}
\]

1. Image derivatives

2. Square of derivatives

3. Gaussian filter \( g(\sigma_I) \)

4. Cornerness function - two strong eigenvalues

\[
R = \det[M(\sigma_I, \sigma_D)] - \alpha[\text{trace}(M(\sigma_I, \sigma_D))]^2
= g(I_x^2)g(I_y^2) - [g(I_x I_y)]^2 - \alpha[g(I_x^2) + g(I_y^2)]^2
\]

5. Perform non-maximum suppression

Slide credit: Krystian Mikolajczyk
Recap: Harris Detector Responses [Harris88]

**Effect:** A very precise corner detector.

Slide credit: Krystian Mikolajczyk
Recap: Hessian Detector [Beaudet78]

- Hessian determinant

\[ \text{Hessian}(I) = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix} \]

\[ \text{det}(\text{Hessian}(I)) = I_{xx} I_{yy} - I_{xy}^2 \]

In Matlab:

\[ I_{xx} \ast I_{yy} - (I_{xy})^2 \]

Slide credit: Krystian Mikolajczyk
Effect: Responses mainly on corners and strongly textured areas.

Slide credit: Krystian Mikolajczyk
Topics of This Lecture

• Local Feature Extraction (cont’d)
  - Scale Invariant Region Selection
  - Orientation normalization
  - Affine Invariant Feature Extraction

• Local Descriptors
  - SIFT

• Applications
From Points to Regions...

- The Harris and Hessian operators define interest points.
  - Precise localization
  - High repeatability

- In order to compare those points, we need to compute a descriptor over a region.
  - How can we define such a region in a scale invariant manner?

- *I.e. how can we detect scale invariant interest regions?*
Naïve Approach: Exhaustive Search

- Multi-scale procedure
  - Compare descriptors while varying the patch size

\[ d(f_A, f_B) \neq 0 \]

Slide credit: Krystian Mikolajczyk
**Naïve Approach: Exhaustive Search**

- **Multi-scale procedure**
  - Compare descriptors while varying the patch size

\[ d(f_A, f_B) \]

Slide credit: Krystian Mikolajczyk
Naïve Approach: Exhaustive Search

• Multi-scale procedure
  - Compare descriptors while varying the patch size

\[ d(f_A, f_B) \neq 0 \]

Slide credit: Krystian Mikolajczyk
Naïve Approach: Exhaustive Search

- **Multi-scale procedure**
  - Compare descriptors while varying the patch size

\[
\begin{align*}
&f_A \quad \text{Similarity measure} \quad f_B \\
&d(f_A, f_B)
\end{align*}
\]
Naïve Approach: Exhaustive Search

- Comparing descriptors while varying the patch size
  - Computationally inefficient
  - Inefficient but possible for matching
  - Prohibitive for retrieval in large databases
  - Prohibitive for recognition

\[ d(f_A, f_B) \]

Slide credit: Krystian Mikolajczyk
Automatic Scale Selection

- **Solution:**
  - Design a function on the region, which is “scale invariant” *(the same for corresponding regions, even if they are at different scales)*

  ![Example](average_intensity.png)

  Example: average intensity. For corresponding regions (even of different sizes) it will be the same.

  - For a point in one image, we can consider it as a function of region size (patch width)
Automatic Scale Selection

- Common approach:
  - Take a local maximum of this function.
  - Observation: region size for which the maximum is achieved should be *invariant* to image scale.

**Important:** this scale invariant region size is found in each image *independently*!

![Graph showing scale selection](image)

- Image 1: $s_1$ is the region size for maximum $f$, and $s_2 = \frac{1}{2} s_1$.
- Image 2: $s_2$ is the region size for maximum $f$, and scale $= \frac{1}{2}$.

Slide credit: Kristen Grauman
Automatic Scale Selection

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

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

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

- Function responses for increasing scale (scale signature)

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

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

Slide credit: Krystian Mikolajczyk
Automatic Scale Selection

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

- Function responses for increasing scale (scale signature)

Slide credit: Krystian Mikolajczyk
Automatic Scale Selection

- Normalize: Rescale to fixed size

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

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

Slide credit: Tinne Tuytelaars
What Is A Useful Signature Function?

- Laplacian-of-Gaussian = “blob” detector
Characteristic Scale

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


Slide credit: Svetlana Lazebnik
Laplacian-of-Gaussian (LoG)

• Interest points:
  - Local maxima in scale space of Laplacian-of-Gaussian

\[
L_{xx}(\sigma) + L_{yy}(\sigma) \rightarrow \sigma^3
\]

Slide adapted from Krystian Mikolajczyk
Laplacian-of-Gaussian (LoG)

- **Interest points:**
  - Local maxima in scale space of Laplacian-of-Gaussian

\[
L_{xx}(\sigma) + L_{yy}(\sigma) \rightarrow \sigma^3
\]
Laplacian-of-Gaussian (LoG)

- **Interest points:**
  - Local maxima in scale space of Laplacian-of-Gaussian

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

Slide adapted from Krystian Mikolajczyk
Laplacian-of-Gaussian (LoG)

- **Interest points:**
  - Local maxima in scale space of Laplacian-of-Gaussian

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

\[ \Rightarrow \text{List of } (x, y, \sigma) \]
LoG Detector: Workflow
LoG Detector: Workflow

\[ \text{sigma} = 11.9912 \]
LoG Detector: Workflow

Slide credit: Svetlana Lazebnik
Difference-of-Gaussian (DoG)

- We can efficiently approximate the Laplacian with a difference of Gaussians:

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

(Laplacian)

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

(Difference of Gaussians)

- Advantages?
  - No need to compute 2\(^{nd}\) derivatives.
  - Gaussians are computed anyway, e.g. in a Gaussian pyramid.
Key point localization with DoG

- Detect maxima of difference-of-Gaussian (DoG) in scale space
- Then reject points with low contrast (threshold)
- Eliminate edge responses

Candidate keypoints:
list of \((x, y, \sigma)\)

Slide credit: David Lowe
DoG - Efficient Computation

- Computation in Gaussian scale pyramid

Original image

Sampling with step \( \sigma^4 = 2 \)

\[
\sigma = 2^4
\]

Scale (first octave)

Scale (next octave)

\[ \frac{1}{\sigma} \]

Gaussian

Difference of Gaussian (DOG)
Results: Lowe’s DoG
Harris-Laplace [Mikolajczyk ‘01]

1. Initialization: Multiscale Harris corner detection

Slide adapted from Krystian Mikolajczyk
Harris-Laplace [Mikolajczyk ‘01]

1. Initialization: Multiscale Harris corner detection
2. Scale selection based on Laplacian
   (same procedure with Hessian ⇒ Hessian-Laplace)
Summary: Scale Invariant Detection

• **Given:** Two images of the same scene with a large *scale difference* between them.

• **Goal:** Find *the same* interest points *independently* in each image.

• **Solution:** Search for *maxima* of suitable functions in *scale* and in *space* (over the image).

• **Two strategies**
  - Laplacian-of-Gaussian (LoG)
  - Difference-of-Gaussian (DoG) as a fast approximation

  *These can be used either on their own, or in combinations with single-scale keypoint detectors (Harris, Hessian).*
Topics of This Lecture

- **Local Feature Extraction (cont’d)**
  - Scale Invariant Region Selection
  - Orientation normalization
  - Affine Invariant Feature Extraction

- **Local Descriptors**
  - SIFT

- **Applications**
Rotation Invariant Descriptors

- Find local orientation
  - Dominant direction of gradient for the image patch

- Rotate patch according to this angle
  - This puts the patches into a canonical orientation.

Slide credit: Svetlana Lazebnik, Matthew Brown
Orientation Normalization: Computation

- Compute orientation histogram
- Select dominant orientation
- Normalize: rotate to fixed orientation

[Lowe, SIFT, 1999]
Topics of This Lecture

• Local Feature Extraction (cont’d)
  - Scale Invariant Region Selection
  - Orientation normalization
  - Affine Invariant Feature Extraction

• Local Descriptors
  - SIFT

• Applications
The Need for Invariance

- Up to now, we had invariance to
  - Translation
  - Scale
  - Rotation
- Not sufficient to match regions under viewpoint changes
  - For this, we need also affine adaptation
Affine Adaptation

- **Problem:**
  - Determine the characteristic shape of the region.
  - Assumption: shape can be described by “local affine frame”.

- **Solution: iterative approach**
  - Use a circular window to compute second moment matrix.
  - Compute eigenvectors to adapt the circle to an ellipse.
  - Recompute second moment matrix using new window and iterate...
Iterative Affine Adaptation

1. Detect keypoints, e.g. multi-scale Harris
2. Automatically select the scales
3. Adapt affine shape based on second order moment matrix
4. Refine point location

Affine Normalization/Deskewing

• Steps
  - Rotate the ellipse’s main axis to horizontal
  - Scale the x axis, such that it forms a circle
Affine Adaptation Example

Scale-invariant regions (blobs)

Slide credit: Svetlana Lazebnik
Affine Adaptation Example

Affine-adapted blobs

Slide credit: Svetlana Lazebnik
Summary: Affine-Inv. Feature Extraction

- Extract affine regions
- Normalize regions
- Eliminate rotational ambiguity
- Compare descriptors

Slide credit: Svetlana Lazebnik
Invariance vs. Covariance

- **Invariance:**
  - \( \text{features(} \text{transform(image)} \text{)} = \text{features(image)} \)

- **Covariance:**
  - \( \text{features(} \text{transform(image)} \text{)} = \text{transform(features(image))} \)

Covariant detection \(\Rightarrow\) invariant description

Slide credit: Svetlana Lazebnik, David Lowe
Topics of This Lecture

- **Local Feature Extraction (cont’d)**
  - Orientation normalization
  - Affine Invariant Feature Extraction

- **Local Descriptors**
  - SIFT
  - Applications

- **Recognition with Local Features**
  - Matching local features
  - Finding consistent configurations
  - Alignment: linear transformations
  - Affine estimation
  - Homography estimation
Local Descriptors

• We know how to detect points
• Next question:

How to *describe* them for matching?

Point descriptor should be:
1. Invariant
2. Distinctive

Slide credit: Kristen Grauman
Local Descriptors

- Simplest descriptor: list of intensities within a patch.
- What is this going to be invariant to?

Write regions as vectors

\[ A \rightarrow a, \quad B \rightarrow b \]
Feature Descriptors

- Disadvantage of patches as descriptors:
  - Small shifts can affect matching score a lot

- Solution: histograms

Slide credit: Svetlana Lazebnik
Feature Descriptors: SIFT

- **Scale Invariant Feature Transform**
- **Descriptor computation:**
  - Divide patch into 4x4 sub-patches: 16 cells
  - Compute histogram of gradient orientations (8 reference angles) for all pixels inside each sub-patch
  - Resulting descriptor: 4x4x8 = 128 dimensions


Slide credit: Svetlana Lazebnik
Overview: SIFT

- Extraordinarily robust matching technique
  - Can handle changes in viewpoint up to ~60 deg. out-of-plane rotation
  - Can handle significant changes in illumination
    - Sometimes even day vs. night (below)
  - Fast and efficient—can run in real time
  - Lots of code available
Working with SIFT Descriptors

- One image yields:
  - $n$ 2D points giving positions of the patches
    - [n x 2 matrix]
  - $n$ scale parameters specifying the size of each patch
    - [n x 1 vector]
  - $n$ orientation parameters specifying the angle of the patch
    - [n x 1 vector]
  - $n$ 128-dimensional descriptors: each one is a histogram of the gradient orientations within a patch
    - [n x 128 matrix]
Local Descriptors: SURF

- Fast approximation of SIFT idea
  - Efficient computation by 2D box filters & integral images
    ⇒ 6 times faster than SIFT
  - Equivalent quality for object identification
    - http://www.vision.ee.ethz.ch/~surf

- GPU implementation available
  - Feature extraction @ 100Hz
    (detector + descriptor, 640×480 img)
You Can Try It At Home...

- For most local feature detectors, executables are available online:
  - [http://robots.ox.ac.uk/~vgg/research/affine](http://robots.ox.ac.uk/~vgg/research/affine)
  - [http://www.vision.ee.ethz.ch/~surf](http://www.vision.ee.ethz.ch/~surf)
Affine Covariant Features

Affine Covariant Region Detectors

**Input Image**

**Detector output**

- Format:
  - `m`
  - `u_1 v_1 a_1 b_1 c_1`
  - ...
  - `u_m v_m a_m b_m c_m`

**Image with displayed regions**

**Parameters defining an affine region**

\[ a(x-u) + 2b(y-v) + c = (y-v)(y-v) = 1 \]

with \((0,0)\) at image top left corner

**Code**

- Provided by the authors, see publications for details and links to authors' web sites.

**Linux binaries**

- Harris-Affine & Heronian Affine
- M5ER - Maximal stable extremal regions (also Windows)
- EBK - Intensity extrema based detector
- EBR - Edge based detector
- Salient region detector

**Example of use**

- Prompt:
  ```
  prompt>./h_affine.in -heraff -i im1.ppm -o im1.heraff -thres 1000
  prompt>./h_affine.in -heraff -i im1.ppm -o im1.heraff -thres 500
  ```

- Displaying 1

**http://www.robots.ox.ac.uk/~vgg/research/affine/detectors.html#binaries**
Topics of This Lecture

- Local Feature Extraction (cont’d)
  - Orientation normalization
  - Affine Invariant Feature Extraction

- Local Descriptors
  - SIFT
  - Applications

- Recognition with Local Features
  - Matching local features
  - Finding consistent configurations
  - Alignment: linear transformations
  - Affine estimation
  - Homography estimation
Applications of Local Invariant Features

- Wide baseline stereo
- Motion tracking
- Panoramas
- Mobile robot navigation
- 3D reconstruction
- Recognition
  - Specific objects
  - Textures
  - Categories
- ...

Slide credit: Kristen Grauman
Wide-Baseline Stereo

Image from T. Tuytelaars ECCV 2006 tutorial
Automatic Mosaicing

[Brown & Lowe, ICCV'03]
Panorama Stitching

(a) Matier data set (7 images)

(b) Matier final stitch

http://www.cs.ubc.ca/~mbrown/autostitch/autostitch.html

[Brown, Szeliski, and Winder, 2005]
Recognition of Specific Objects, Scenes

Schmid and Mohr 1997

Rothganger et al. 2003

Sivic and Zisserman, 2003

Lowe 2002

Slide credit: Kristen Grauman
Recognition of Categories

### Constellation model

- Weber et al. (2000)
- Fergus et al. (2003)
- Csurka et al. (2004)
- Dorko & Schmid (2005)
- Sivic et al. (2005)
- Lazebnik et al. (2006), ...

### Bags of words

<table>
<thead>
<tr>
<th>Database</th>
<th>Sample cluster #1</th>
<th>Sample cluster #2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Airplanes</td>
<td><img src="image1" alt="Sample images" /></td>
<td><img src="image2" alt="Sample images" /></td>
</tr>
<tr>
<td>Motorbikes</td>
<td><img src="image3" alt="Sample images" /></td>
<td><img src="image4" alt="Sample images" /></td>
</tr>
<tr>
<td>Leaves</td>
<td><img src="image5" alt="Sample images" /></td>
<td><img src="image6" alt="Sample images" /></td>
</tr>
<tr>
<td>Wild Cats</td>
<td><img src="image7" alt="Sample images" /></td>
<td><img src="image8" alt="Sample images" /></td>
</tr>
<tr>
<td>Faces</td>
<td><img src="image9" alt="Sample images" /></td>
<td><img src="image10" alt="Sample images" /></td>
</tr>
<tr>
<td>Bicycles</td>
<td><img src="image11" alt="Sample images" /></td>
<td><img src="image12" alt="Sample images" /></td>
</tr>
<tr>
<td>People</td>
<td><img src="image13" alt="Sample images" /></td>
<td><img src="image14" alt="Sample images" /></td>
</tr>
</tbody>
</table>

---

Slide credit: Svetlana Lazebnik
Value of Local Features

- **Advantages**
  - Critical to find distinctive and repeatable local regions for multi-view matching.
  - Complexity reduction via selection of distinctive points.
  - Describe images, objects, parts without requiring segmentation; robustness to clutter & occlusion.
  - Robustness: similar descriptors in spite of moderate view changes, noise, blur, etc.

- **How can we use local features for such applications?**
  - Next week: matching and recognition
References and Further Reading

• More details on homography estimation can be found in Chapter 4.7 of
  - R. Hartley, A. Zisserman
    Multiple View Geometry in Computer Vision
    2nd Ed., Cambridge Univ. Press, 2004

• Details about the DoG detector and the SIFT descriptor can be found in
  - D. Lowe, Distinctive image features from scale-invariant keypoints,
    IJCV 60(2), pp. 91-110, 2004

• Try the available local feature detectors and descriptors
  - http://www.robots.ox.ac.uk/~vgg/research/affine/detectors.html#binaries