Today

Feature and keypoint extraction
Cylindrical panorama

1. Take pictures on a tripod (or handheld)
2. Warp to cylindrical coordinate
3. Compute pairwise alignments
4. Fix up the end-to-end alignment
5. Blending
6. Crop the result and import into a viewer
Invariant Local Features

Image content is transformed into local feature coordinates that are (ideally!!) invariant to translation, rotation, scale, and other imaging parameters.
More motivation…

• Feature points are used for:
  – Homography and fundamental matrix estimation
  – Alignment for Mosaics
  – 3D reconstruction
  – Motion tracking
  – Object recognition
  – Robot navigation
  – … others
Aperture problem

(Seitz)
Corner detector
The Basic Idea

• We should easily recognize the point by looking through a small window
• Shifting a window in *any direction* should give a *large change* in intensity
Moravec corner detector

flat
Moravec corner detector

flat
Moravec corner detector

flat

edge
Moravec corner detector

flat

edge

corner isolated point
Moravec corner detector

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

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

Change of intensity for the shift $[u,v]$:

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

- **Window function**
- **Shifted intensity**
- **Intensity**
Moravec corner detector

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

Four shifts: \((u,v) = (1,0), (1,1), (0,1), (-1, 1)\)

Look for local maxima in \(\min\{E\}\) over the principle directions
Problems of Moravec detector

- At most only a set of shifts at every $45^\circ$ is considered
- Only minimum of $E$ is taken into account

Harris corner detector (1988) solves these problems.
Rewriting the operations in Moravec

Horizontal gradient at $A_5$:

$$\frac{\partial I_{A_5}}{\partial x} \approx (I_{A_6} - I_{A_4}) = I_{A_5} \otimes (-1, 0, 1)$$
Rewriting the operations in Moravec

Horizontal gradient at $A_5$:

$$\frac{\partial I_{A5}}{\partial x} \approx (I_{A6} - I_{A4}) = I_{A5} \otimes (-1, 0, 1)$$

Vertical gradient at $A_5$:

$$\frac{\partial I_{A5}}{\partial y} \approx (I_{A2} - I_{A8}) = I_{A5} \otimes (-1, 0, 1)^r$$
Rewriting the operations in Moravec

Simple discrete gradient approximations
Rewriting the operations in Moravec

Horizontal gradient at \( A_5 \):

\[
\frac{\partial I_{A5}}{\partial x} \approx (I_{A6} - I_{A4}) = I_{A5} \otimes (-1, 0, 1)
\]

Vertical gradient at \( A_5 \):

\[
\frac{\partial I_{A5}}{\partial y} \approx (I_{A2} - I_{A8}) = I_{A5} \otimes (-1, 0, 1)^T
\]

Upper-right diagonal gradient at \( A_5 \):

\[
\frac{\partial I_{A5}}{\partial h} \approx (I_{A3} - I_{A7}) = I_{A5} \otimes \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}
\]
Rewriting the operations in Moravec

Intensity variation can be written as a function of the gradient of the image. For an arbitrary shift \((u,v)\) we can state the intensity variation as

\[
V_{u,v}(x,y) = \sum_{\forall i \text{ in the window centered at } (x,y)} \left( u \frac{\partial I_i}{\partial x} + v \frac{\partial I_i}{\partial y} \right)^2
\]
Harris corner detector

Noisy response due to a binary window function
Use a Gaussian function

\[ w(x, y) = \exp\left( -\frac{(x^2 + y^2)}{2\sigma^2} \right) \]

Window function \( w(x, y) = \)

Gaussian
Harris corner detector

**Gaussian Window**

<table>
<thead>
<tr>
<th>w1</th>
<th>w2</th>
<th>w3</th>
</tr>
</thead>
<tbody>
<tr>
<td>.04</td>
<td>.12</td>
<td>.04</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>w4</th>
<th>w5</th>
<th>w6</th>
</tr>
</thead>
<tbody>
<tr>
<td>.12</td>
<td>.36</td>
<td>.12</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>w7</th>
<th>w8</th>
<th>w9</th>
</tr>
</thead>
<tbody>
<tr>
<td>.04</td>
<td>.12</td>
<td>.04</td>
</tr>
</tbody>
</table>

**Weighted Horizontal Intensity Variation:**

\[
V_x = \sum_{i=1}^{9} w_i (A_i - B_i)^2 = \sum_{i=1}^{9} w_i (B_i - A_i)^2 \approx \sum_{i=1}^{9} w_i \left( \frac{\partial I_i}{\partial x} \right)^2
\]

where \( \frac{\partial I_i}{\partial x} \equiv I_i \otimes (-1, 0, 1) \approx B_i - A_i \)

**X-direction Shift**

<table>
<thead>
<tr>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>B1</th>
<th>A4</th>
<th>A5</th>
<th>A6</th>
<th>B4</th>
<th>A7</th>
<th>A8</th>
<th>A9</th>
<th>B7</th>
<th>B3</th>
<th>B6</th>
<th>B9</th>
</tr>
</thead>
<tbody>
<tr>
<td>B1</td>
<td>B2</td>
<td>B3</td>
<td></td>
<td>B4</td>
<td>B5</td>
<td>B6</td>
<td></td>
<td>B7</td>
<td>B8</td>
<td>B9</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Harris corner detector

\[ V_{u,v}(x,y) = \sum_{\forall i \text{ in the window centered at } (x,y)} w_i \left( u \frac{\partial I_i}{\partial x} + v \frac{\partial I_i}{\partial y} \right)^2 \]

\[ V_{u,v}(x,y) = \sum_{\forall i \text{ in the window centered at } (x,y)} w_i \left( u \frac{\partial I_i}{\partial x} + v \frac{\partial I_i}{\partial y} \right)^2 \]

\[ = \sum_{\forall i \text{ in the window centered at } (x,y)} w_i \left( u^2 \frac{\partial I_i}{\partial x}^2 + 2uv \frac{\partial I_i}{\partial x} \frac{\partial I_i}{\partial y} + v^2 \frac{\partial I_i}{\partial y}^2 \right) \]

\[ = Au^2 + 2Cuv + Bv^2 \]

where: \[ A = \left( \frac{\partial I}{\partial x} \right)^2 \otimes w, \ B = \left( \frac{\partial I}{\partial y} \right)^2 \otimes w, \ C = \left( \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} \right) \otimes w \]

\( \otimes \) is the convolution operator
Harris corner detector

\[ V_{u,v}(x,y) = \sum_{i \text{ in the window centered at } (x,y)} w_i \left( u \frac{\partial I_i}{\partial x} + v \frac{\partial I_i}{\partial y} \right)^2 \]

\[ V_{u,v}(x,y) = Au^2 + 2Cu v + Bv^2 \]

\[ = \begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix} \]

where: \( M = \begin{bmatrix} A & C \\ C & B \end{bmatrix} \)
Harris corner detector

Consider all small shifts

\[ V(u, v) = A u^2 + 2C u v + B v^2 \]

\[ A = \sum_{x, y} w(x, y) I_x^2 (x, y) \]

\[ B = \sum_{x, y} w(x, y) I_y^2 (x, y) \]

\[ C = \sum_{x, y} w(x, y) I_x (x, y) I_y (x, y) \]

\[ M = \sum_{x, y} w(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \]
Harris corner detector

Intensity change in shifting window: eigenvalue analysis

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

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

Do you want isotropic or anisotropic?
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**
  - $\lambda_2 >> \lambda_1$;

- **Flat**
  - $\lambda_1$ and $\lambda_2$ are small;
  - $E$ is almost constant in all directions
Harris corner detector

Measure of corner response $R$:

$$R = \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2} = \frac{\text{det} \, M}{\text{Trace} \, M}$$
Harris Detector

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

Take only the points of local maxima of $R$
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: $I \rightarrow a \cdot I$

Thresholds?
Harris Detector: Some Properties

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

All points will be classified as *edges*.
Scale Invariant Detection

• Consider regions (e.g. circles) of different sizes around a point
• Regions of corresponding sizes will look the same in both images
Scale Invariant Detection

- The problem: how do we choose corresponding circles *independently* in each image?
- Choose the scale of the “best” corner (pyramids?)
Adaptive Non-maximal Suppression

- Desired: Fixed # of features per image
  - Want evenly distributed spatially…
  - Sort points by non-maximal suppression radius

[Brown, Szeliski, Winder, CVPR’05]
Feature descriptors

- We know how to detect points
- Next question: **How to match them?**
Feature descriptors

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

Point **descriptor** should be:

1. Invariant
2. Distinctive
Descriptors Invariant to Rotation

- Find local orientation
  Dominant direction of gradient

- Extract image patches relative to this orientation
Descriptor Vector

- Rotation Invariant Frame
- Orientation = blurred gradient
MOPS descriptor vector

- 8x8 oriented patch (Multi-scale oriented patch)
  - Sampled at 5 x scale (=40x40)
- Perform bias/gain normalization (mean 0, standard deviation 1)
Detections at multiple scales (pyramid)

Figure 1. Multi-scale Oriented Patches (MOPS) extracted at five pyramid levels from one of the Matier images. The boxes show the feature orientation and the region from which the descriptor vector is sampled.
Multi-Scale Oriented Patches (Summary)

• Interest points
  – Multi-scale Harris corners
  – Orientation from blurred gradient
  – Geometrically invariant to rotation

• Descriptor vector
  – Bias/gain normalized sampling of local patch (8x8)
  – Invariant to affine changes in intensity

• [Brown, Szeliski, Winder, CVPR’2005]
Invariance

- Good features should be robust to all sorts of nastiness that can occur between images.
Types of invariance

- Illumination
Types of invariance

- Illumination
- Scale
Types of invariance

• Illumination
• Scale
• Rotation
Types of invariance

- Illumination
- Scale
- Rotation
- Affine
Types of invariance

- Illumination
- Scale
- Rotation
- Affine
- Full Perspective