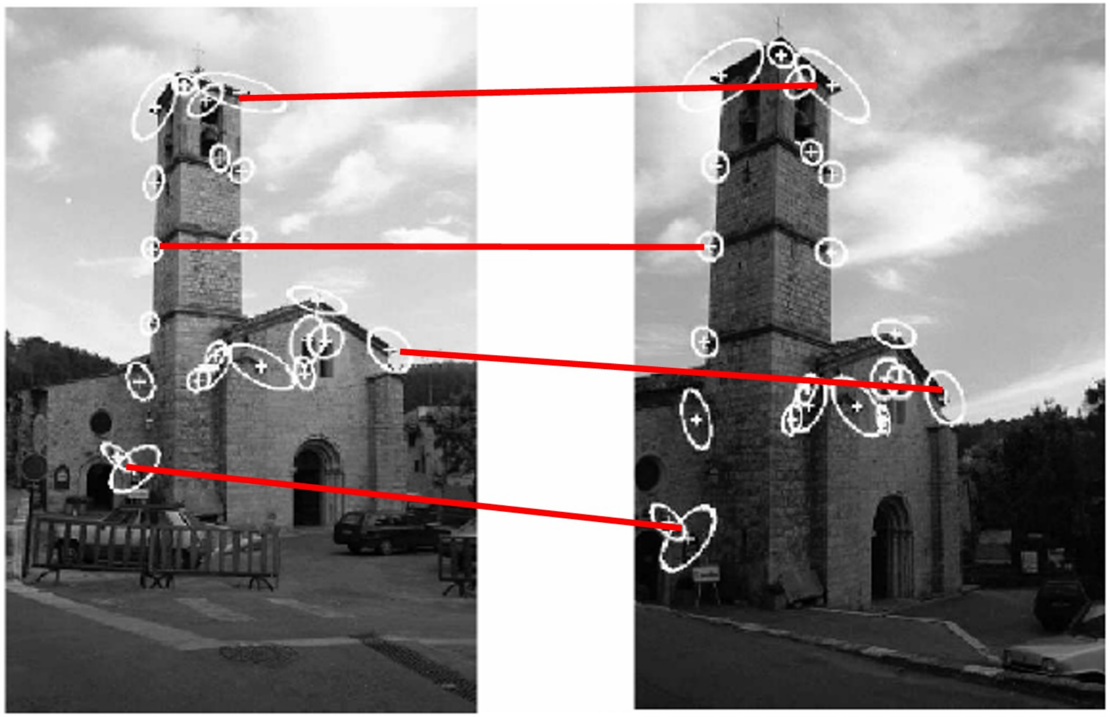
CSE 185 Introduction to Computer Vision
Lecture 4: Corner Detector

Slides credit: Yuri Boykov, Ming-Hsuan Yang, Robert Collins, Richard Szeliski, Steve Seitz, Alyosha Efros, Fei-Fei Li, etc.

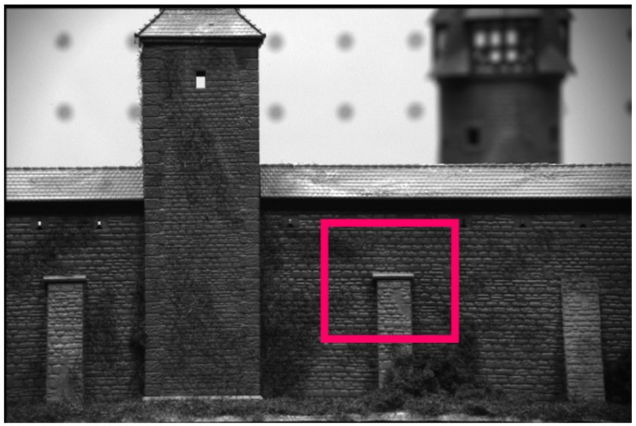
Motivation for feature points

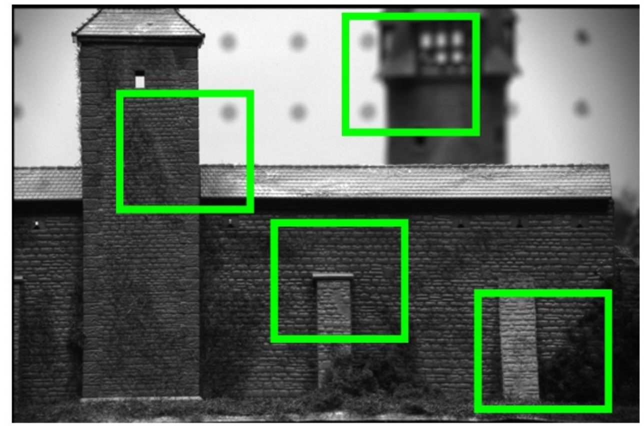
Many applications require
generic “discriminant” feature points with
<u>identifiable appearance and location</u>
(so that they can be matched across multiple images)



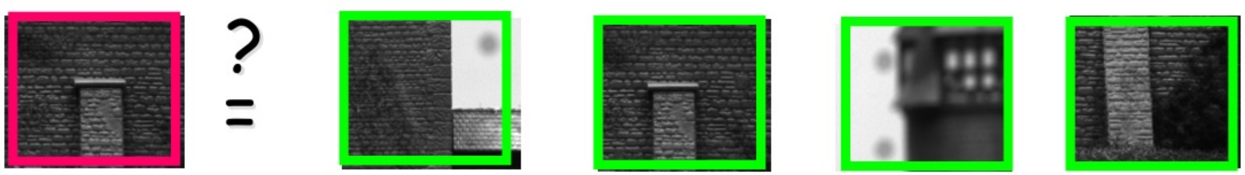
Motivation: patch matching

- Elements to be matched are image patches of fixed size





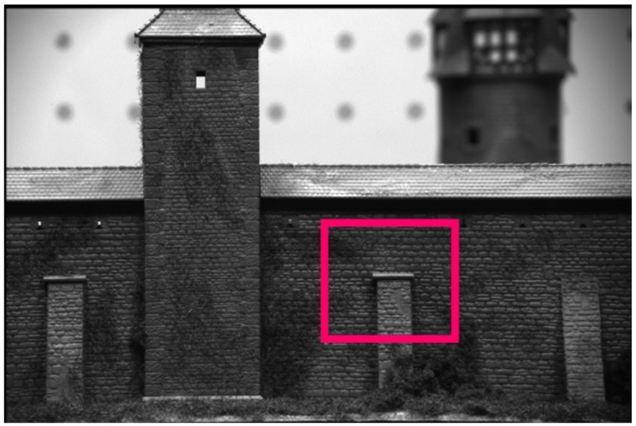
Task: find the best (most similar) patch in a second image

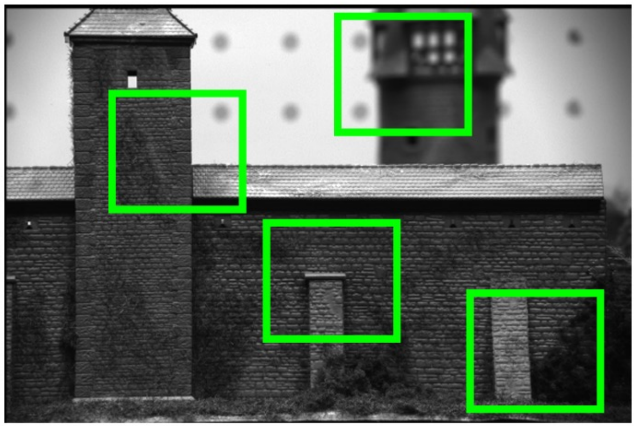


slide credit: Robert Collins

Not all patches are created equal!

- Elements to be matched are image patches of fixed size



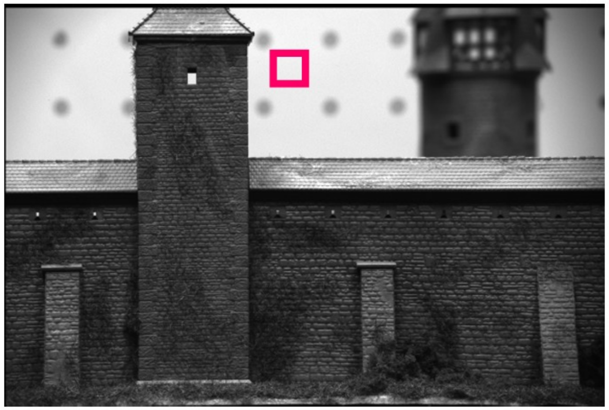


Intuition: this would be a good patch for matching, since it is very distinctive (there is only one patch in the second frame that looks similar).

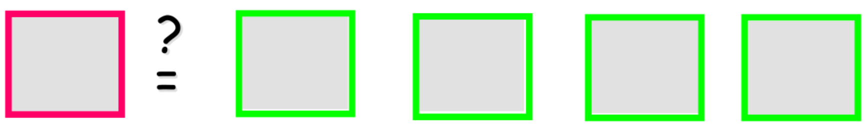
slide credit: Robert Collins

Not all patches are created equal!

- Elements to be matched are image patches of fixed size

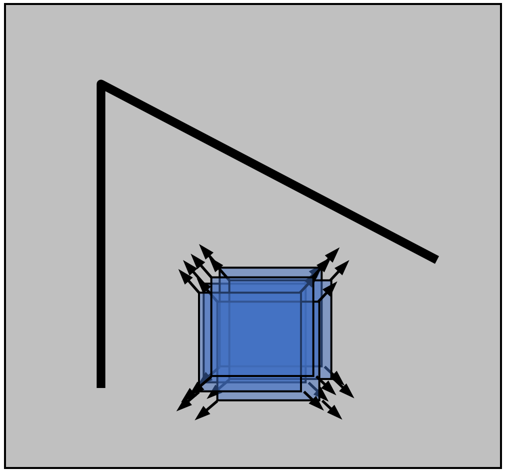


Intuition: this would be a BAD patch for matching, since it is not very distinctive

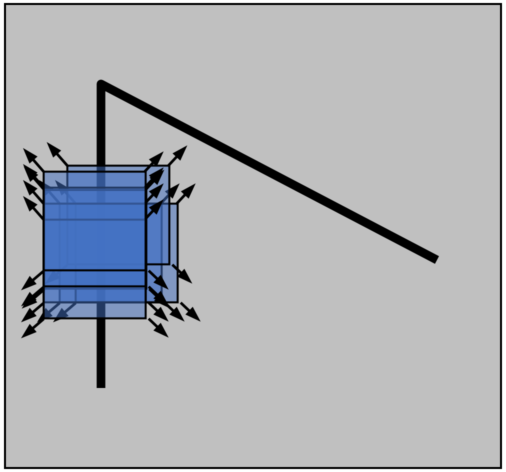


slide credit: Robert Collins

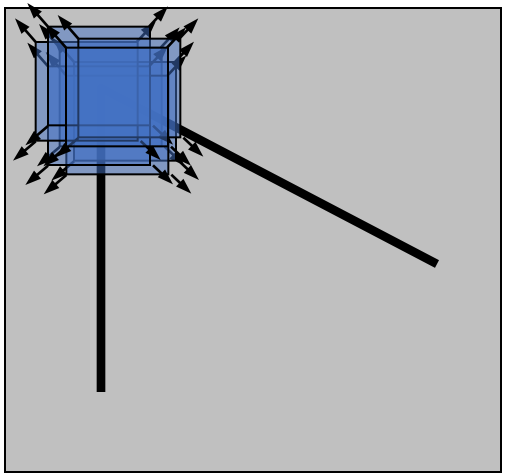
Harris detector: basic idea



“flat” region:
no change in all directions



“edge”:
no change along the edge direction



“corner”:
significant change in all directions

Harris detector: mathematics

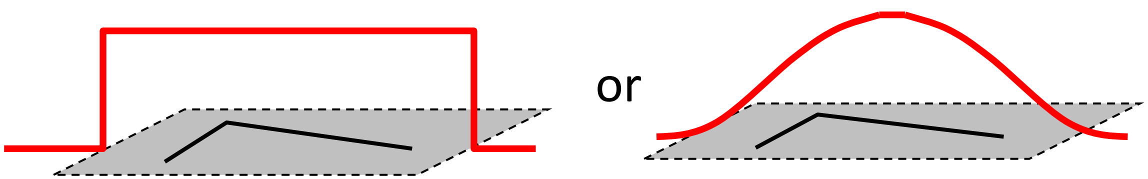
patch *w* change measure for shift $ds = \begin{bmatrix} u \\ v \end{bmatrix}$:

weighted sum of squared differences

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

Window function — Shifted intensity — Intensity

NOTE:
window support functions *w(x,y)* =



1 in window, 0 outside

Gaussian (weighted) support

Harris detector: mathematics

Change of intensity for the shift $ds = \begin{bmatrix} u \\ v \end{bmatrix}$ assuming image gradient $\nabla I \equiv \begin{bmatrix} I_x \\ I_y \end{bmatrix}$

$$I(x+u, y+v) - I(x,y) \approx I_x \cdot u + I_y \cdot v = ds^T \cdot \nabla I$$

difference/change in *I* at (x,y) for shift (u,v) = ds

(remember gradient definition on earlier slides!!!!)

this is 1st order Taylor expansion

$$[\, I(x+u, y+v) - I(x,y) \,]^2 \approx ds^T \cdot \nabla I \cdot \nabla I^T \cdot ds$$

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

$$\approx ds^T \cdot \left(\sum_{x,y} w(x,y) \cdot \nabla I \cdot \nabla I^T \right) \cdot ds = ds^T \cdot M_w \cdot ds$$

M_w

Harris detector: mathematics

Change of intensity for the shift $ds = \begin{bmatrix} u \\ v \end{bmatrix}$ assuming image gradient $\nabla I \equiv \begin{bmatrix} I_x \\ I_y \end{bmatrix}$

$$E_w(u,v) \cong [u \;\; v] \cdot M_w \cdot \begin{bmatrix} u \\ v \end{bmatrix} = ds^T \cdot M_w \cdot ds = E_w(ds)$$

where M_w is a 2×2 matrix computed from image derivatives inside patch w

matrix M is also called
Harris matrix or ***structure tensor***

$$\ldots \cdot \left(\sum_{x,y} w(x,y) \cdot \underbrace{\nabla I \cdot \nabla I^T}_{\begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{bmatrix}} \right) \cdot \ldots$$

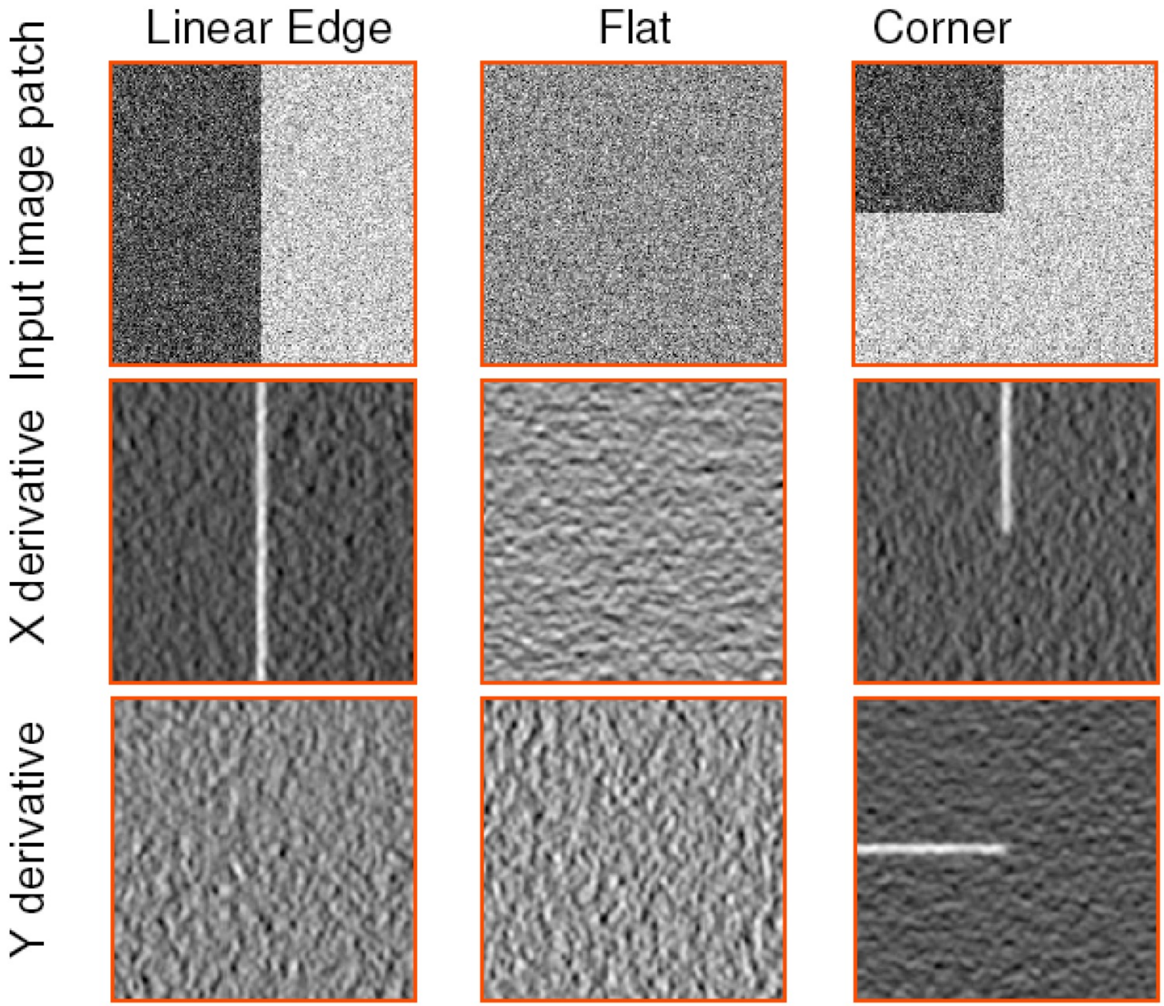
M_w

This tells you how to compute M_w at any window w (t.e. any image patch)

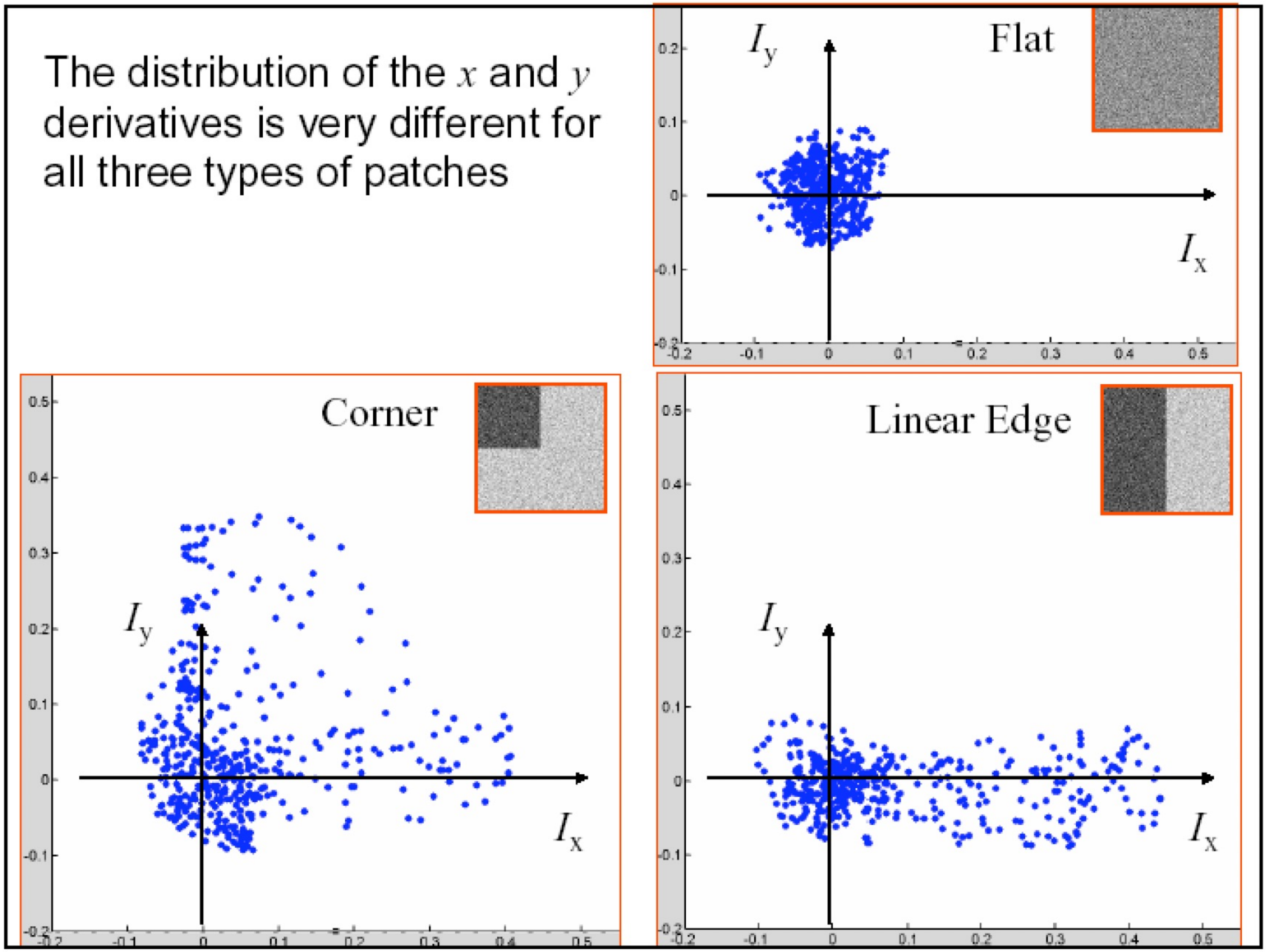
Intuitive way to understand Harris

- Treat gradient vectors as a set of (dx,dy) points with a center of mass defined as being at (0,0).
- Fit an ellipse to that set of points via scatter matrix
- Analyze ellipse parameters for varying cases…

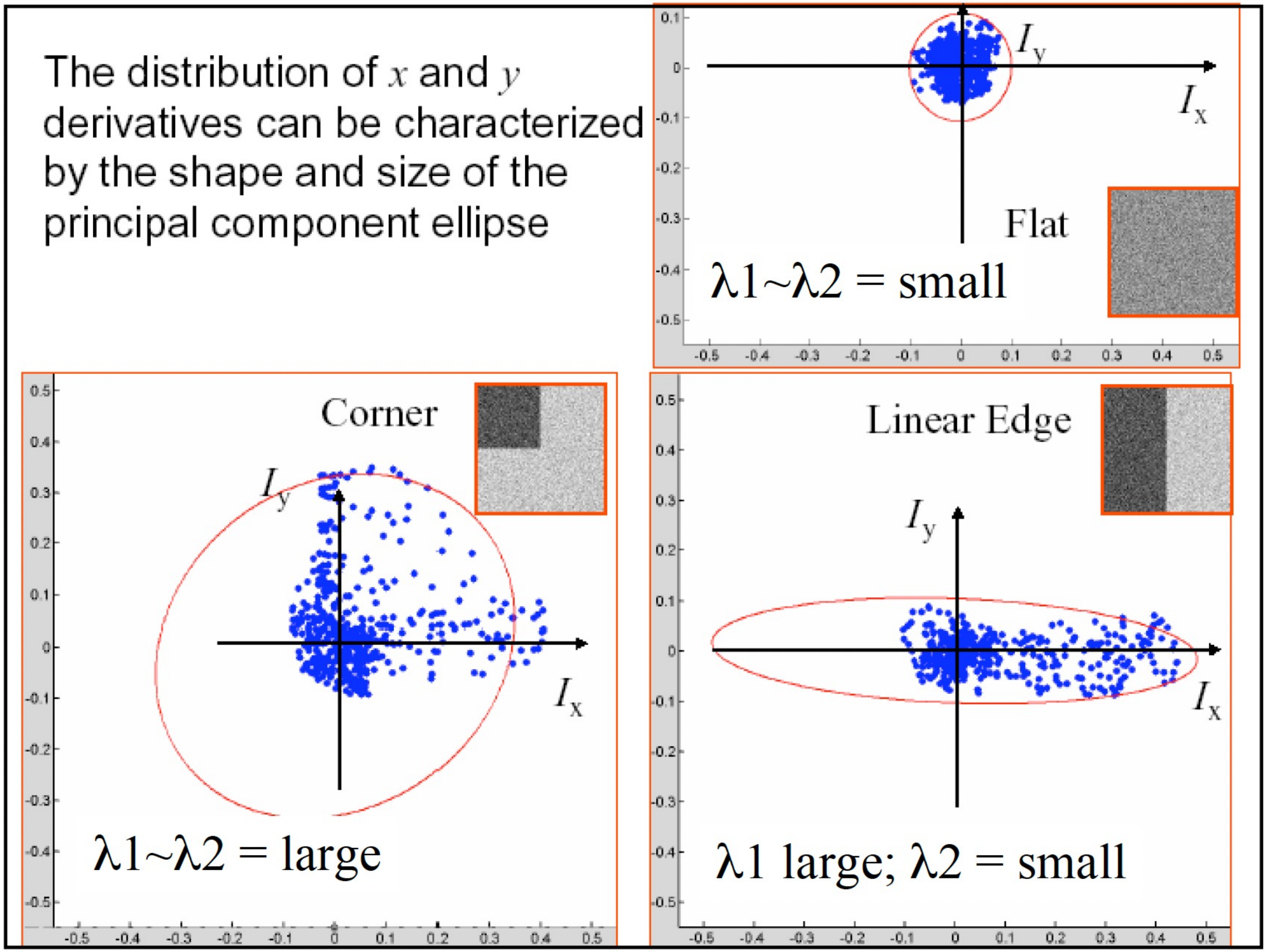
Example: cases and 2D derivatives



Plotting derivatives as 2D points

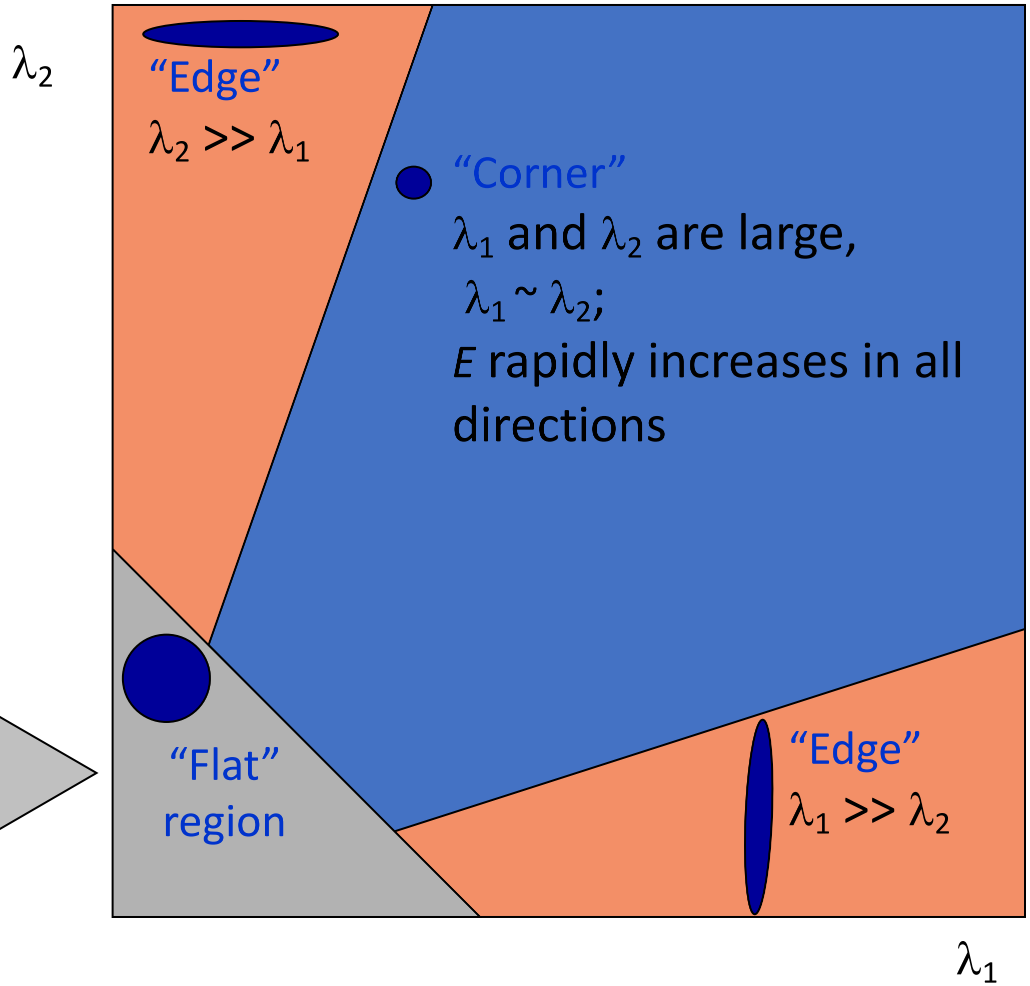


Fitting ellipse to 2D points



Classification via eigenvalues

Classification of image points using eigenvalues of *M*:



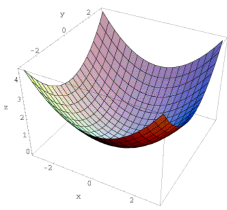
λ_1 and λ_2 are small;
E is almost constant in all directions

Harris detector: mathematics

Change of intensity for the shift $ds = \begin{bmatrix} u \\ v \end{bmatrix}$ assuming image gradient $\nabla I \equiv \begin{bmatrix} I_x \\ I_y \end{bmatrix}$

$$E_w(u,v) \cong [u \;\; v] \cdot M_w \cdot \begin{bmatrix} u \\ v \end{bmatrix} = ds^T \cdot M_w \cdot ds$$

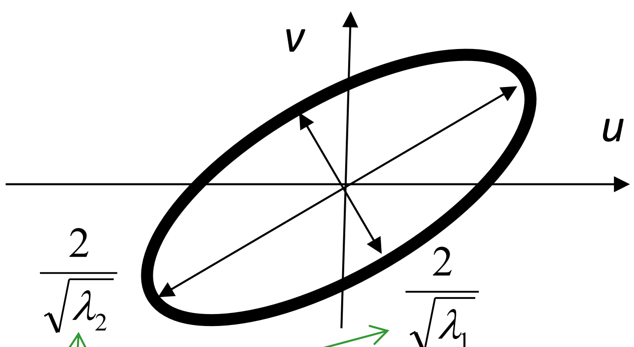
paraboloi



M is a *positive semi-definite* (p.s.d.) matrix (**Exercise**: show that $ds^T \cdot M \cdot ds \geq 0$ for any *ds*)

M can be analyzed via ***isolines***, *e.g.* $ds^T \cdot M_w \cdot ds = 1$ (**ellipsoid**)

see next slide



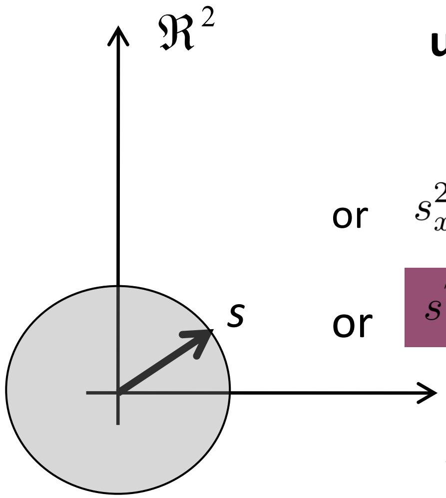
$\frac{2}{\sqrt{\lambda_2}}$ $\frac{2}{\sqrt{\lambda_1}}$

two eigen values of matrix M_w

Points on this ellipsoid are shifts $ds=[u,v]^T$ giving the same sum of squared differences $E(u,v)=1$.

$\Rightarrow$ This isoline visually illustrates how differences E depend on shifts $ds=[u,v]^T$ in different directions.

Note for linear algebra: Ellipsoids

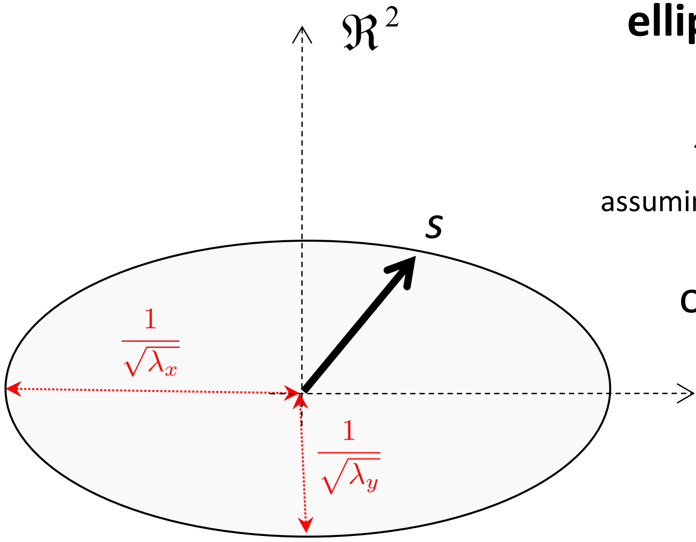


unit circle

$$\|s\| \;=\; 1$$

or $s_x^2 + s_y^2 = 1$

or $s^T \cdot s \;=\; 1$



ellipsoid aligned with coordinate axes

$$\lambda_x s_x^2 + \lambda_y s_y^2 = 1$$

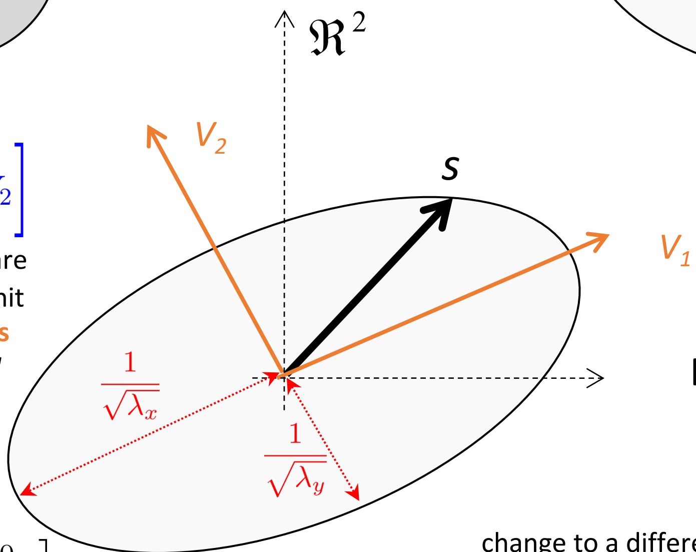
assuming non-negative λ_x and λ_y

or $s^T \cdot \Lambda \cdot s \;=\; 1$

for $\Lambda \;=\; \begin{bmatrix} \lambda_x & 0 \\ 0 & \lambda_y \end{bmatrix}$

$V \;=\; \left[V_1 \middle| V_2 \right]$

two columns are orthogonal unit **eigenvectors** of matrix *M*



eigenvalues of matrix *M*

$\Lambda \;=\; \begin{bmatrix} \lambda_x & 0 \\ 0 & \lambda_y \end{bmatrix}$

ellipsoid
with general principal axes

$s^T \cdot M \cdot s \;=\; 1$ for any p.s.d. matrix *M*

Explanation: $M \;=\; V \Lambda V^T$ (*eigendecomposition* for any p.s.d. M)

$$s^T \cdot M \cdot s \;=\; 1 \quad \Rightarrow \quad s^T \cdot V \cdot \Lambda \cdot V^T \cdot s \;=\; 1$$

$$\Rightarrow \quad (V^T s)^T \cdot \Lambda \cdot (V^T s) \;=\; 1$$

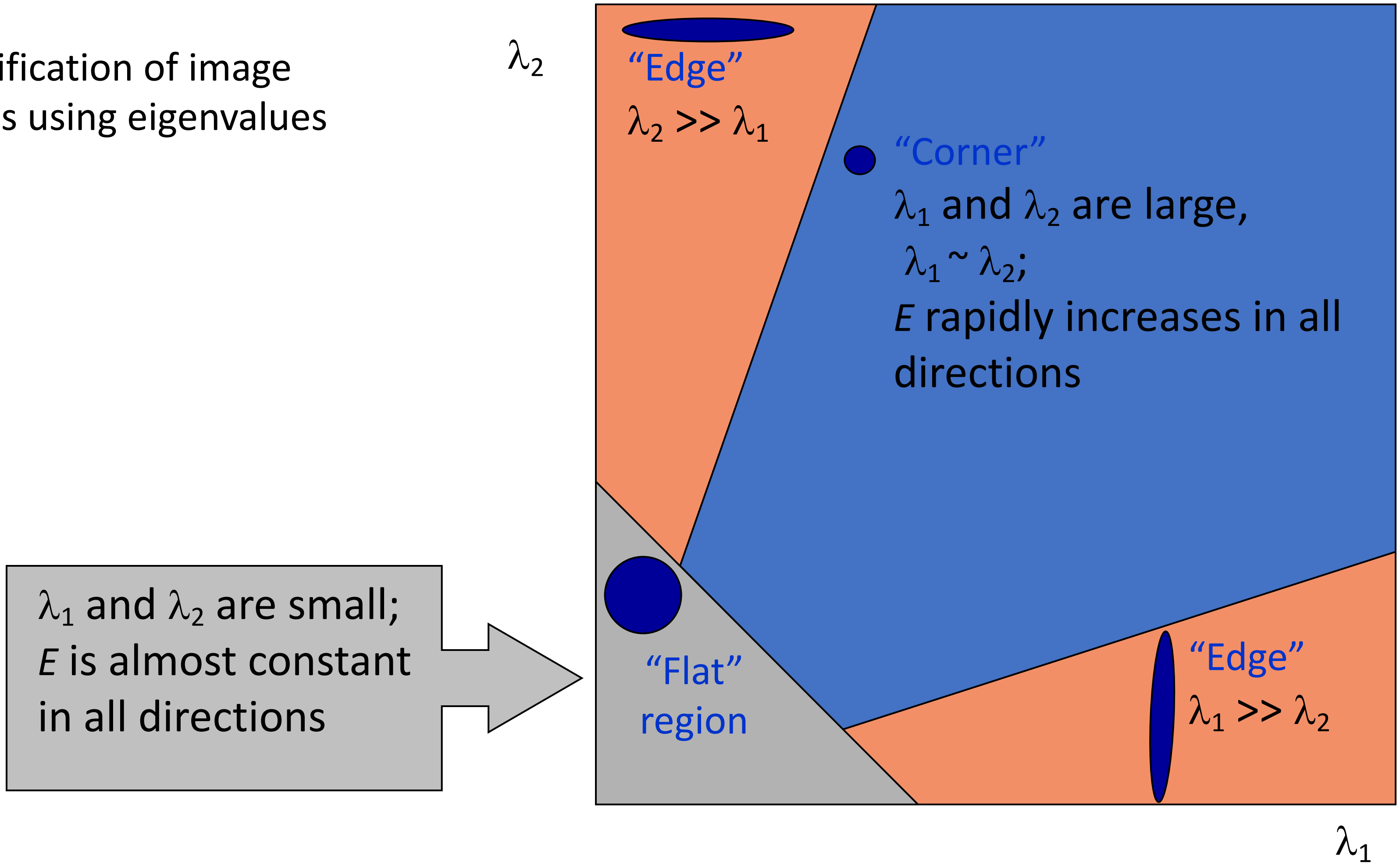
$$\Rightarrow \quad t^T \cdot \Lambda \cdot t \;=\; 1$$

change to a different orthogonal coordinate basis $t \;=\; V^T \cdot s$ (more in next topic)

rotation

Harris detector: mathematics

Classification of image points using eigenvalues of *M*:

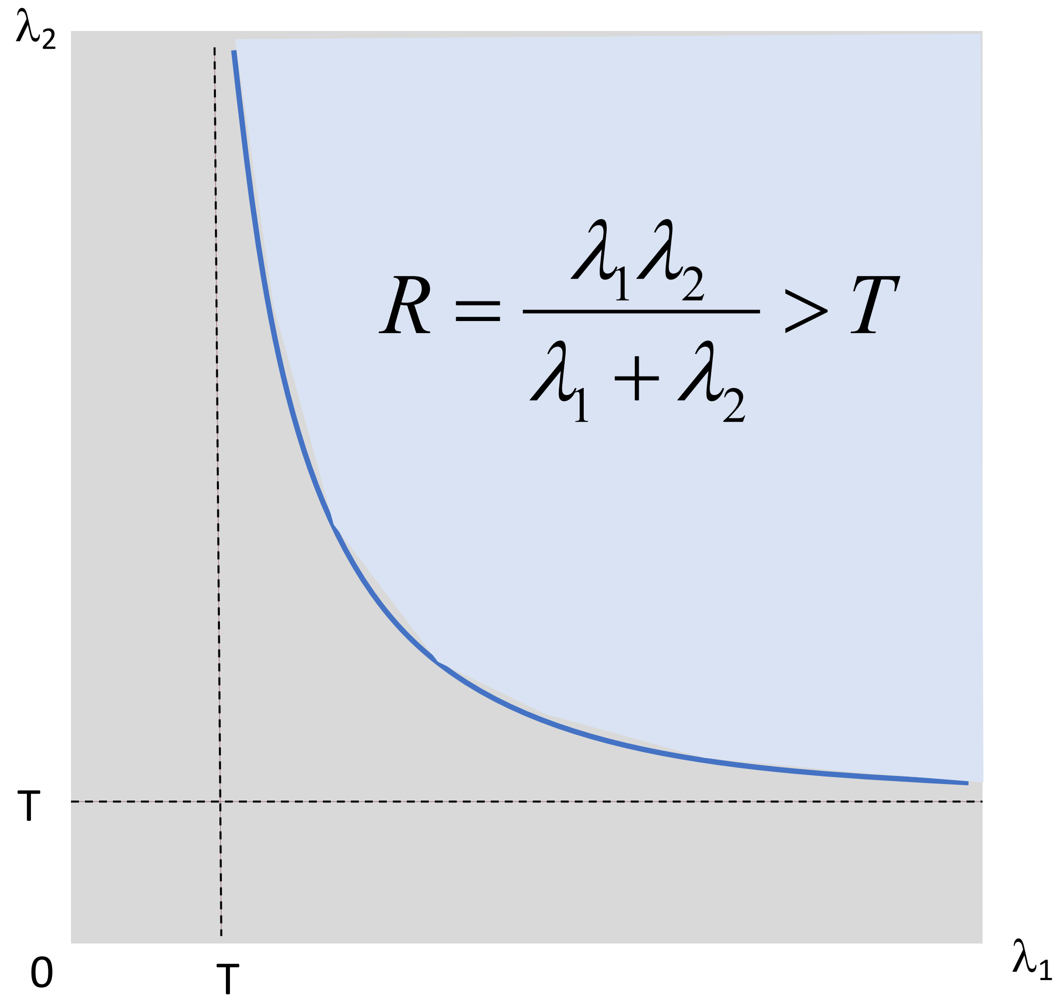


Harris detector: mathematics

One common measure of corner response:

$$R = \frac{\det M}{\operatorname{Trace} M}$$

$$\det M = \lambda_1 \lambda_2$$
$$\operatorname{trace} M = \lambda_1 + \lambda_2$$



Harris corner detection algorithm

1. Compute x and y derivatives of image

$$I_x = G_\sigma^x * I \quad I_y = G_\sigma^y * I$$

2. Compute products of derivatives at every pixel

$$I_{x2} = I_x . I_x \quad I_{y2} = I_y . I_y \quad I_{xy} = I_x . I_y$$

3. Compute the sums of the products of derivatives at each pixel

$$S_{x2} = G_{\sigma'} * I_{x2} \quad S_{y2} = G_{\sigma'} * I_{y2} \quad S_{xy} = G_{\sigma'} * I_{xy}$$

4. Define at each pixel (x, y) the matrix

$$H(x, y) = \begin{bmatrix} S_{x2}(x, y) & S_{xy}(x, y) \\ S_{xy}(x, y) & S_{y2}(x, y) \end{bmatrix}$$

5. Compute the response of the detector at each pixel

$$R = Det(H) - k(Trace(H))^2$$

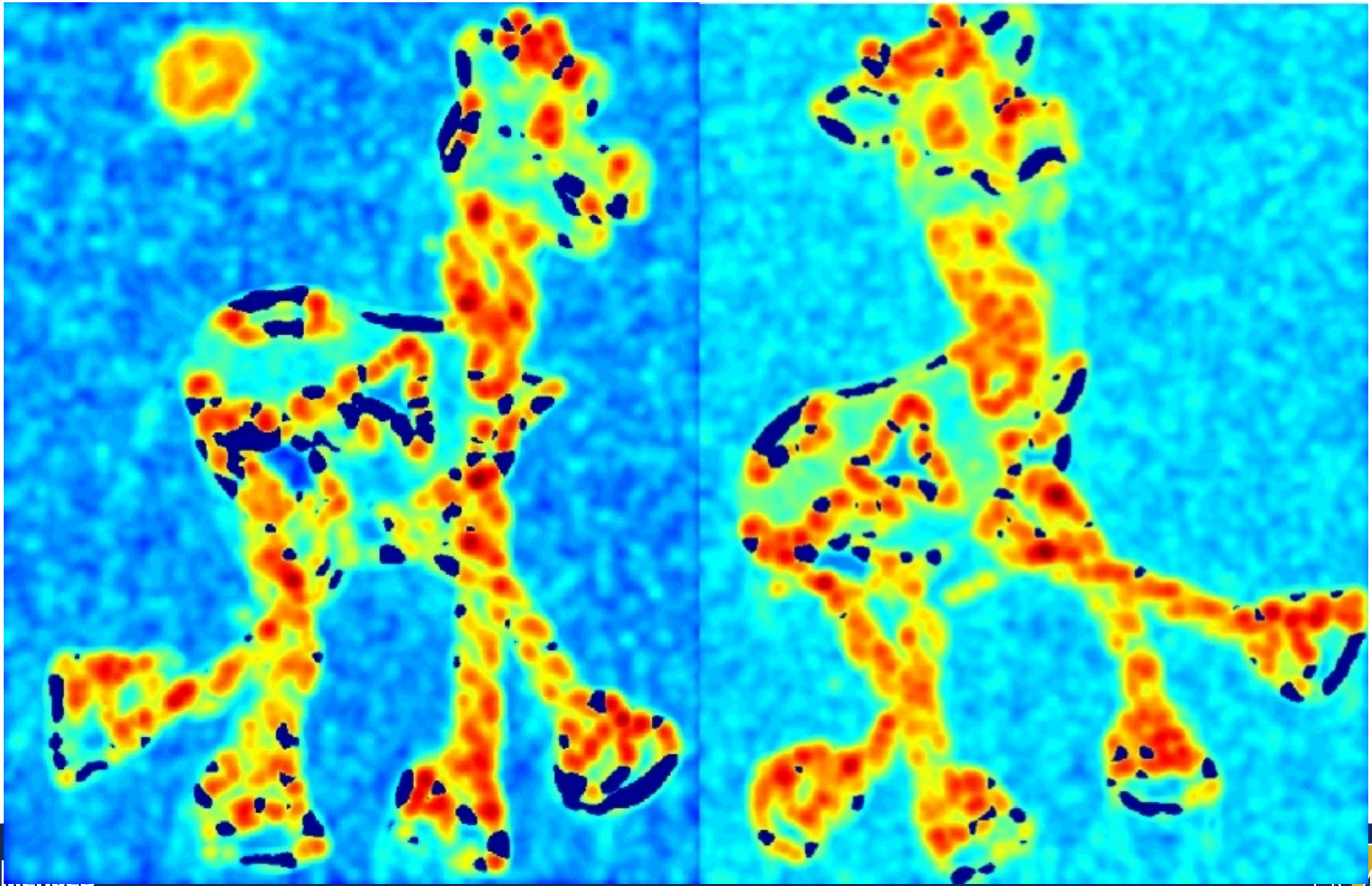
6. Threshold on value of R. Compute nonmax suppression.

Harris detector workflow

Features are often needed to register different views of the same object

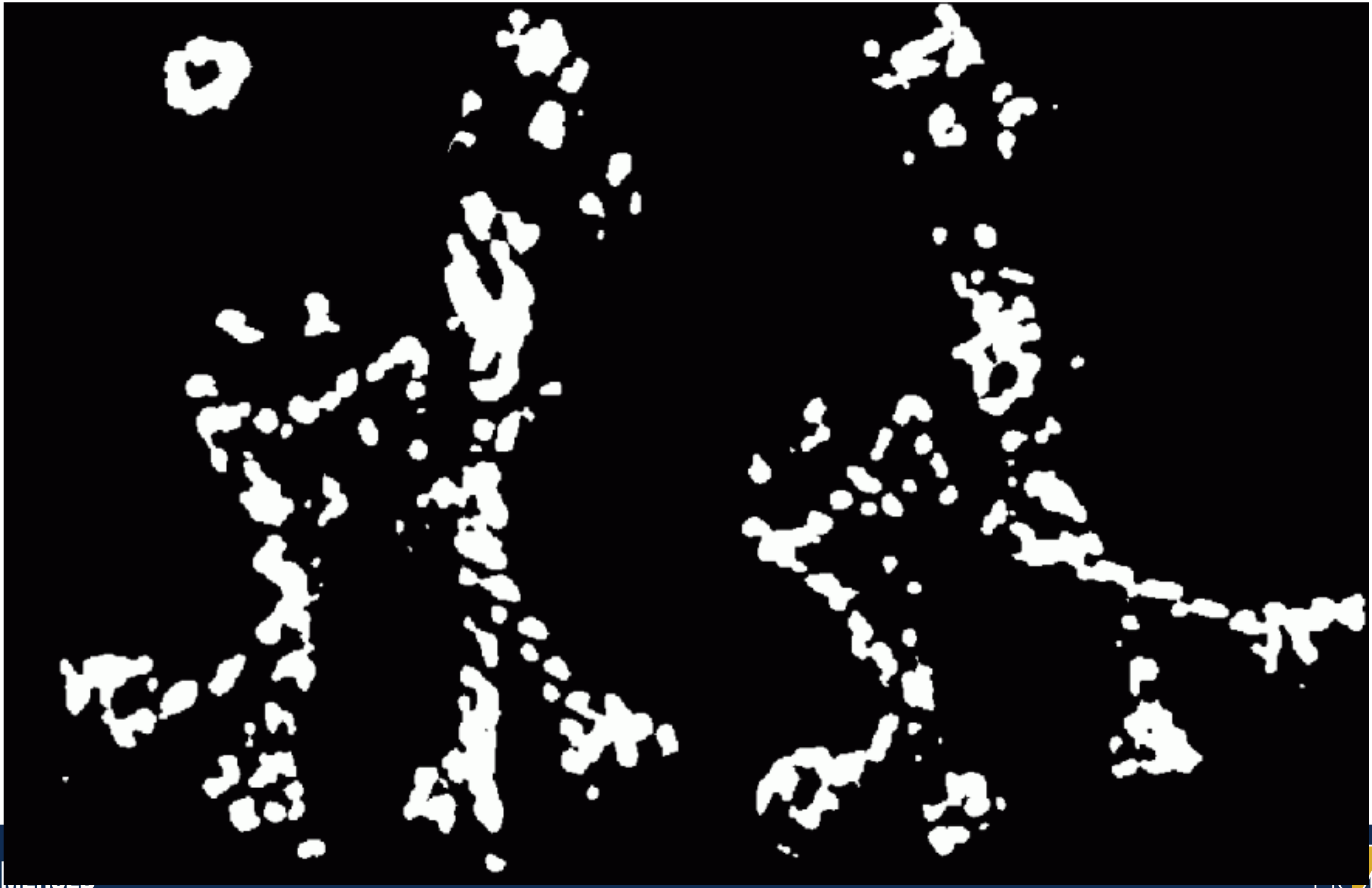
Harris detector workflow

Compute corner response R



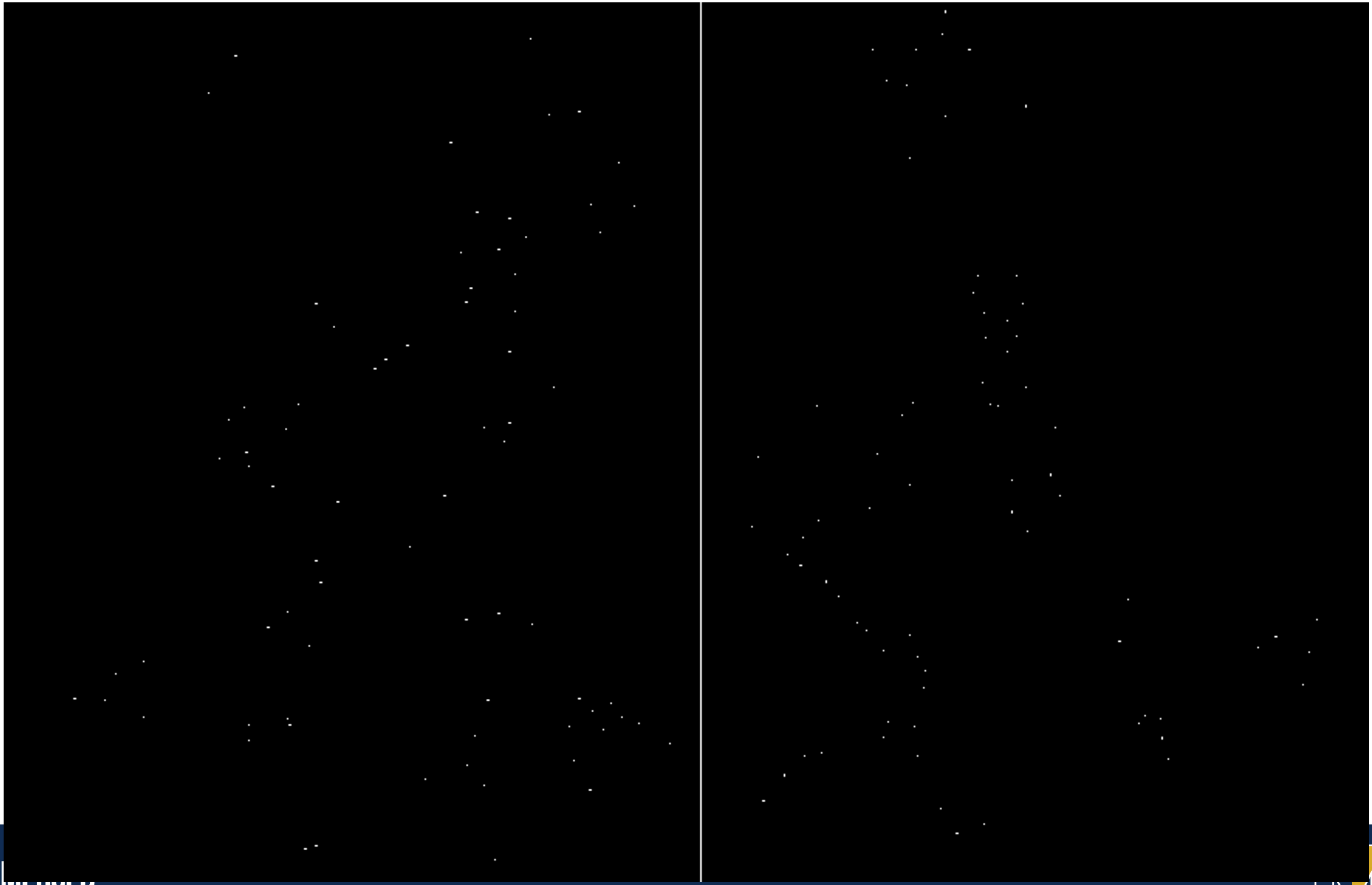
Harris detector workflow

Find points with large corner response: R>threshold



Harris detector workflow

Take only the points of local maxima of R



Harris detector workflow