Image Features
So far

- In images (practically) the lowest level of organization (Pixel)
- Filtering (Extract structure from a collection of pixels)
- Convolution (a key operations) (Implemented in working systems through Fourier transform)
Seventeen Equations.....
Seventeen Equations.....
So far

- In images (practically) the lowest level of organization (Pixel)
- Filtering (Extract structure from a collection of pixels)
- Convolution (a key operations) (Implemented in working systems through Fourier transform)
- Smoothening (Gaussian)
- Edges (Good Abstraction)
- Role of Derivatives
- Scale of the images (Downsample Repeatedly Pyramid)
- Subsampling (Nyquist Rate)
- Upsampling (Interpolation)
- Interpolation also implemented through convolutions
Image Features

- What skyline is this?
Image Features

- What skyline is this?

NYC
What skyline is this?
How could we tell which type of scene this is from a movie?

What kind of scene is behind the actors?  
Kitchen? Bedroom? Street? Bar?
What Points to Choose for matching?

- Textureless patches are nearly impossible to localize.
- Patches with large contrast changes (gradients) are easier to localize.
- But straight line segments cannot be localized on lines segments with the same orientation (aperture problem)
- Gradients in at least two different orientations are easiest, e.g., corners!

[Adopted from: Szelski (Book)]
Aperture Problem

- “Corner-like” patch can be reliably matched
- A straight line patch can have multiple matches (Aperture Problem)
- Zero texture, useless, can have infinite matches

[Adopted from: Szelski (Book)]
How can we find corners in an image?
Interest Points: Corners

- 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.

Figure: (left) flat region: no change in all directions, (center) edge: no change along the edge direction, (right) corner: significant change in all directions

[Source: Alyosha Efros, Darya Frolova, Denis Simakov]
Interest Points: Corners

- Compare two image patches using (weighted) summed square difference
- Measures change in appearance of window $w(x, y)$ for the shift

$$E_{WSSD}(u, v) = \sum_x \sum_y w(x, y)[I(x + u, y + v) - I(x, y)]^2$$

Window function $w(x, y) = \begin{cases} 1 & \text{in window, 0 outside} \\ \text{Gaussian} & \text{or} \end{cases}$

[Source: J. Hays]
Interest Points: Corners

- Compare two image patches using (weighted) summed square difference
- Measures change in appearance of window $w(x, y)$ for the shift

$$E_{WSSD}(u, v) = \sum_x \sum_y w(x, y)[I(x + u, y + v) - I(x, y)]^2$$

[Source: J. Hays]
Interest Points: Corners

- Compare two image patches using (weighted) summed square difference
- Measures change in appearance of window \( w(x, y) \) for the shift

\[
E_{WSSD}(u, v) = \sum_x \sum_y w(x, y)[I(x + u, y + v) - I(x, y)]^2
\]

window function  shifted intensity  intensity
Interest Points: Corners

- Let’s look at $E_{WSSD}$
- We want to find out how this function behaves for small shifts

$$E(u, v)$$

- Remember our goal to detect corners:
Interest Points: Corners

- Using a simple first-order Taylor Series expansion about $x, y$:

$$I(x + u, y + v) \approx I(x, y) + u \cdot \frac{\partial I}{\partial x}(x, y) + v \cdot \frac{\partial I}{\partial y}(x, y)$$

- Using a series of polynomials to approximate $I$, more info on Taylor Series here

- And plugging it in our expression for $E_{\text{WSSD}}$:

$$E_{\text{WSSD}}(u, v) = \sum_x \sum_y w(x, y) \left( I(x + u, y + v) - I(x, y) \right)^2$$

$$\approx \sum_x \sum_y w(x, y) \left( I(x, y) + u \cdot I_x + v \cdot I_y - I(x, y) \right)^2$$

$$= \sum_x \sum_y \sum w(x, y) \left( u^2 I_x^2 + 2u \cdot v \cdot I_x \cdot I_y + v^2 I_y^2 \right)$$

$$= \sum_x \sum_y w(x, y) \cdot [u \ v] \begin{bmatrix} I_x^2 & I_x \cdot I_y \\ I_x \cdot I_y & I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$
Interest Points: Corners

- Since \((u, v)\) doesn't depend on \((x, y)\) we can rewrite it slightly:

\[
E_{WSSD}(u, v) = \sum_x \sum_y w(x, y) [u \ v] \begin{bmatrix} I_x^2 & I_x \cdot I_y \\ I_x \cdot I_y & I_y^2 \end{bmatrix} [u \ v]
\]

\[
= [u \ v] \left( \sum_x \sum_y w(x, y) \begin{bmatrix} I_x^2 & I_x \cdot I_y \\ I_x \cdot I_y & I_y^2 \end{bmatrix} \right) [u \ v]
\]

Let's denote this with \(M\)

\[
= [u \ v] M [u \ v]
\]

- \(M\) is a \(2 \times 2\) second moment matrix computed from image gradients:

\[
M = \sum_x \sum_y w(x, y) \begin{bmatrix} I_x^2 & I_x \cdot I_y \\ I_x \cdot I_y & I_y^2 \end{bmatrix}
\]
How Do I Compute $M$?

Let’s say I have this image

![Image](image.png)

Let’s say I have this image
How Do I Compute $M$?

Let’s say I have this image

I need to compute a $2 \times 2$ second moment matrix in each image location
How Do I Compute $M$?

Let’s say I have this image

I need to compute a $2 \times 2$ second moment matrix in each image location

In a particular location I need to compute $M$ as a weighted average of gradients in a window
How Do I Compute $M$?

Let's say I have this image

In a particular location I need to compute $M$ as a weighted average of gradients in a window

I can do this efficiently by computing three matrices, $I_x^2$, $I_y^2$ and $I_x \cdot I_y$, and convolving each one with a filter, e.g. a box or Gaussian filter.
Interest Points: Corners

- We now have $M$ computed in each image location
- Our $E_{WSSD}$ is a **quadratic function** where $M$ implies its shape

$$E_{WSSD}(u, v) = \begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix}$$

$$M = \sum_x \sum_y w(x, y) \begin{bmatrix} I_x^2 & I_x \cdot I_y \\ I_x \cdot I_y & I_y^2 \end{bmatrix}$$

[Source: J. Hays]
Let’s take a horizontal “slice” of $E_{WSSD}(u, v)$:

$$\begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix} = \text{const}$$

This is the equation of an ellipse

**Figure:** Different ellipses obtained by different horizontal “slices”
Interest Points: Corners

- Let’s take a horizontal “slice” of $E_{WSSD}(u, v)$:

  $$\begin{bmatrix} u & v \end{bmatrix} M \begin{bmatrix} u \\ v \end{bmatrix} = \text{const}$$

- This is the equation of an ellipse

**Figure:** Different ellipses obtained by different horizontal “slices”
Interest Points: Corners

- Our matrix $M$ is symmetric:

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

- And thus we can diagonalize it (in Matlab: $[V, D] = \text{eig}(M)$):

\[
M = V \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} V^{-1}
\]

- Columns of $V$ are major and minor axes of ellipse, the lengths of the radii proportional to $\lambda^{-1/2}$

[Source: J. Hays]
Interest Points: Corners

- The eigenvalues of $M$ ($\lambda_1$, $\lambda_2$) reveal the amount of intensity change in the two principal orthogonal gradient directions in the window.

[Source: R. Szeliski, slide credit: R. Urtasun]
Interest Points: Corners

“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
\[ \lambda_1 \text{ and } \lambda_2 \text{ are small;} \]

[Source: K. Grauman, slide credit: R. Urtasun]
Interest Points: Criteria to Find Corners

- Harris and Stephens, ’88, is rotationally invariant and downweighs edge-like features where $\lambda_1 \gg \lambda_0$

$$R = \lambda_0 \lambda_1 - \alpha (\lambda_0 + \lambda_1)^2 = \det(M) - \alpha \cdot \text{trace}(M)^2$$

- **Why** go via det and trace and not use a criteria with $\lambda$?
- $\alpha$ a constant (0.04 to 0.06)

- The corresponding detector is called **Harris corner detector**

- \text{"edge":} $R < 0$
- \text{"corner":} $R > 0$
- \text{"flat" region} $|R|$ small
Interest Points: Criteria to Find Corners

- Harris and Stephens, 88 is rotationally invariant and downweighs edge-like features where $\lambda_1 \gg \lambda_0$

$$R = \lambda_0\lambda_1 - \alpha(\lambda_0 + \lambda_1)^2 = \det(M) - \alpha \text{trace}(M)^2$$

- Shi and Tomasi, 94 proposed the smallest eigenvalue of $A$, i.e., $\lambda_0^{-1/2}$.

- Triggs, 04 suggested

$$\lambda_0 - \alpha \lambda_1$$

also reduces the response at 1D edges, where aliasing errors sometimes inflate the smaller eigenvalue

- Brown et al, 05 use the harmonic mean

$$\frac{\det(A)}{\text{trace}(A)} = \frac{\lambda_0\lambda_1}{\lambda_0 + \lambda_1}$$

[Source Mubarak Shah, Szelski]
Harris Corner detector

1. Compute gradients $I_x$ and $I_y$

2. Compute $I_x^2$, $I_y^2$, $I_x \cdot I_y$

3. Average (Gaussian) → gives $M$ per voxel

4. Compute
   \[ R = \det(M) - \alpha \text{trace}(M)^2 \]
   for each image window (cornerness score)

5. Find points with large $R$ ($R >$ threshold).

6. Take only points of local maxima, i.e., perform non-maximum suppression
Example

[Source: K. Grauman]
1) Compute Cornerness

[Source: K. Grauman]
2) Find High Response

[Source: K. Grauman]
3) Non-maxima Suppression

[Source: K. Grauman]
Results

[Source: K. Grauman]
Another Example

[Source: K. Grauman]
Cornerness

[Source: K. Grauman]
Interest Points

[Source: K. Grauman]
Properties of Harris Corner Detector

- Rotation and Shift Invariance of Corners

- Second moment ellipse rotates but its shape (i.e. eigenvalues) remains the same
- Harris corner detector is rotation-covariant

[Source: J. Hays]
Properties of Harris Corner Detector

- Scale?

Corner

All points will be classified as edges

- Corner location is **not scale invariant/covariant**!

[Source: J. Hays]
Can we also define keypoints that are shift, rotation and scale invariant/covariant?

What should be our description around keypoint?