Feature detectors and descriptors

Fei-Fei Li
Feature Detection

detected points – (~300) coordinates, neighbourhoods

Feature Description

local descriptors – (invariant) vectors

database of local descriptors

Matching / Indexing / Recognition

e.g. Mahalanobis distance + Voting algorithm [Mikolajczyk & Schmid ’01]

e.g. DoG

e.g. SIFT
Some of the challenges...

- **Geometry**
  - Rotation
  - Similarity (rotation + uniform scale)
  - Affine (scale dependent on direction)
    valid for: orthographic camera, locally planar object

- **Photometry**
  - Affine intensity change ($I \rightarrow aI + b$)
<table>
<thead>
<tr>
<th>Detector</th>
<th>Descriptor</th>
<th>Intensity</th>
<th>Rotation</th>
<th>Scale</th>
<th>Affine</th>
</tr>
</thead>
<tbody>
<tr>
<td>Harris corner</td>
<td>2nd moment(s)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mikolajczyk &amp; Schmid ’01, ‘02</td>
<td>2nd moment(s)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tuytelaars, ‘00</td>
<td>2nd moment(s)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lowe ’99 (DoG)</td>
<td>SIFT, PCA-SIFT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kadir &amp; Brady, 01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Matas, ‘02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>others</td>
<td>others</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detector</td>
<td>Descriptor</td>
<td>Intensity</td>
<td>Rotation</td>
<td>Scale</td>
<td>Affine</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>----------------</td>
<td>-----------</td>
<td>----------</td>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>Harris corner</td>
<td>2\textsuperscript{nd} moment(s)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mikolajczyk &amp; Schmid '01, '02</td>
<td>2\textsuperscript{nd} moment(s)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tuytelaars, '00</td>
<td>2\textsuperscript{nd} moment(s)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lowe '99 (DoG)</td>
<td>SIFT, PCA-SIFT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kadir &amp; Brady, 01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Matas, '02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>others</td>
<td>others</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
An introductory example:

**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 in window, 0 outside

or

Gaussian
Harris Detector: Mathematics

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

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

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

\[
M = \sum_{x, y} w(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}
\]
Intensity change in shifting window: eigenvalue analysis

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

\[ \lambda_1, \lambda_2 - \text{eigenvalues of } M \]

Ellipse \( E(u,v) = \text{const} \)
Classification of image points using eigenvalues of $M$:

- **"Corner"**
  - $\lambda_1$ and $\lambda_2$ are large;
  - $\lambda_1 \sim \lambda_2$;
  - $E$ increases in all directions

- **"Edge"**
  - $\lambda_1 >> \lambda_2$

- **"Flat" region**
  - $\lambda_1$ and $\lambda_2$ are small;
  - $E$ is almost constant in all directions
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

• The Algorithm:
  – Find points with large corner response function $R$ ($R > \text{threshold}$)
  – Take the points of local maxima of $R$
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: Summary

• Average intensity change in direction \([u,v]\) can be expressed as a bilinear form:

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

• Describe a point in terms of eigenvalues of \(M\): measure of corner response

\[
R = \lambda_1 \lambda_2 - k \left( \lambda_1 + \lambda_2 \right)^2
\]

• A good (corner) point should have a large intensity change in all directions, i.e. \(R\) should be large positive
Harris Detector: Some Properties

• Rotation invariance

Ellipse rotates but its shape (i.e. eigenvalues) remains the same

*Corner response $R$ is invariant to image rotation*
Harris Detector: Some Properties

- Partial invariance to affine intensity change
  - Only derivatives are used => invariance to intensity shift $I \rightarrow I + b$
  - Intensity scale: $I \rightarrow aI$
Harris Detector: Some Properties

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

All points will be classified as *edges*.

Corner!
Harris Detector: Some Properties

- Quality of Harris detector for different scale changes

Repeatability rate:
\[
\frac{\text{# correspondences}}{\text{# possible correspondences}}
\]

<table>
<thead>
<tr>
<th>Detector</th>
<th>Descriptor</th>
<th>Intensity</th>
<th>Rotation</th>
<th>Scale</th>
<th>Affine</th>
</tr>
</thead>
<tbody>
<tr>
<td>Harris corner</td>
<td>2\textsuperscript{nd} moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Mikolajczyk &amp; Schmid '01, '02</td>
<td>2\textsuperscript{nd} moment(s)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tuytelaars, ‘00</td>
<td>2\textsuperscript{nd} moment(s)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lowe '99 (DoG)</td>
<td>SIFT, PCA-SIFT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kadir &amp; Brady, 01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Matas, ‘02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>others</td>
<td>others</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detector</td>
<td>Descriptor</td>
<td>Intensity</td>
<td>Rotation</td>
<td>Scale</td>
<td>Affine</td>
</tr>
<tr>
<td>--------------------------</td>
<td>---------------------</td>
<td>-----------</td>
<td>----------</td>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>Harris corner</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Mikolajczyk &amp; Schmid ’01, ’02</td>
<td>2nd moment(s)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tuytelaars, ’00</td>
<td>2nd moment(s)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lowe ’99 (DoG)</td>
<td>SIFT, PCA-SIFT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kadir &amp; Brady, 01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Matas, ’02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>others</td>
<td>others</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Interest point detectors

**Harris-Laplace** [Mikolajczyk & Schmid ’01]

- Adds scale invariance to Harris points
  - Set $s_i = \lambda s_d$
  - Detect at several scales by varying $s_d$
  - Only take local maxima (8-neighbourhood) of scale adapted Harris points
  - Further restrict to scales at which Laplacian is local maximum
• Selected scale determines size of support region
• Laplacian justified experimentally
  – compared to gradient squared & DoG
  – [Lindeberg '98] gives thorough analysis of scale-space

Interest point detectors
Harris-Laplace [Mikolajczyk & Schmid '01]
Interest point detectors

**Harris-Affine** [Mikolajczyk & Schmid ’02]

- Adds invariance to affine image transformations
- Initial locations and isotropic scale found by Harris-Laplace
- Affine invariant neighbourhood evolved iteratively using the 2\textsuperscript{nd} moment matrix $\mu$:

$$g(\Sigma) = \frac{1}{2\pi \sqrt{|\Sigma|}} \exp\left(-\frac{x^T \Sigma^{-1} x}{2}\right)$$

$$L(x, \Sigma) = g(\Sigma) \otimes I(x)$$

$$\mu(x, \Sigma_I, \Sigma_D) = g(\Sigma_I) \otimes ((\nabla L(x, \Sigma_D))(\nabla L(x, \Sigma_D))^T)$$
Interest point detectors

**Harris-Affine** [Mikolajczyk & Schmid ’02]

For affinely related points:
\[ \mathbf{x}_L = A \mathbf{x}_R \]

If \( \mu(\mathbf{x}_L, \Sigma_{I,L}, \Sigma_{D,L}) = M_L \quad \Sigma_{I,L} = tM_L^{-1} \quad \Sigma_{D,L} = dM_L^{-1} \)

and \( \mu(\mathbf{x}_R, \Sigma_{I,R}, \Sigma_{D,R}) = M_R \quad \Sigma_{I,R} = tM_R^{-1} \quad \Sigma_{D,R} = dM_R^{-1} \)

Then by normalising:
\[ \mathbf{x}'_L \rightarrow M_L^{-1/2} \mathbf{x}_L \quad \text{and} \quad \mathbf{x}'_R \rightarrow M_R^{-1/2} \mathbf{x}_R \]

We get:
\[ \mathbf{x}'_L \rightarrow R \mathbf{x}'_R \]

so the normalised regions are related by a pure rotation

See also [Lindeberg & Garding ’97] and [Baumberg ’00]
Interest point detectors

Harris-Affine [Mikolajczyk & Schmid ’02]

- Algorithm iteratively adapts
  - shape of support region
  - spatial location $x^{(k)}$
  - integration scale $\sigma_i$ (based on Laplacian)
  - derivation scale $\sigma_D = s\sigma_i$

![Diagram of Harris-Affine interest point detectors](image)
<table>
<thead>
<tr>
<th>Detector</th>
<th>Descriptor</th>
<th>Intensity</th>
<th>Rotation</th>
<th>Scale</th>
<th>Affine</th>
</tr>
</thead>
<tbody>
<tr>
<td>Harris corner</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Mikolajczyk &amp; Schmid ’01, ‘02</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Tuytelaars, ‘00</td>
<td>2nd moment(s)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lowe ’99 (DoG)</td>
<td>SIFT, PCA-SIFT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kadir &amp; Brady, 01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Matas, ‘02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>others</td>
<td>others</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detector</td>
<td>Descriptor</td>
<td>Intensity</td>
<td>Rotation</td>
<td>Scale</td>
<td>Affine</td>
</tr>
<tr>
<td>---------------------------</td>
<td>--------------------------</td>
<td>-----------</td>
<td>----------</td>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>Harris corner</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Mikolajczyk &amp; Schmid '01, '02</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Tuytelaars, ‘00</td>
<td>2nd moment(s)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lowe ’99 (DoG)</td>
<td>SIFT, PCA-SIFT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kadir &amp; Brady, 01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Matas, ‘02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>others</td>
<td>others</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Affine Invariant Detection

• Take a local intensity extremum as initial point
• Go along every ray starting from this point and stop when extremum of function \( f \) is reached

\[
f(t) = \frac{|I(t) - I_0|}{\frac{1}{t} \int_o^t |I(t) - I_0|\,dt}
\]

• We will obtain approximately corresponding regions

Affine Invariant Detection

• The regions found may not exactly correspond, so we approximate them with ellipses

• Geometric Moments:

$$m_{pq} = \int_{-\infty}^{\infty} x^p y^q f(x, y) dxdy$$

Fact: moments $m_{pq}$ uniquely determine the function $f$

Taking $f$ to be the characteristic function of a region (1 inside, 0 outside), moments of orders up to 2 allow to approximate the region by an ellipse

This ellipse will have the same moments of orders up to 2 as the original region
Affine Invariant Detection

- **Covariance matrix** of region points defines an ellipse:

\[ p^T \Sigma_1^{-1} p = 1 \]
\[ \Sigma_1 = \langle pp^T \rangle_{\text{region 1}} \]

\[ q = Ap \]
\[ q^T \Sigma_2^{-1} q = 1 \]
\[ \Sigma_2 = \langle qq^T \rangle_{\text{region 2}} \]

\( p = [x, y]^T \) is relative to the center of mass

\[ \Sigma_2 = A \Sigma_1 A^T \]

Ellipses, computed for corresponding regions, also correspond!
Affine Invariant Detection

- Algorithm summary (detection of affine invariant region):
  - Start from a *local intensity extremum* point
  - Go in *every direction* until the point of extremum of some function $f$
  - Curve connecting the points is the region boundary
  - Compute *geometric moments* of orders up to 2 for this region
  - Replace the region with *ellipse*

---

<table>
<thead>
<tr>
<th>Detector</th>
<th>Descriptor</th>
<th>Intensity</th>
<th>Rotation</th>
<th>Scale</th>
<th>Affine</th>
</tr>
</thead>
<tbody>
<tr>
<td>Harris corner</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Mikolajczyk &amp; Schmid ‘01, ‘02</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Tuytelaars, ‘00</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Lowe ’99 (DoG)</td>
<td>SIFT, PCA-SIFT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kadir &amp; Brady, 01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Matas, ‘02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>others</td>
<td>others</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detector</td>
<td>Descriptor</td>
<td>Intensity</td>
<td>Rotation</td>
<td>Scale</td>
<td>Affine</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>--------------------------</td>
<td>-----------</td>
<td>----------</td>
<td>---------</td>
<td>--------</td>
</tr>
<tr>
<td>Harris corner</td>
<td>2\textsuperscript{nd} moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Mikolajczyk &amp; Schmid ’01, ’02</td>
<td>2\textsuperscript{nd} moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Tuytelaars, ’00</td>
<td>2\textsuperscript{nd} moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No (Yes ’04 ?)</td>
<td>Yes</td>
</tr>
<tr>
<td>Lowe ’99 (DoG)</td>
<td>SIFT, PCA-SIFT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kadir &amp; Brady, 01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Matas, ’02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>others</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Interest point detectors

*Difference of Gaussians* [Lowe '99]

- Difference of Gaussians in scale-space
  - detects ‘blob’-like features
- Can be computed efficiently with image pyramid
- Approximates Laplacian for correct scale factor
- Invariant to rotation and scale changes
Key point localization

- Detect maxima and minima of difference-of-Gaussian in scale space (Lowe, 1999)
- Fit a quadratic to surrounding values for sub-pixel and sub-scale interpolation (Brown & Lowe, 2002)
- Taylor expansion around point:
  \[ D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x \]
- Offset of extremum (use finite differences for derivatives):
  \[ \hat{x} = -\frac{\partial^2 D^{-1}}{\partial x^2} \frac{\partial D}{\partial x} \]
Select canonical orientation

• Create histogram of local gradient directions computed at selected scale
• Assign canonical orientation at peak of smoothed histogram
• Each key specifies stable 2D coordinates (x, y, scale, orientation)
Example of keypoint detection

Threshold on value at DOG peak and on ratio of principle curvatures (Harris approach)

(a) 233x189 image
(b) 832 DOG extrema
(c) 729 left after peak value threshold
(d) 536 left after testing ratio of principle curvatures
Creating features stable to viewpoint change

- Edelman, Intrator & Poggio (97) showed that complex cell outputs are better for 3D recognition than simple correlation
SIFT vector formation

- Thresholded image gradients are sampled over 16x16 array of locations in scale space
- Create array of orientation histograms
- 8 orientations x 4x4 histogram array = 128 dimensions
Feature stability to noise

- Match features after random change in image scale & orientation, with differing levels of image noise
- Find nearest neighbor in database of 30,000 features
Feature stability to affine change

- Match features after random change in image scale & orientation, with 2% image noise, and affine distortion
- Find nearest neighbor in database of 30,000 features
<table>
<thead>
<tr>
<th>Detector</th>
<th>Descriptor</th>
<th>Intensity</th>
<th>Rotation</th>
<th>Scale</th>
<th>Affine</th>
</tr>
</thead>
<tbody>
<tr>
<td>Harris corner</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Mikolajczyk &amp; Schmid '01, '02</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Tuytelaars, ‘00</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No (Yes ’04 ?)</td>
<td>Yes</td>
</tr>
<tr>
<td>Lowe ’99 (DoG)</td>
<td>SIFT, PCA-SIFT</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Kadir &amp; Brady, 01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Matas, ‘02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>others</td>
<td>others</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detector</td>
<td>Descriptor</td>
<td>Intensity</td>
<td>Rotation</td>
<td>Scale</td>
<td>Affine</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------</td>
<td>-----------</td>
<td>----------</td>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>Harris corner</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Mikolajczyk &amp; Schmid ’01, ’02</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Tuytelaars, ‘00</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Lowe ’99 (DoG)</td>
<td>SIFT, PCA-SIFT</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Kadir &amp; Brady, 01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Matas, ’02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>others</td>
<td>others</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Other interest point detectors

Scale Saliency [Kadir & Brady ’01, ’03]
Other interest point detectors

Scale Saliency [Kadir & Brady ’01, ’03]

- Uses entropy measure of local pdf of intensities:

\[
H_D(s, x) = - \int_{d \in D} p(d, s, x) \log_2 p(d, s, x).dd
\]

- Takes local maxima in scale

- Weights with ‘change’ of distribution with scale:

\[
W_D(s, x) = s \int_{d \in D} \left| \frac{\partial}{\partial s} p(d, s, x) \right|.dd
\]

- To get saliency measure:

\[
Y_D(s, x) = H_D(s, x) \times W_D(s, x)
\]
Other interest point detectors

Scale Saliency [Kadir & Brady ’01, ’03]

Most salient parts detected

Just using 
$H_D(s, x)$

Using 
$Y_D(s, x) = H_D W_D$
Other interest point detectors
maximum stable extremal regions [matas et al. 02]

- Sweep threshold of intensity from black to white
- Locate regions based on stability of region with respect to change of threshold
<table>
<thead>
<tr>
<th>Detector</th>
<th>Descriptor</th>
<th>Intensity</th>
<th>Rotation</th>
<th>Scale</th>
<th>Affine</th>
</tr>
</thead>
<tbody>
<tr>
<td>Harris corner</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Mikolajczyk &amp; Schmid ’01, ’02</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Tuytelaars, ’00</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No (Yes ’04 ?)</td>
<td>Yes</td>
</tr>
<tr>
<td>Lowe ’99 (DoG)</td>
<td>SIFT, PCA-SIFT</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Kadir &amp; Brady, 01</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>?</td>
</tr>
<tr>
<td>Matas, ’02</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>?</td>
</tr>
<tr>
<td>others</td>
<td>others</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detector</td>
<td>Descriptor</td>
<td>Intensity</td>
<td>Rotation</td>
<td>Scale</td>
<td>Affine</td>
</tr>
<tr>
<td>---------------------------</td>
<td>------------------</td>
<td>-----------</td>
<td>----------</td>
<td>-------</td>
<td>--------</td>
</tr>
<tr>
<td>Harris corner</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Mikolajczyk &amp; Schmid ’01, ‘02</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Tuytelaars, ’00</td>
<td>2nd moment(s)</td>
<td>Yes</td>
<td>Yes</td>
<td>No (Yes ’04 ?)</td>
<td>Yes</td>
</tr>
<tr>
<td>Lowe ’99 (DoG)</td>
<td>SIFT, PCA-SIFT</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Kadir &amp; Brady, 01</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>?</td>
</tr>
<tr>
<td>Matas, ’02</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>?</td>
</tr>
<tr>
<td>others</td>
<td>others</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Affine-invariant texture recognition

- Texture recognition under viewpoint changes and non-rigid transformations

- Use of affine-invariant regions
  - invariance to viewpoint changes
  - spatial selection => more compact representation, reduction of redundancy in texton dictionary

Overview of the approach

1. Extract affine regions
2. Compute affine-invariant descriptors
3. Find clusters and signatures
4. Compute EMD matrix
Region extraction

Harris detector

Laplace detector
Descriptors – Spin images
Spatial selection

clustering each pixel

clustering selected pixels
Signature and EMD

• Hierarchical clustering
  => Signature : \( S = \{ (m_1, w_1), \ldots, (m_k, w_k) \} \)

• Earth movers distance

\[
D( S, S' ) = \frac{\sum_{i,j} f_{ij} d(m_i, m'_j)}{\sum_{i,j} f_{ij}}
\]

– robust distance, optimizes the flow between distributions
– can match signatures of different size
– not sensitive to the number of clusters
Database with viewpoint changes

20 samples of 10 different textures
Results

Spin images Gabor-like filters

[Graphs showing average recognition rate vs. number of retrievals for spin images and Gabor-like filters]
Feature detectors and descriptors
Widely used descriptors

- SIFT
- Gray-scale intensity values
- Steerable filters
- GLOH
- Shape context & geometric blur
SIFT

Image gradients

Keypoint descriptor
Gray-scale intensity

Normalize → 11x11 patch → Projection onto PCA basis

\[
\begin{pmatrix}
c_1 \\
c_2 \\
\vdots \\
c_{15}
\end{pmatrix}
\]
Steerable filters

\[
R_1^{0^\circ} = G_1^0 \ast I \\
R_1^{90^\circ} = G_1^{90} \ast I \\
\text{then} \\
R_1^\theta = \cos(\theta)R_1^{0^\circ} + \sin(\theta)R_1^{90^\circ}
\]
Steerable filters

Gaussian derivatives up to 4\textsuperscript{th} order. The remaining derivatives can be computed by rotation of 90 degrees.
GLOH

- GLOH: Gradient Location and Orientation Histogram (Miko04)
  - Very similar to SIFT.
  - Log-polar location grid:
    - 3 bins in radial direction;
    - 8 bins in angular direction
    - Gradient orientation quantized in 16 bins.
  - Total: $(2 \times 8 + 1) \times 16 = 272$ bins $\rightarrow$ PCA dimension reduction.
Shape context

Belongie et al. 2002
Geometric blur

In practice compute discrete blur levels for whole image and sample as needed for each feature location.

Berg et al. 2001
Geometric blur