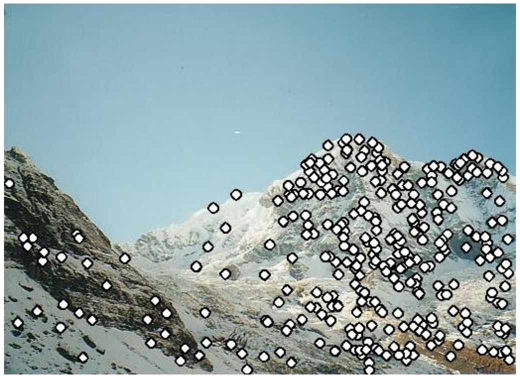
Interest Point Detection

Lecture-4

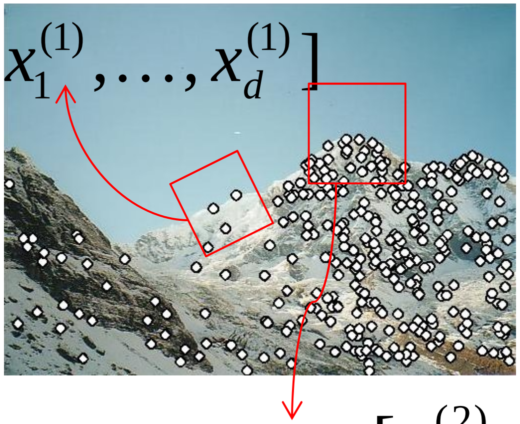
Local features: main components

1) **Detection:** Identify the interest points



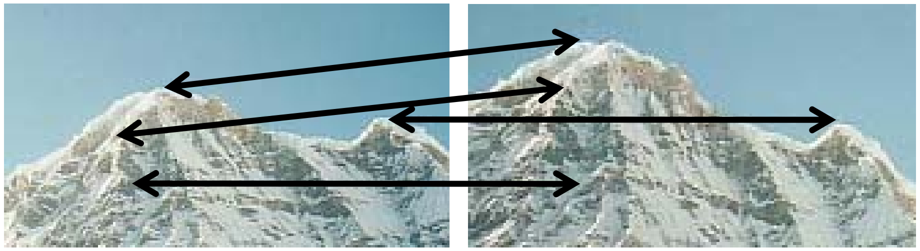
2) **Description** :Extract feature vector descriptor surrounding each interest point.

$$\mathbf{x}_1 = [x_1^{(1)}, \ldots, x_d^{(1)}]$$



$$\mathbf{x}_2 = [x_1^{(2)}, \ldots, x_d^{(2)}]$$

3) **Matching:** Determine correspondence between descriptors in two views



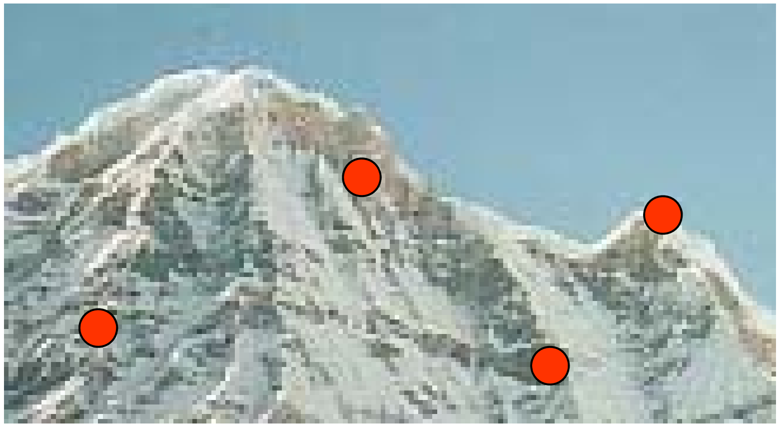
Kristen Grauman

Where can we use it?

- Automate object tracking
- Point matching for computing disparity
- Stereo calibration
  - Estimation of fundamental matrix
- Motion based segmentation
- Recognition
- 3D object reconstruction
- Robot navigation
- Image retrieval and indexing

Goal: interest operator repeatability

- We want to detect (at least some of) the same points in both images.



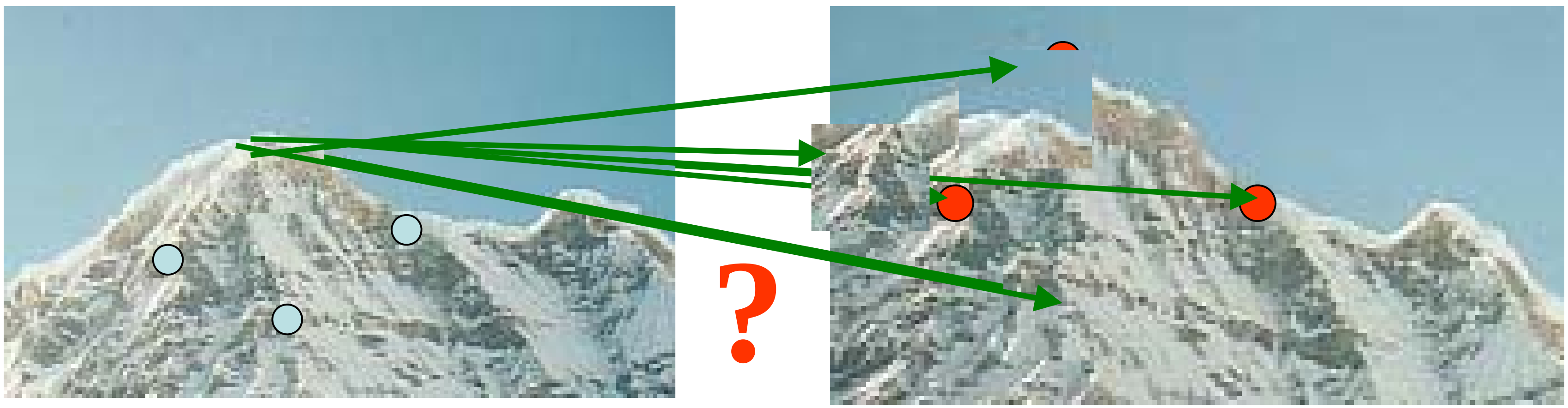
No chance to find true matches!

- Yet we have to be able to run the detection procedure *independently* per image.

Kristen Grauman

Goal: descriptor distinctiveness

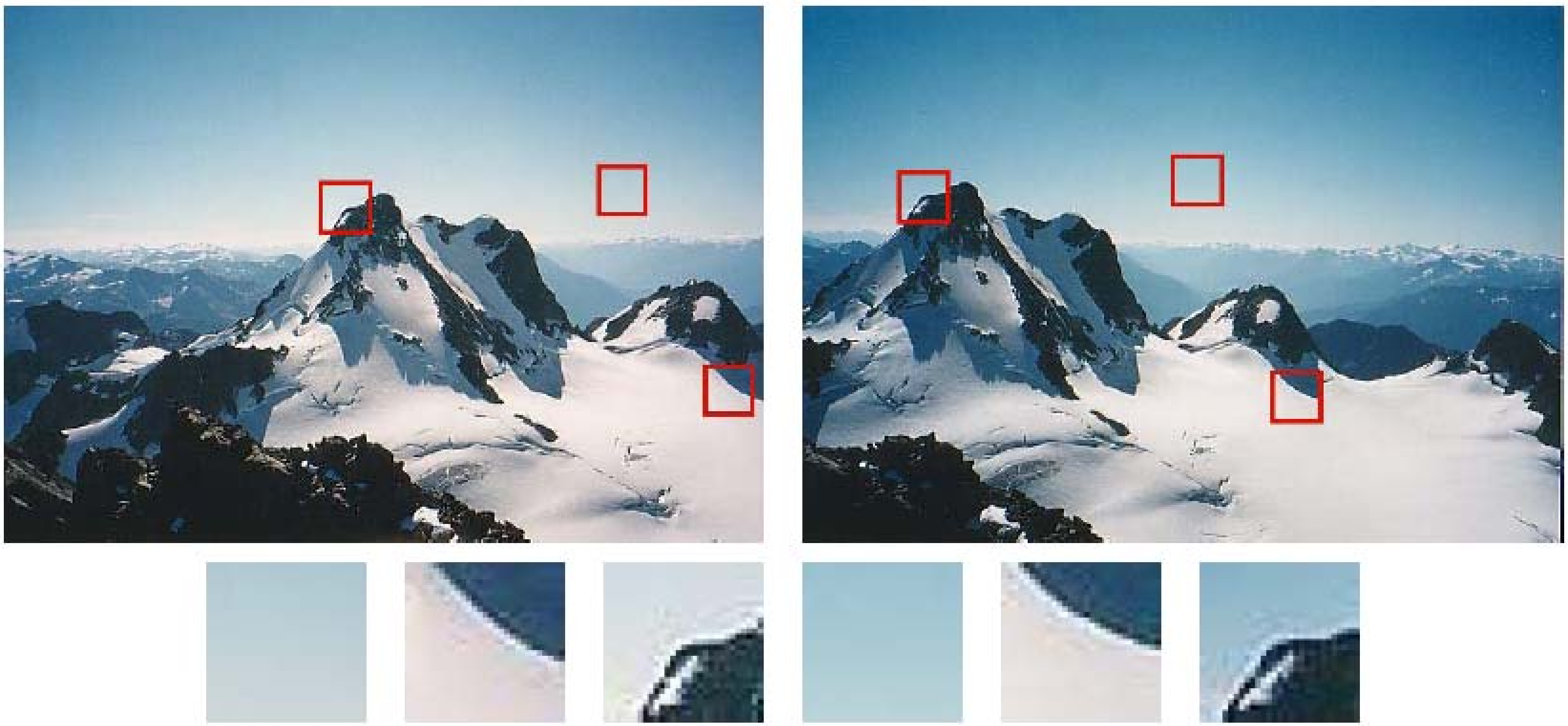
- We want to be able to reliably determine which point goes with which.



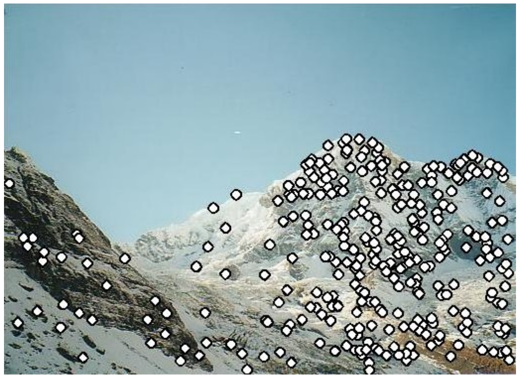
- Must provide some invariance to geometric and photometric differences between the two views.

Kristen Grauman

Some patches can be localized or matched with higher accuracy than others.



Local features: main components



1) Detection: Identify the interest points

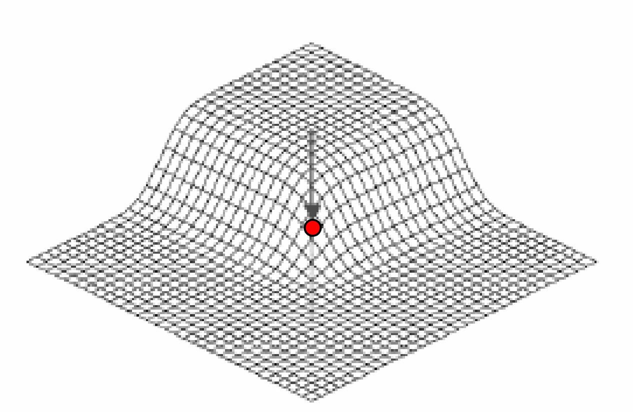
2) Description: Extract vector feature descriptor surrounding each interest point.

3) Matching: Determine correspondence between descriptors in two views

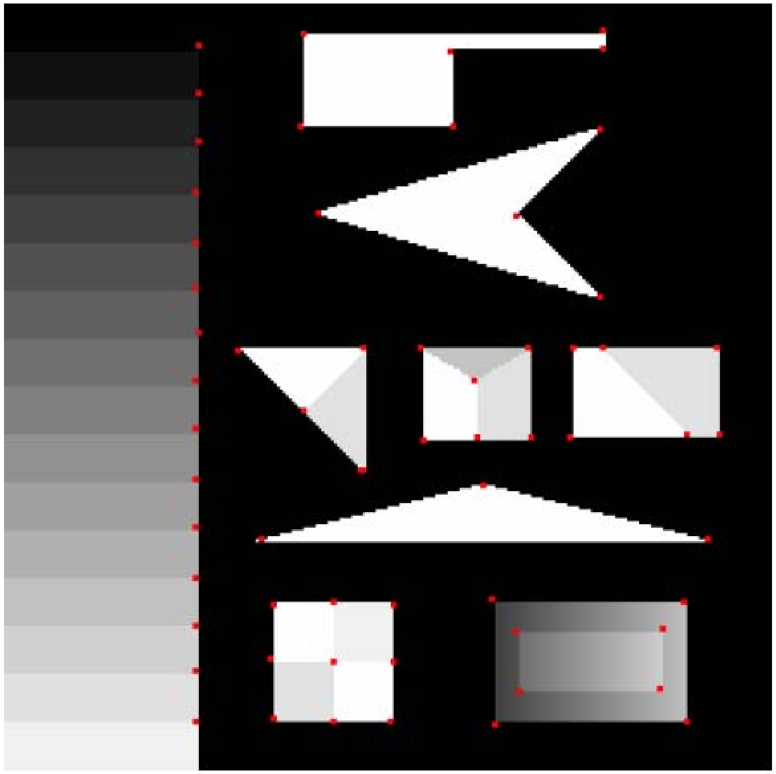
Kristen Grauman

What is an interest point

- Expressive texture
  - The point at which the direction of the boundary of object changes abruptly
  - Intersection point between two or more edge segments



Synthetic & Real Interest Points



Corners are indicated in red

Properties of Interest Point Detectors

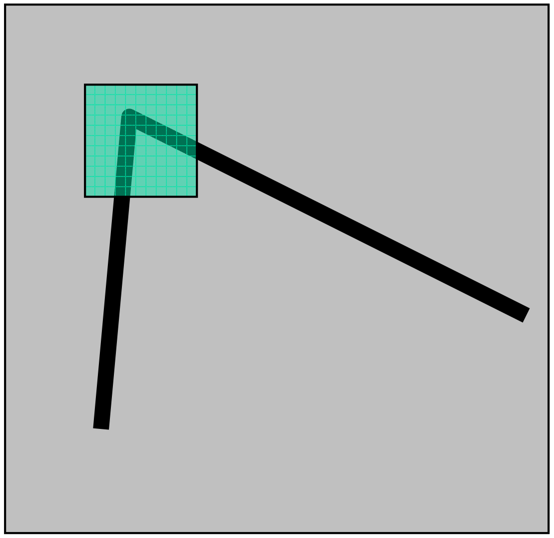
- Detect all (or most) true interest points
- No false interest points
- Well localized.
- Robust with respect to noise.
- Efficient detection

Possible Approaches to Corner Detection

- Based on brightness of images
  - Usually image derivatives
- Based on boundary extraction
  - First step edge detection
  - Curvature analysis of edges

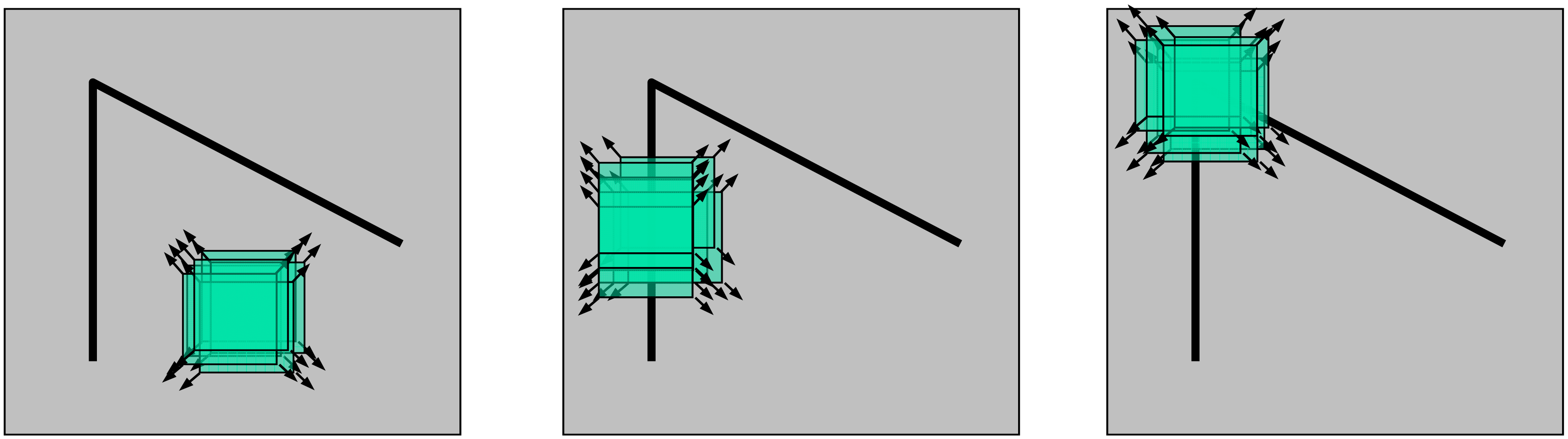
Harris Corner Detector

- Corner point can be recognized in a window
- Shifting a window in any direction should give a large change in intensity



C.Harris, M.Stephens. “A Combined Corner and Edge Detector”. 1988

Basic Idea

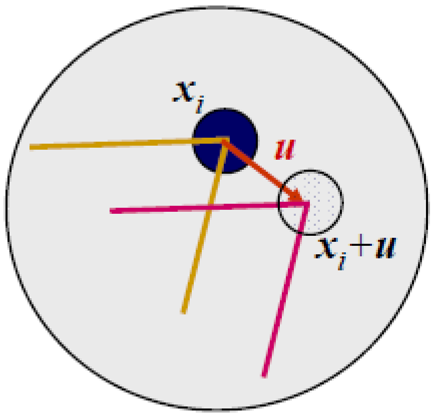


“flat” region:
no change in
all directions

“edge”:
no change along
the edge direction

“corner”:
significant change
in all directions

Aperture Problem



(a)

Correlation

$$f \otimes h = \sum_k \sum_l f(k,l)\,h(i+k, j+l)$$

f = Image

h = Kernel

f

f_1	f_2	f_3
f_4	f_5	f_6
f_7	f_8	f_9

$\otimes$

h

h_1	h_2	h_3
h_4	h_5	h_6
h_7	h_8	h_9

$$\begin{aligned} f * h = {} & f_1 h_1 + f_2 h_2 + f_3 h_3 \\ & + f_4 h_4 + f_5 h_5 + f_6 h_6 \\ & + f_7 h_7 + f_8 h_8 + f_9 h_9 \end{aligned}$$

Correlation

$$f \otimes h = \sum_k \sum_l f(k,l)\, h(i+k, j+l)$$

Cross correlation

$$f \otimes f = \sum_k \sum_l f(k,l)\, f(i+k, j+l)$$

Auto correlation

Correlation Vs SSD

$$SSD = \sum_k \sum_l \left(f(k,l) - h(i+k, j+l) \right)^2$$ Sum of Squares Difference

minimize

$$SSD = \sum_k \sum_l \left(f(k,l)^2 - 2h(i+k, j+l) f(k,l) + h(i+k, j+l)^2 \right)$$

minimize

$$SSD = \sum_k \sum_l \left(-2h(i+k, j+l) f(k,l) \right)$$

$$SSD = \sum_k \sum_l \left(2h(i+k, j+l) f(k,l) \right)$$

maximize

$$Correlation = \sum_k \sum_l \left(h(i+k, j+l) f(k,l) \right)$$

maximize

$$f \otimes f = \sum_k \sum_l f(k,l) f(i+k, j+l)$$

Mathematics of Harris Detector

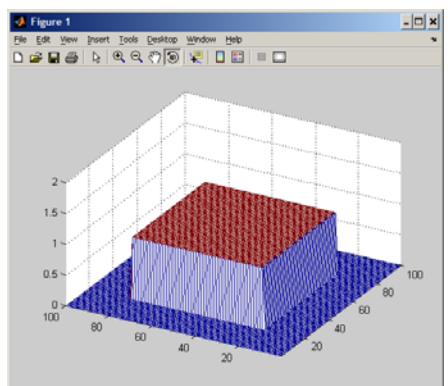
- Change of intensity for the shift (u,v)

$$E(u,v) = \sum_{x,y} \quad [\underbrace{I(x+u, y+v)}_{\text{shifted intensity}} - \underbrace{I(x,y)}_{\text{intensity}}]^2$$

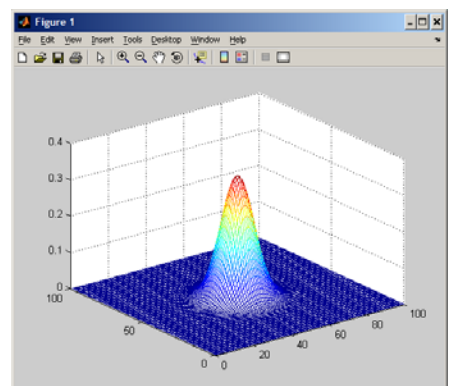
n

Auto-correlation

Window functions →

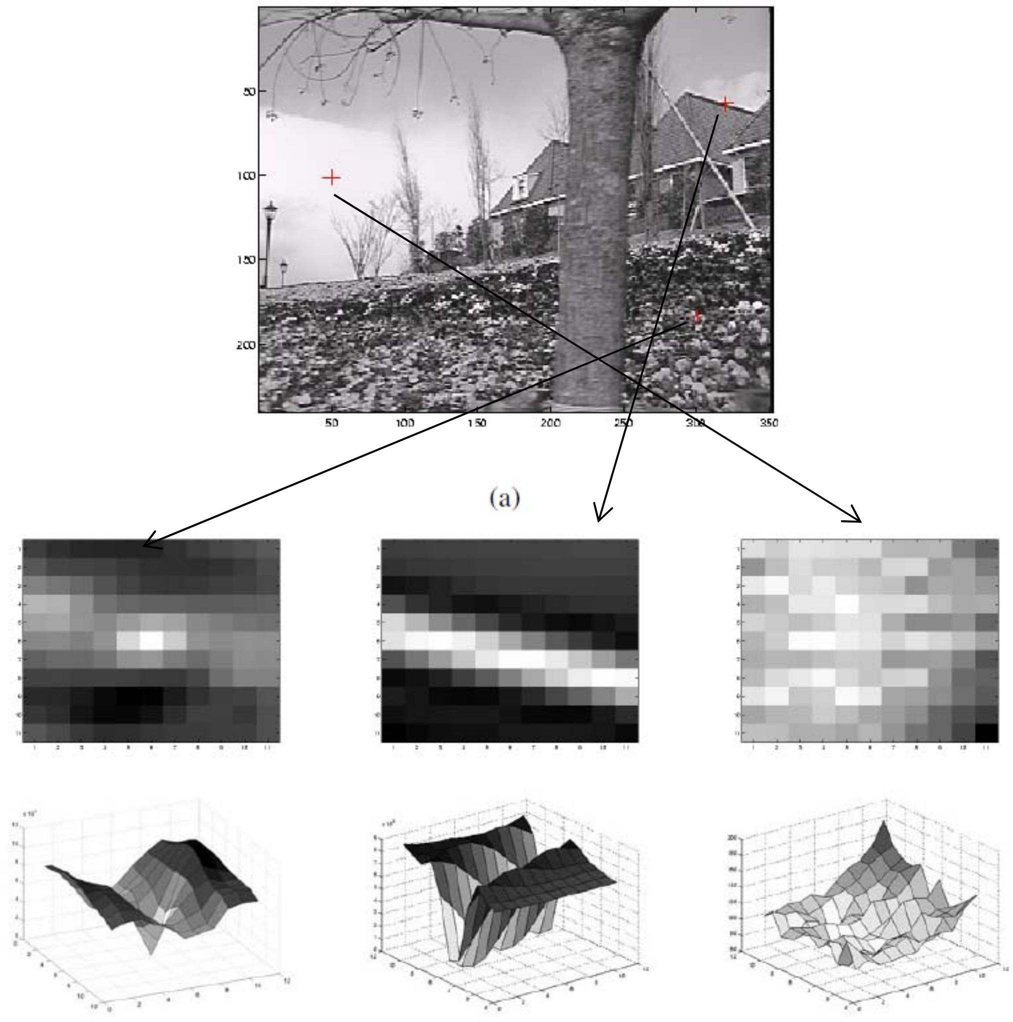


UNIFORM



GAUSSIAN

Auto-Correlation



Taylor Series

f(*x*) Can be represented at point *a* in terms of its derivatives

$$f(a)+\frac{f'(a)}{1!}(x-a)+\frac{f''(a)}{2!}(x-a)^2+\frac{f^{(3)}(a)}{3!}(x-a)^3+\cdots.$$

Mathematics of Harris Detector

$$E(u,v) = \sum_{x,y} \underbrace{w(x,y)}_{\text{window function}} \; [\underbrace{I(x+u, y+v)}_{\text{shifted intensity}} - \underbrace{I(x,y)}_{\text{intensity}}]^2$$

$$E(u,v) = \sum_{x,y} \underbrace{w(x,y)}_{\text{window function}} \; [\underbrace{I(x,y) + uI_x + vI_y}_{\text{shifted intensity}} - \underbrace{I(x,y)}_{\text{intensity}}]^2$$

Taylor Series

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

$$E(u,v) = \sum_{x,y} w(x,y)\left[\begin{pmatrix} u & v \end{pmatrix}\begin{pmatrix} I_x \\ I_y \end{pmatrix}\right]^2$$

$$E(u,v) = \sum_{x,y} w(x,y)\begin{pmatrix} u & v \end{pmatrix}\begin{pmatrix} I_x \\ I_y \end{pmatrix}\begin{pmatrix} I_x & I_y \end{pmatrix}\begin{pmatrix} u \\ v \end{pmatrix}$$

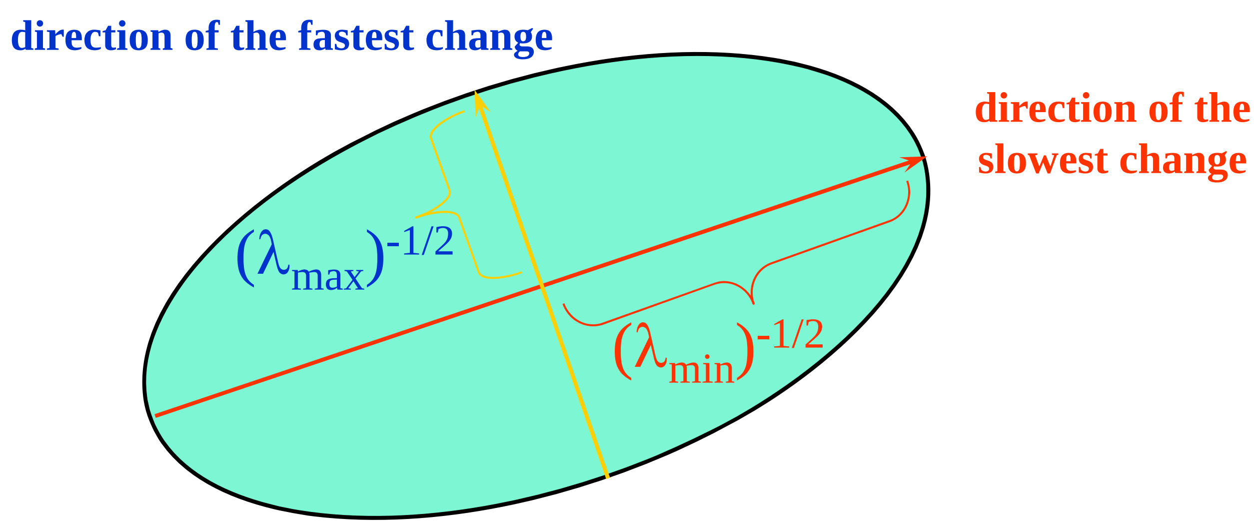
$$E(u,v) = \begin{pmatrix} u & v \end{pmatrix}\left[\sum_{x,y} w(x,y)\begin{pmatrix} I_x \\ I_y \end{pmatrix}\begin{pmatrix} I_x & I_y \end{pmatrix}\right]\begin{pmatrix} u \\ v \end{pmatrix}$$

$$E(u,v) = \begin{pmatrix} u & v \end{pmatrix} M \begin{pmatrix} u \\ v \end{pmatrix}$$

Mathematics of Harris Detector

$$E(u,v) = (u \quad v) M \begin{pmatrix} u \\ v \end{pmatrix} \qquad M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x I_x & I_x I_y \\ I_x I_y & I_y I_y \end{bmatrix}$$

- E(u,v) is an equation of an ellipse, where M is the covariance
- Let λ_1 and λ_2 be eigenvalues of M



Eigen Vectors and Eigen Values

The eigen vector, x, of a matrix A is a special vector, with the following property

$$Ax = \lambda x$$

Where λ is called eigen value

To find eigen values of a matrix A first find the roots of:

$$\det(A - \lambda I) = 0$$

Then solve the following linear system for each eigen value to find corresponding eigen vector

$$(A - \lambda I)x = 0$$

Example

$$A = \begin{bmatrix} -1 & 2 & 0 \\ 0 & 3 & 4 \\ 0 & 0 & 7 \end{bmatrix}$$

Eigen Values

$$\lambda_1 = 7,\ \lambda_2 = 3, \lambda_3 = -1$$

$$\mathbf{x_1} = \begin{bmatrix} 1 \\ 4 \\ 4 \end{bmatrix},\ \mathbf{x_2} = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix},\ \mathbf{x_3} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

Eigen Vectors

Eigen Values

$$\det(A-\lambda I)=0$$

$$\det\left(\begin{bmatrix}-1 & 2 & 0\\ 0 & 3 & 4\\ 0 & 0 & 7\end{bmatrix}-\lambda\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}\right)=0$$

$$\det\left(\begin{bmatrix}-1-\lambda & 2 & 0\\ 0 & 3-\lambda & 4\\ 0 & 0 & 7-\lambda\end{bmatrix}\right)=0$$

$$(-1-\lambda)((3-\lambda)(7-\lambda)-0)=0$$

$$(-1-\lambda)(3-\lambda)(7-\lambda)=0$$

$$\lambda=-1,\quad \lambda=3,\quad \lambda=7$$

Eigen Vectors

$\lambda = -1$

$$(A - \lambda I)x = 0$$

$$\left(\begin{bmatrix} -1 & 2 & 0 \\ 0 & 3 & 4 \\ 0 & 0 & 7 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\right)\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 2 & 0 \\ 0 & 4 & 4 \\ 0 & 0 & 8 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$0 + 2x_2 + 0 \;= 0$$

$$0 + 4x_2 + 4x_3 = 0$$

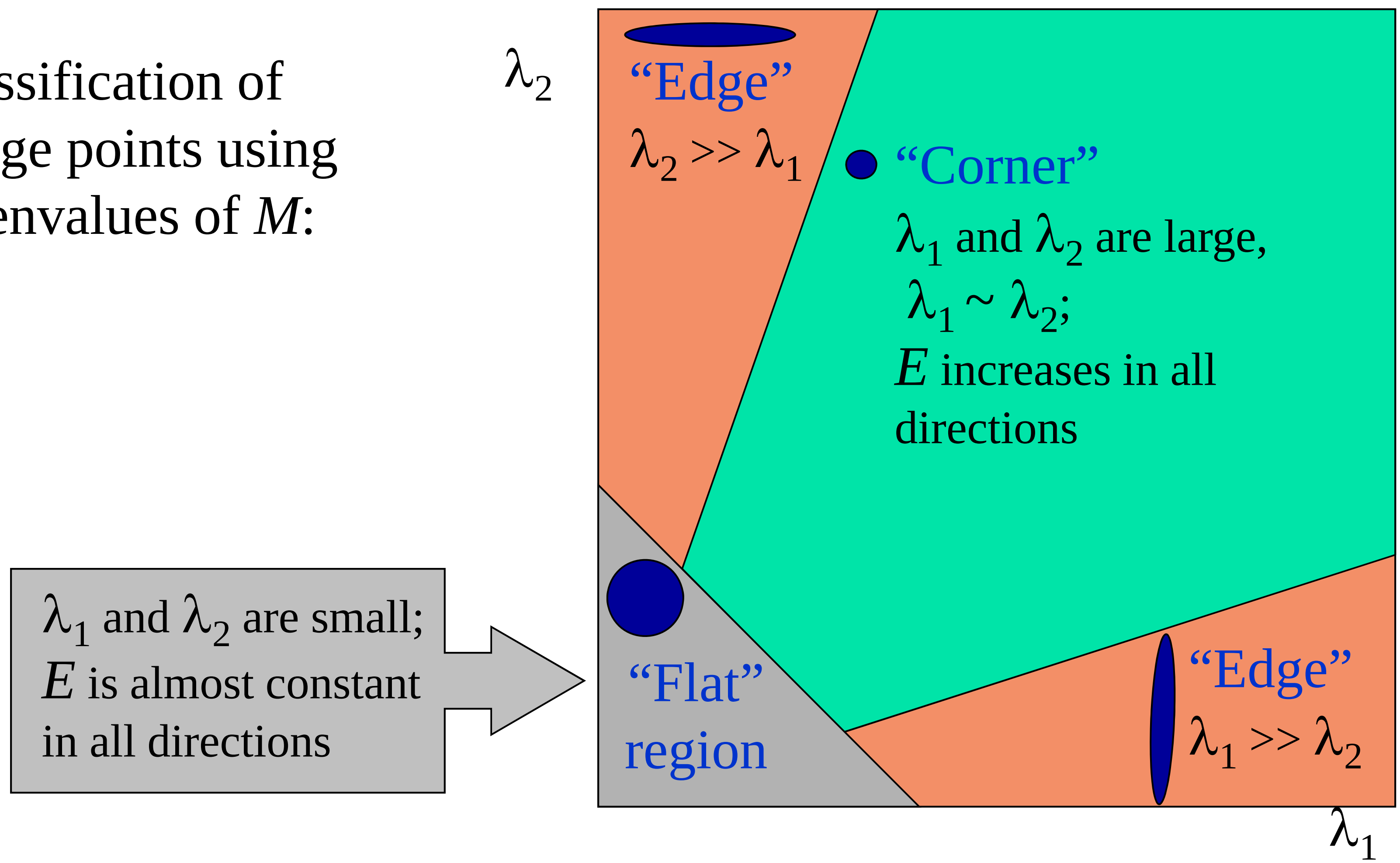
$$0 + 0 + 8x_3 = 0$$

$$x_1 = 1, \quad x_2 = 0, \quad x_3 = 0$$

$$\mathbf{x_1} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

Mathematics of Harris Detector

Classification of image points using eigenvalues of *M*:



Mathematics of Harris Detector

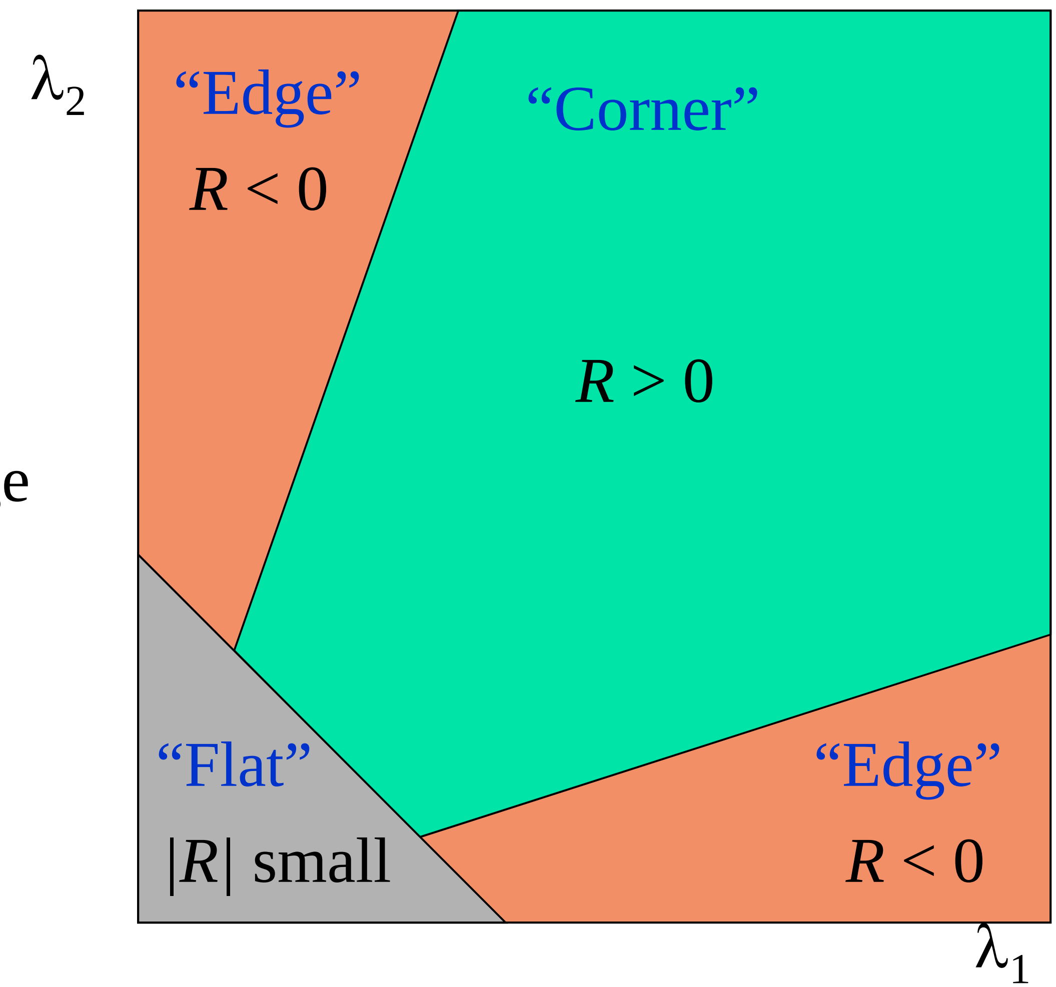
- Measure of cornerness in terms of λ_1, λ_2

$$M = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$

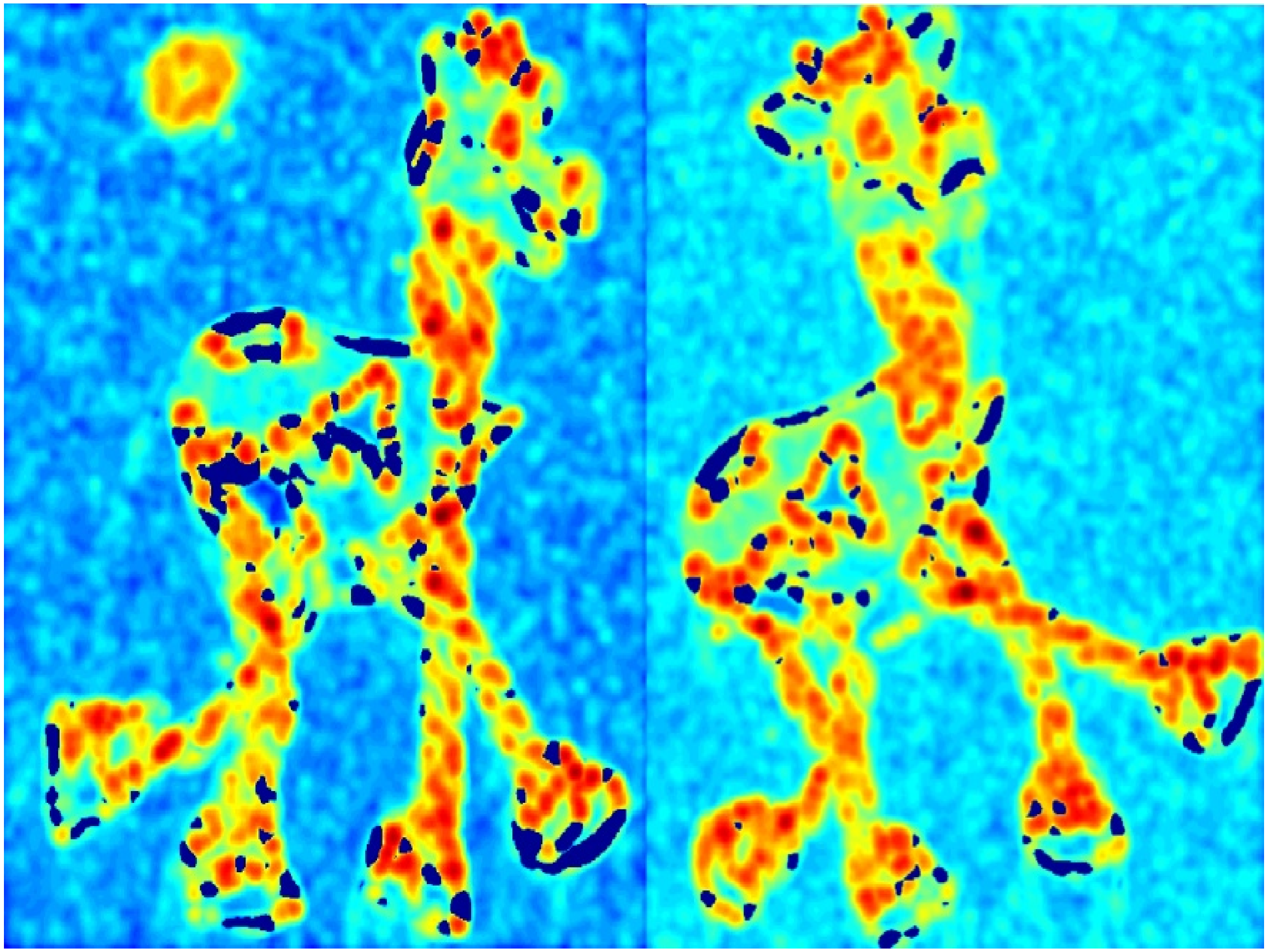
$$R = \det M - k(trace M)^2 \qquad R = \lambda_1 \lambda_2 - k(\lambda_1 + \lambda_2)^2$$

Mathematics of Harris Detector

- 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

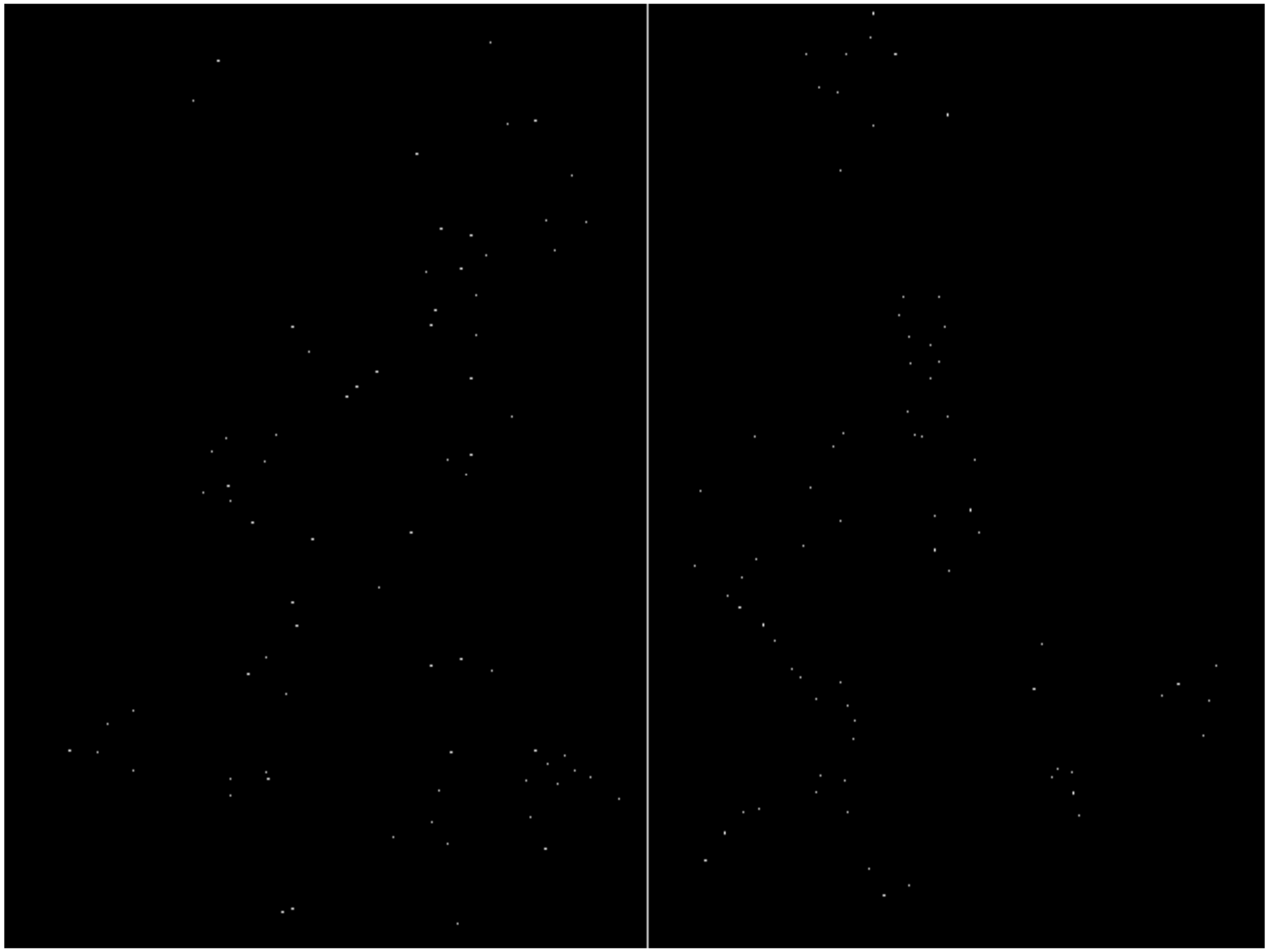


Compute corner response



Find points with large corner response: *R*> threshold

Take only the points of local maxima of *R*



If pixel value is greater than its neighbors then it is a local maxima.

Other Version of Harris Detectors

$$R = \lambda_1 - \alpha \lambda_2$$

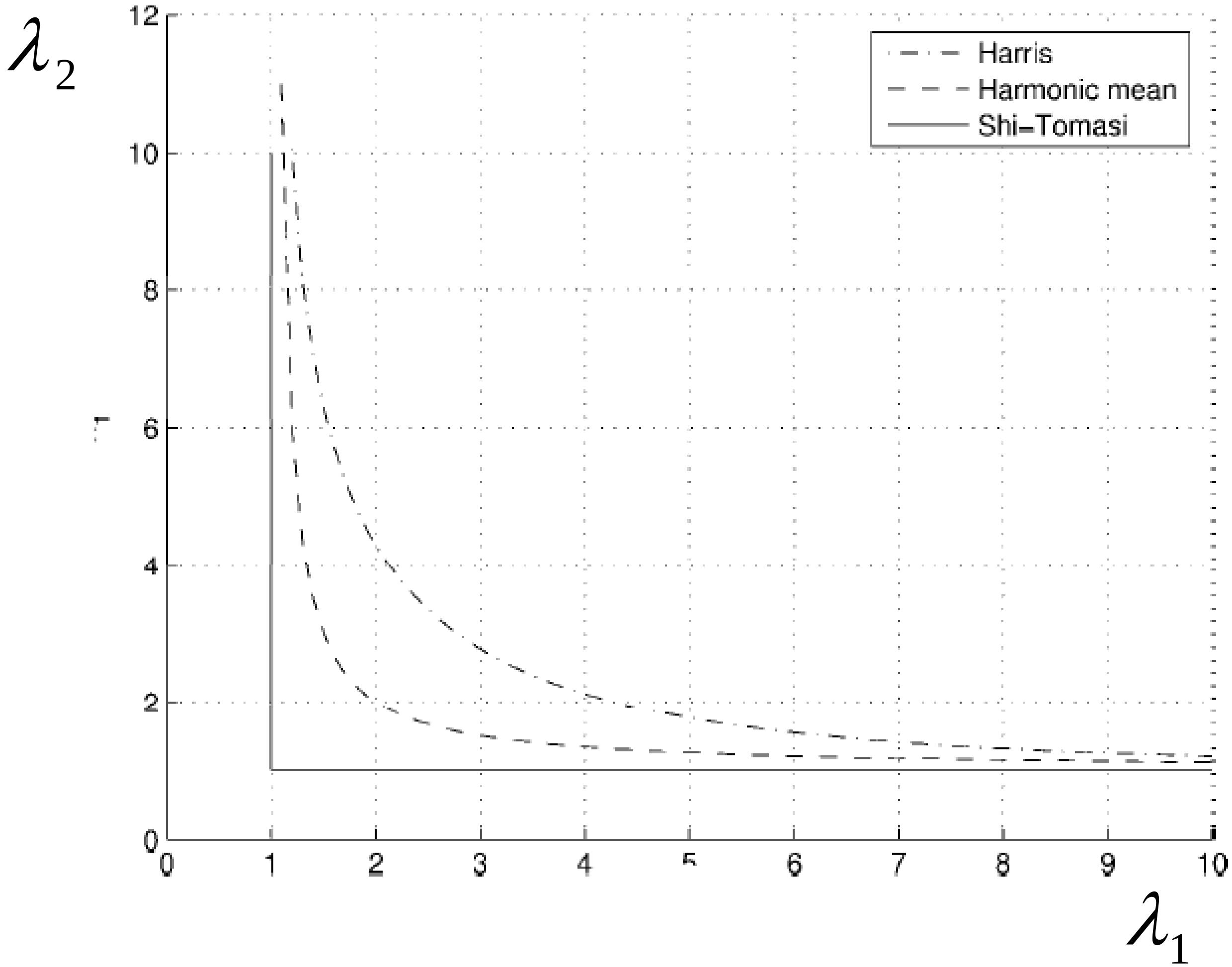
Triggs

$$R = \frac{\det(M)}{trace(M)_1} = \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2}$$

Szeliski (Harmonic mean)

$$R = \lambda_1$$

Shi-Tomasi



λ_2
12
10
8
6
4
2
0
0 1 2 3 4 5 6 7 8 9 10
Harris
Harmonic mean
Shi–Tomasi
λ_1

Algorithm

- Compute horizontal and vertical derivatives of image I_x and I_y.
- Compute three images corresponding to three terms in matrix *M*.
- Convolve these three images with a larger Gaussian (window).
- Compute scalar cornerness value using one of the *R* measures.
- Find local maxima above some threshold as detected interest points.

Reading Material

- Section 4.1.1 Feature Detectors
  - Richard Szeliski, "Computer Vision: Algorithms and Application“, Springer.