

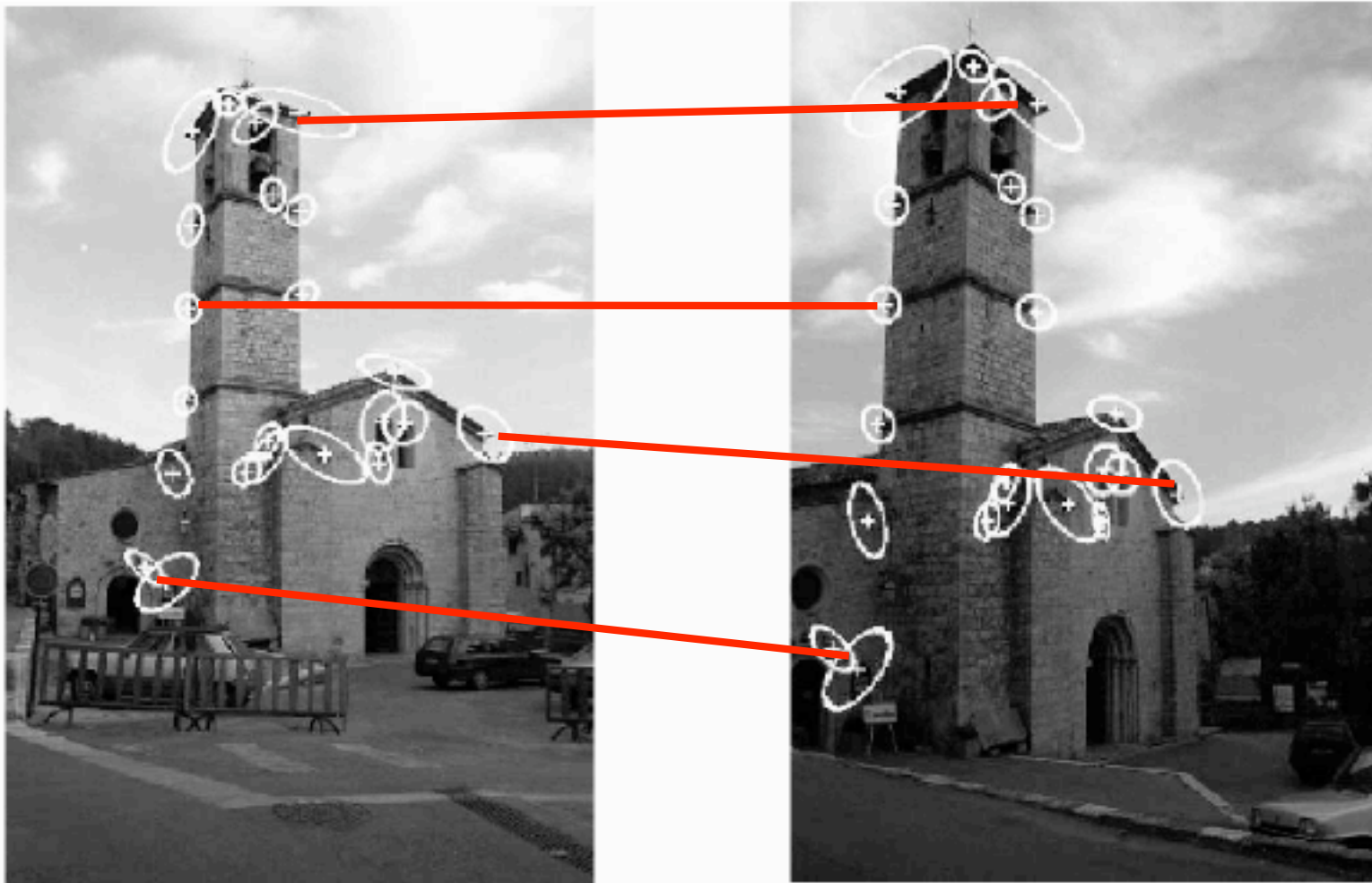
Lecture 06:

Harris Corner Detector

Reading: T&V Section 4.3

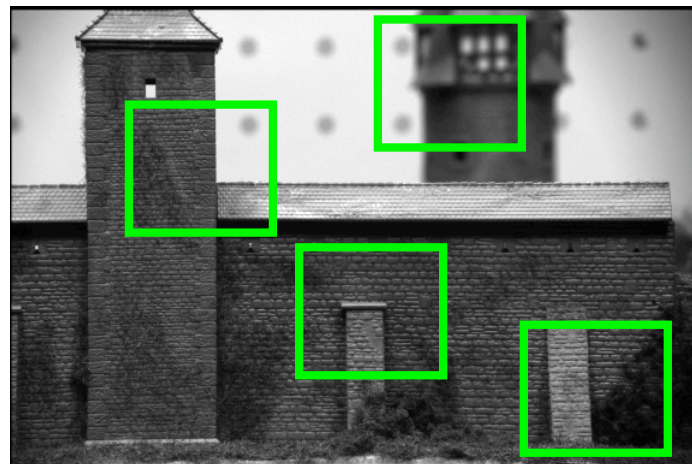
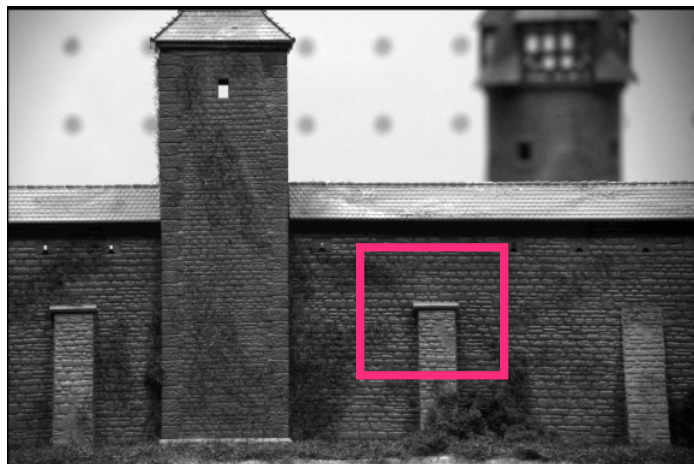
Motivation: Matching Problem

Vision tasks such as stereo and motion estimation require finding corresponding features across two or more views.



Motivation: Patch Matching

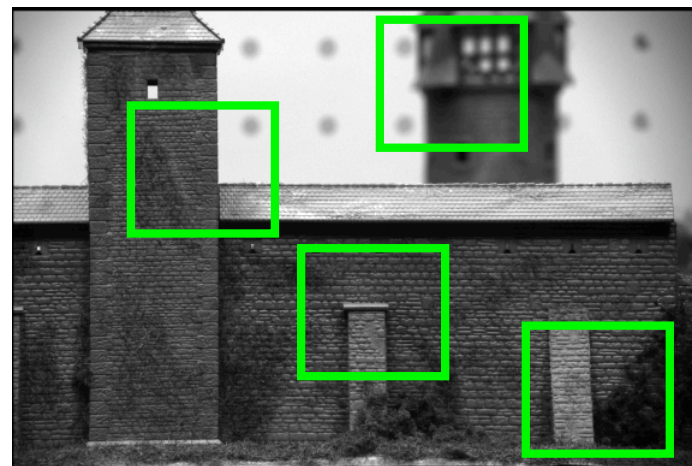
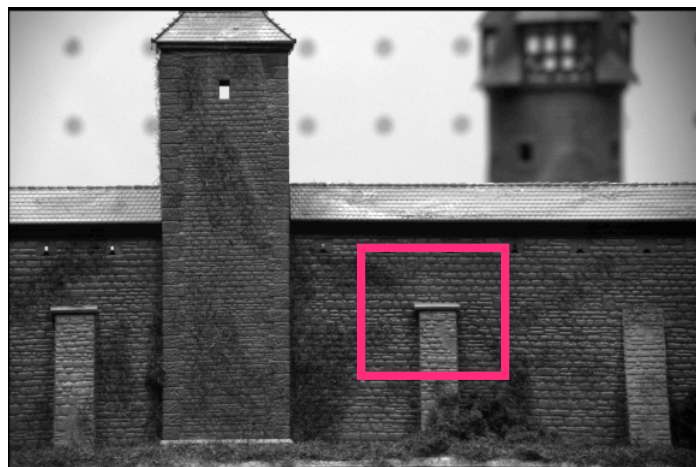
Elements to be matched are image patches of fixed size



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



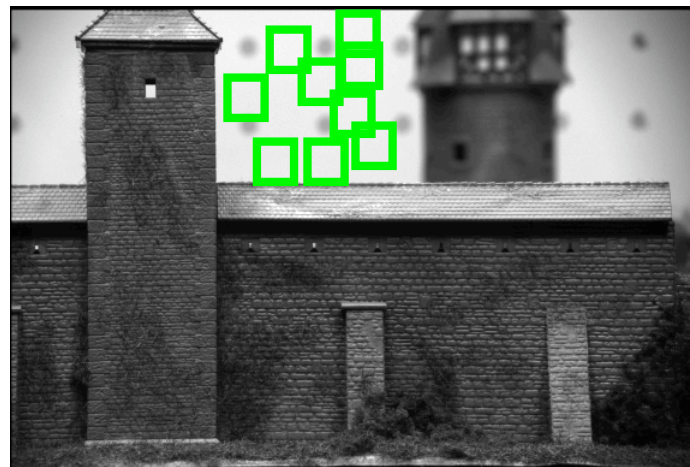
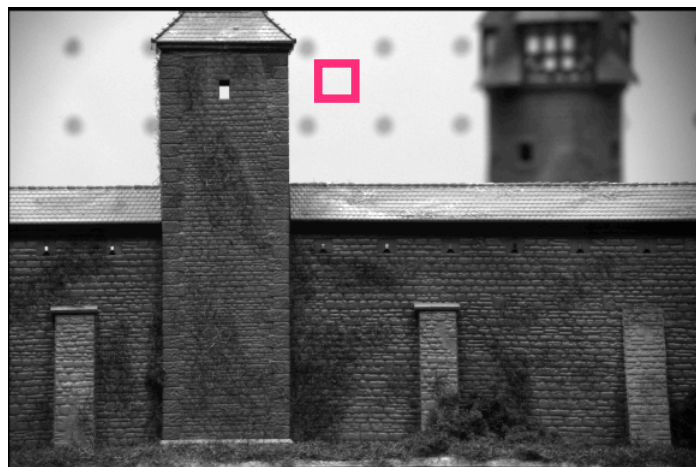
Not all Patches are Created Equal!



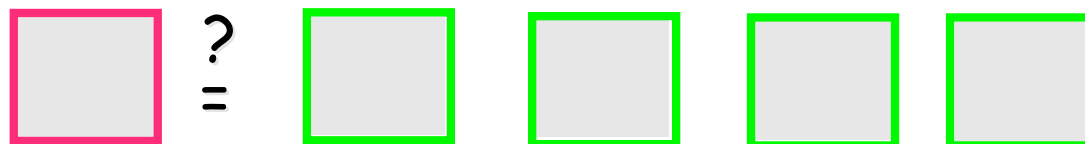
Intuition: this would be a good patch for matching, since it is very distinctive (there is only one patch in the second frame that looks similar).



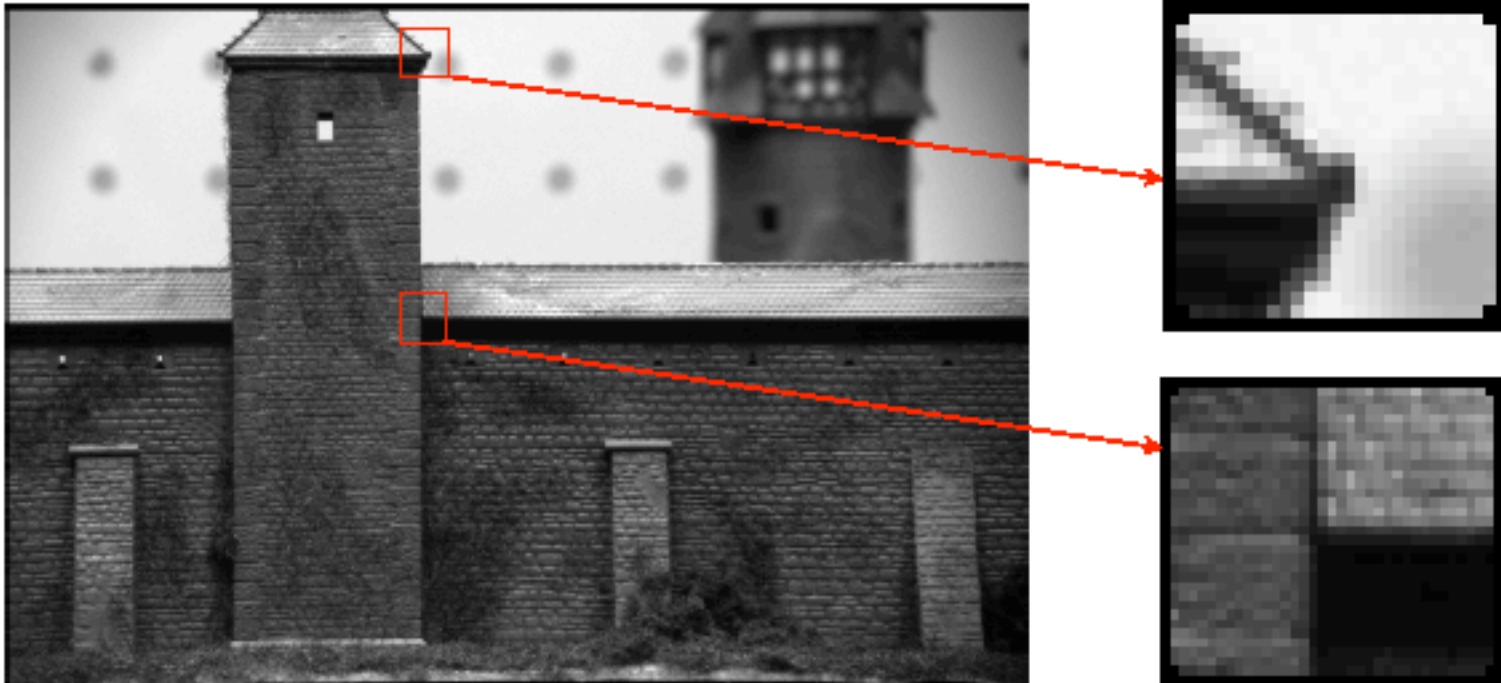
Not all Patches are Created Equal!



Intuition: this would be a BAD patch for matching, since it is not very distinctive (there are many similar patches in the second frame)



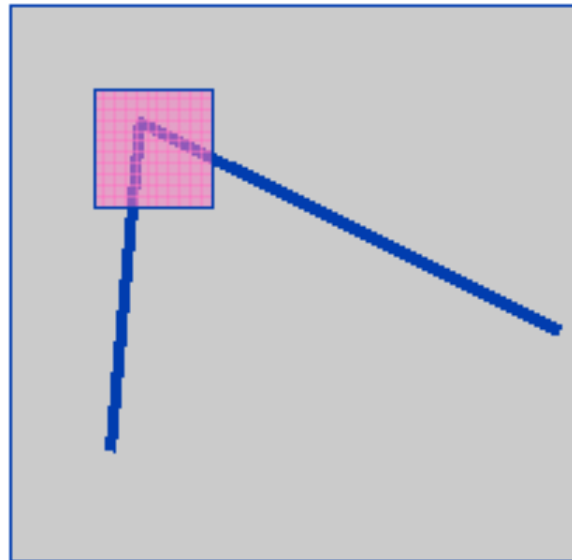
What are Corners?



- Intuitively, junctions of contours.
- Generally more stable features over changes of viewpoint
- Intuitively, large variations in the neighborhood of the point in all directions
- They are good features to match!

Corner Points: Basic Idea

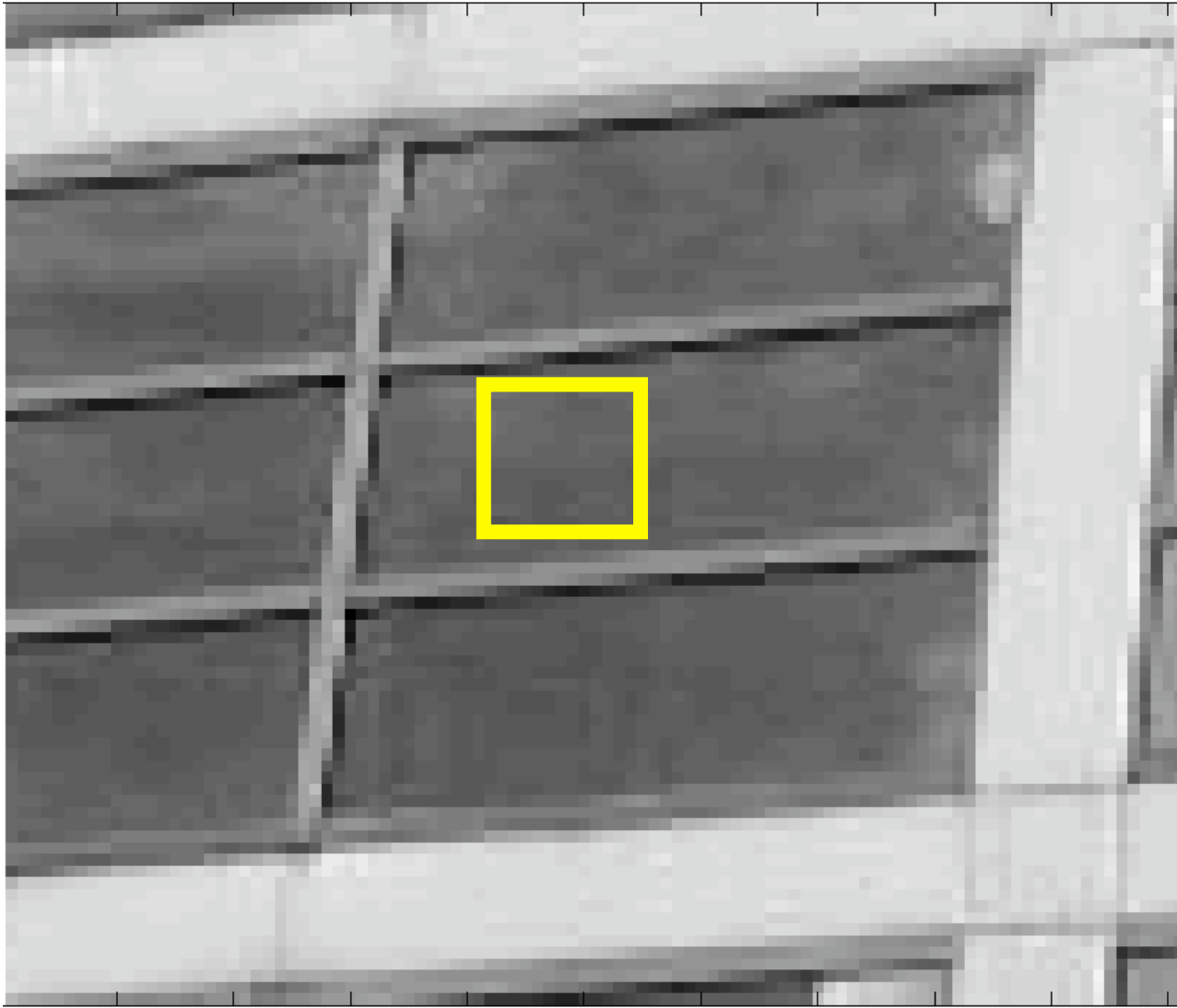
- We should easily recognize the point by looking at intensity values within a small window
- Shifting the window in *any direction* should yield a *large change* in appearance.



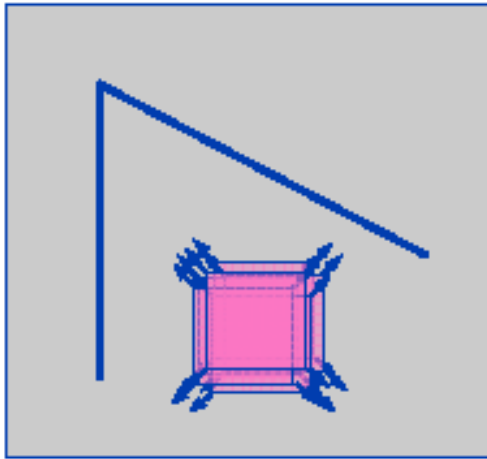
Robert Collins
CSE486, Penn State

Appearance Change in Neighborhood of a Patch

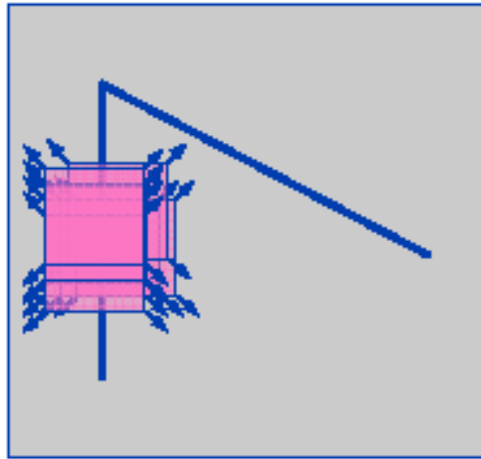
Interactive
“demo”



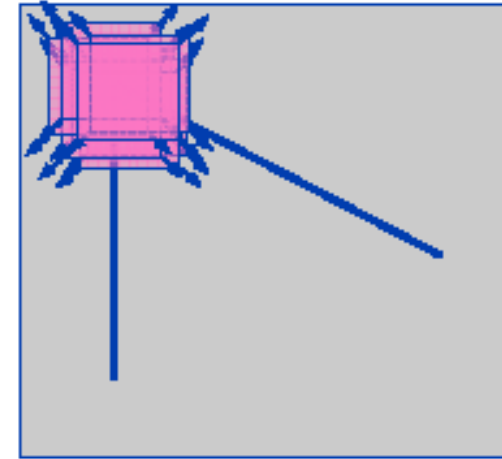
Harris Corner Detector: Basic Idea



“flat” region:
no change in
all directions



“edge”:
no change along
the edge direction



“corner”:
significant change
in all directions

Harris corner detector gives a mathematical approach for determining which case holds.

Harris Detector: Mathematics

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

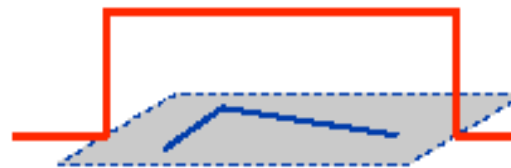
$$E(u, v) = \sum_{x, y} w(x, y) [I(x+u, y+v) - I(x, y)]^2$$

Window
function

Shifted
intensity

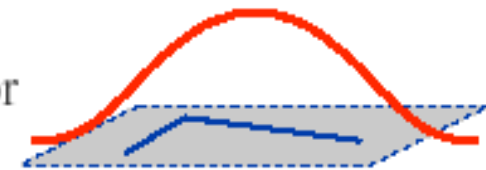
Intensity

Window function $w(x, y) =$



1 in window, 0 outside

or



Gaussian

Harris Detector: Intuition

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

For nearly constant patches, this will be near 0.
For very distinctive patches, this will be larger.
Hence... we want patches where $E(u, v)$ is LARGE.

Taylor Series for 2D Functions

$$f(x+u, y+v) = f(x, y) + uf_x(x, y) + vf_y(x, y) +$$

First partial derivatives

$$\frac{1}{2!} [u^2 f_{xx}(x, y) + uv f_{xy}(x, y) + v^2 f_{yy}(x, y)] +$$

Second partial derivatives

$$\frac{1}{3!} [u^3 f_{xxx}(x, y) + u^2 v f_{xxy}(x, y) + uv^2 f_{xyy}(x, y) + v^3 f_{yyy}(x, y)]$$

Third partial derivatives

+ ... (Higher order terms)

First order approx

$$f(x+u, y+v) \approx f(x, y) + uf_x(x, y) + vf_y(x, y)$$

Harris Corner Derivation

$$\sum [I(x+u, y+v) - I(x, y)]^2$$

$$\approx \sum [I(x, y) + uI_x + vI_y - I(x, y)]^2 \quad \text{First order approx}$$

$$= \sum u^2 I_x^2 + 2uv I_x I_y + v^2 I_y^2$$

$$= \sum \begin{bmatrix} u & v \end{bmatrix} \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \quad \text{Rewrite as matrix equation}$$

$$= \begin{bmatrix} u & v \end{bmatrix} \left(\sum \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \right) \begin{bmatrix} u \\ v \end{bmatrix}$$

Harris Detector: Mathematics

For small shifts $[u, v]$ we have a *bilinear* approximation:

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

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

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

Windowing function - computing a weighted sum (simplest case, $w=1$)

Note: these are just products of components of the gradient, I_x, I_y

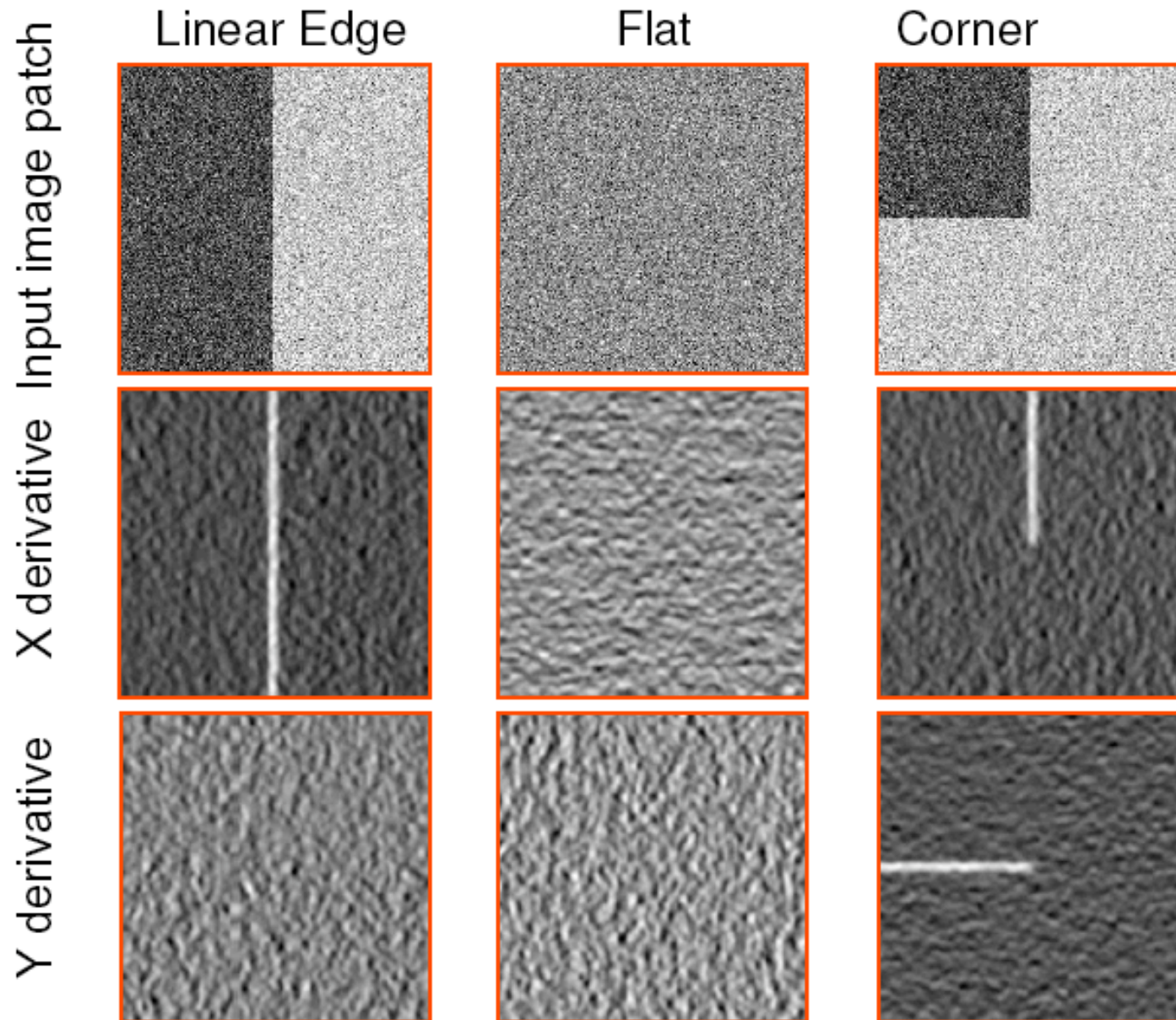
Intuitive Way to Understand Harris

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

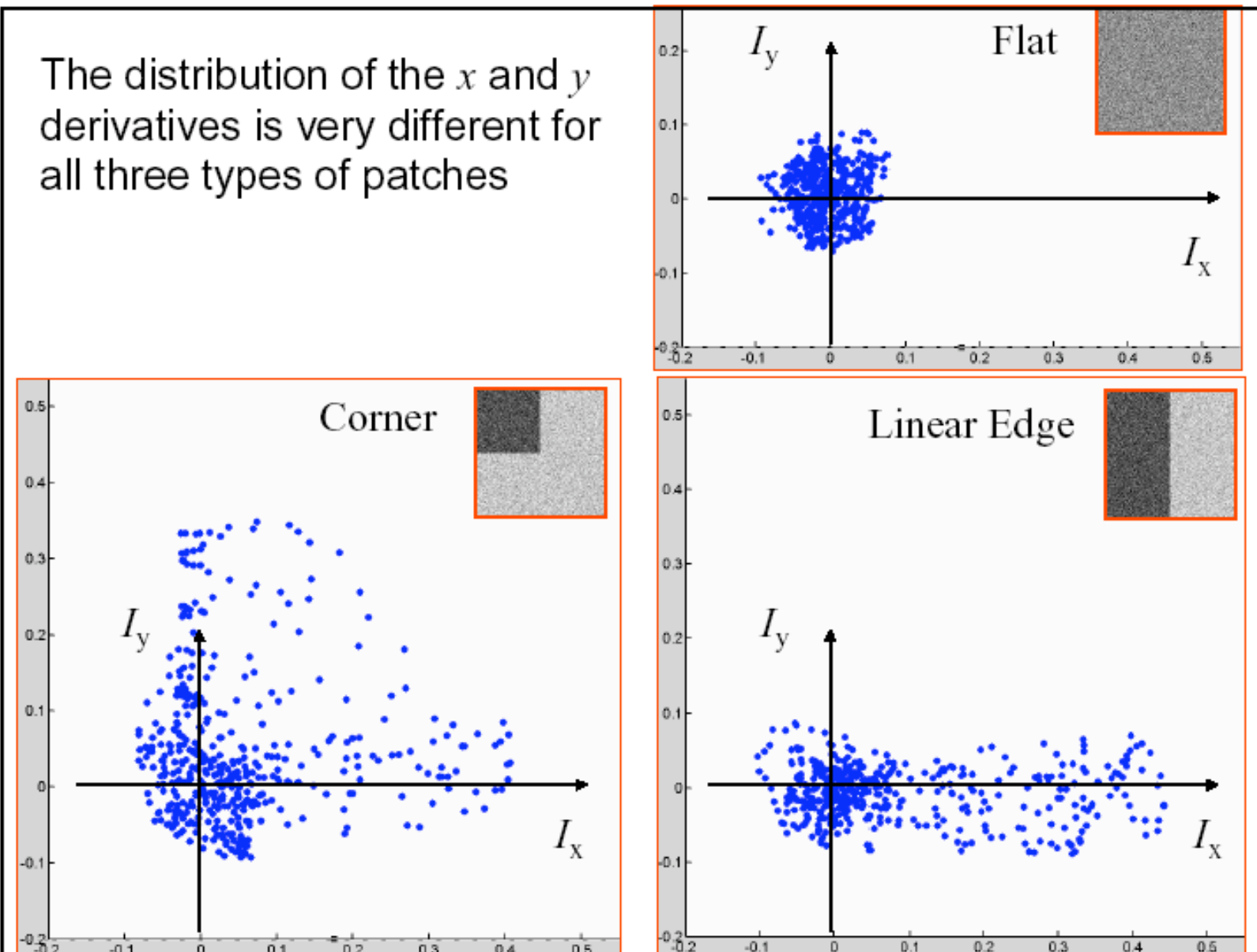
Analyze ellipse parameters for varying cases...

Example: Cases and 2D Derivatives



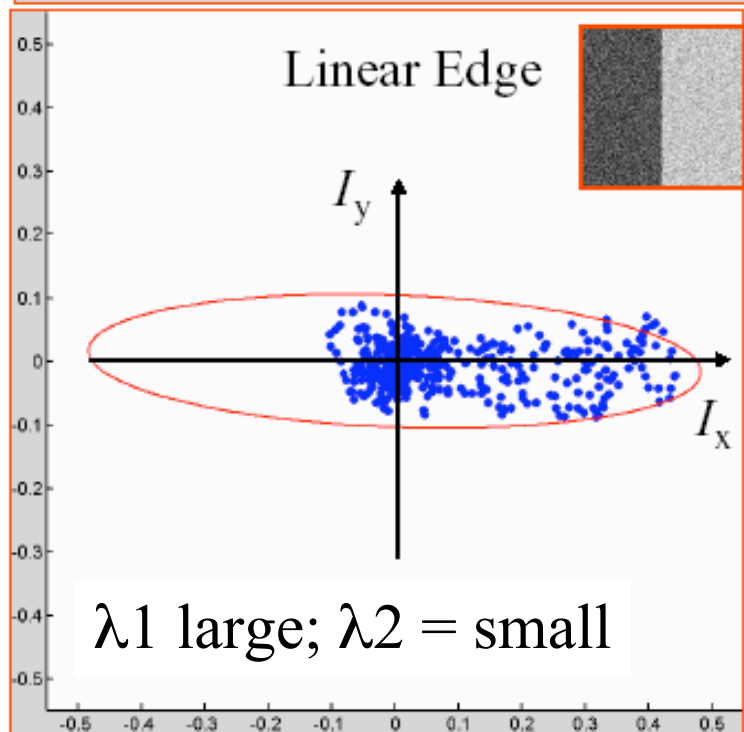
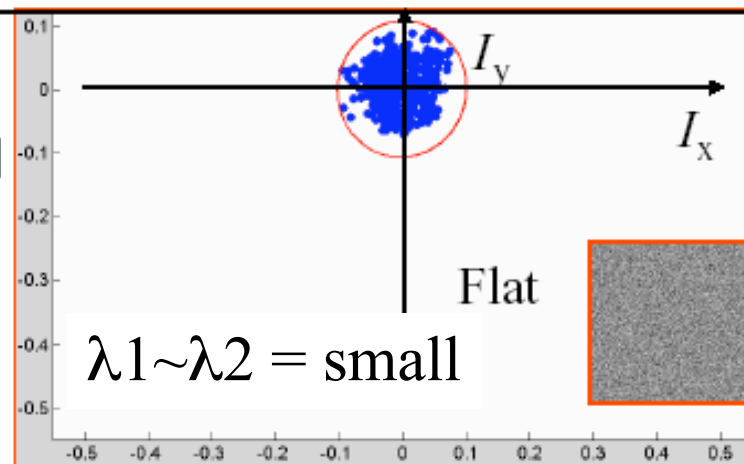
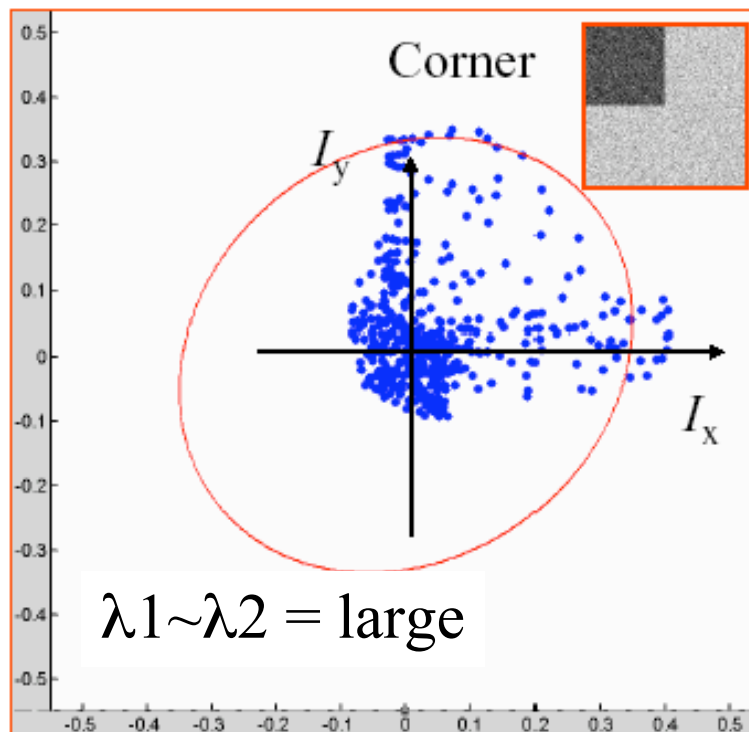
Plotting Derivatives as 2D Points

The distribution of the x and y derivatives is very different for all three types of patches



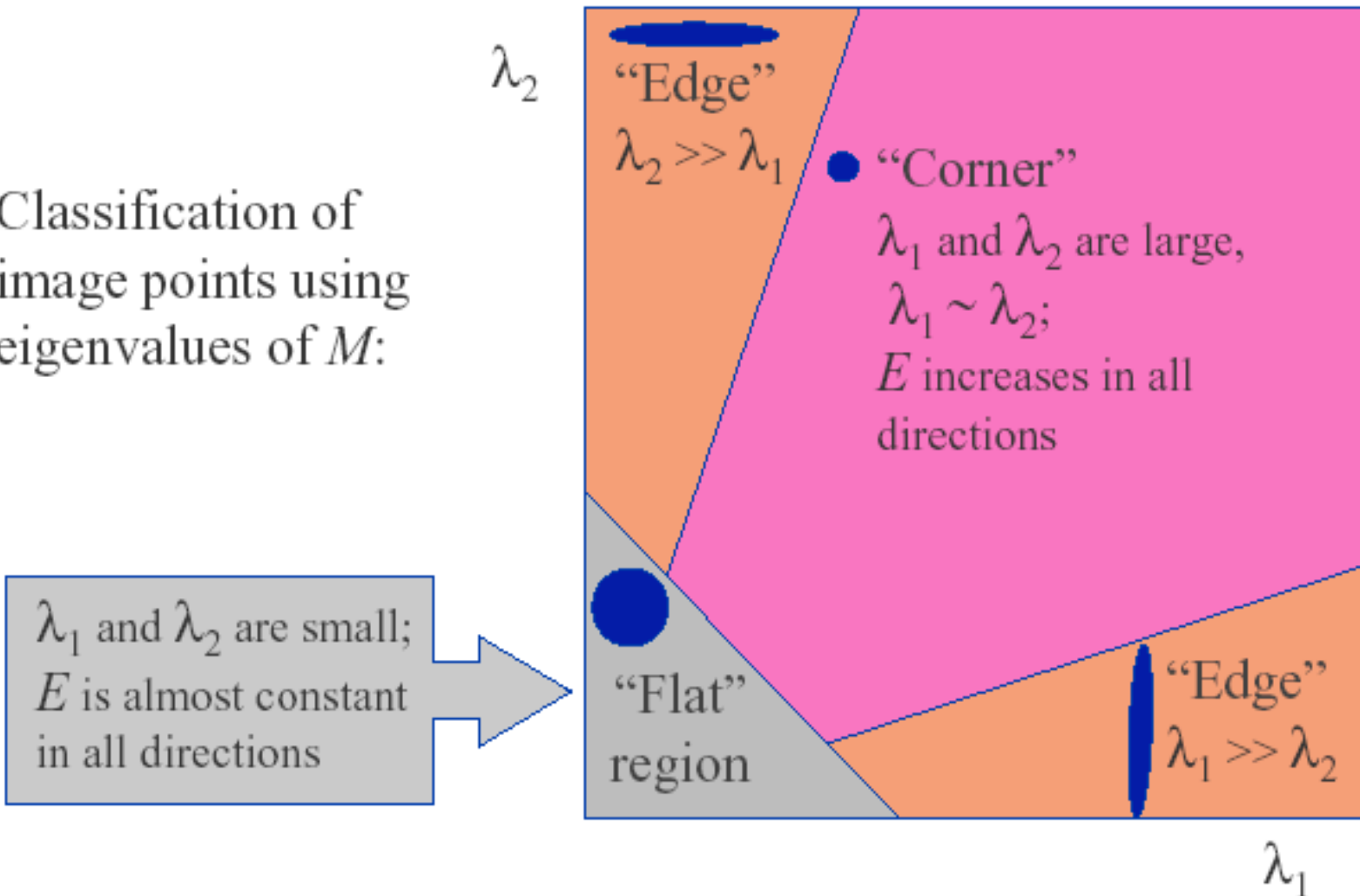
Fitting Ellipse to each Set of Points

The distribution of x and y derivatives can be characterized by the shape and size of the principal component ellipse



Classification via Eigenvalues

Classification of
image points using
eigenvalues of M :



Corner Response Measure

Measure of corner response:

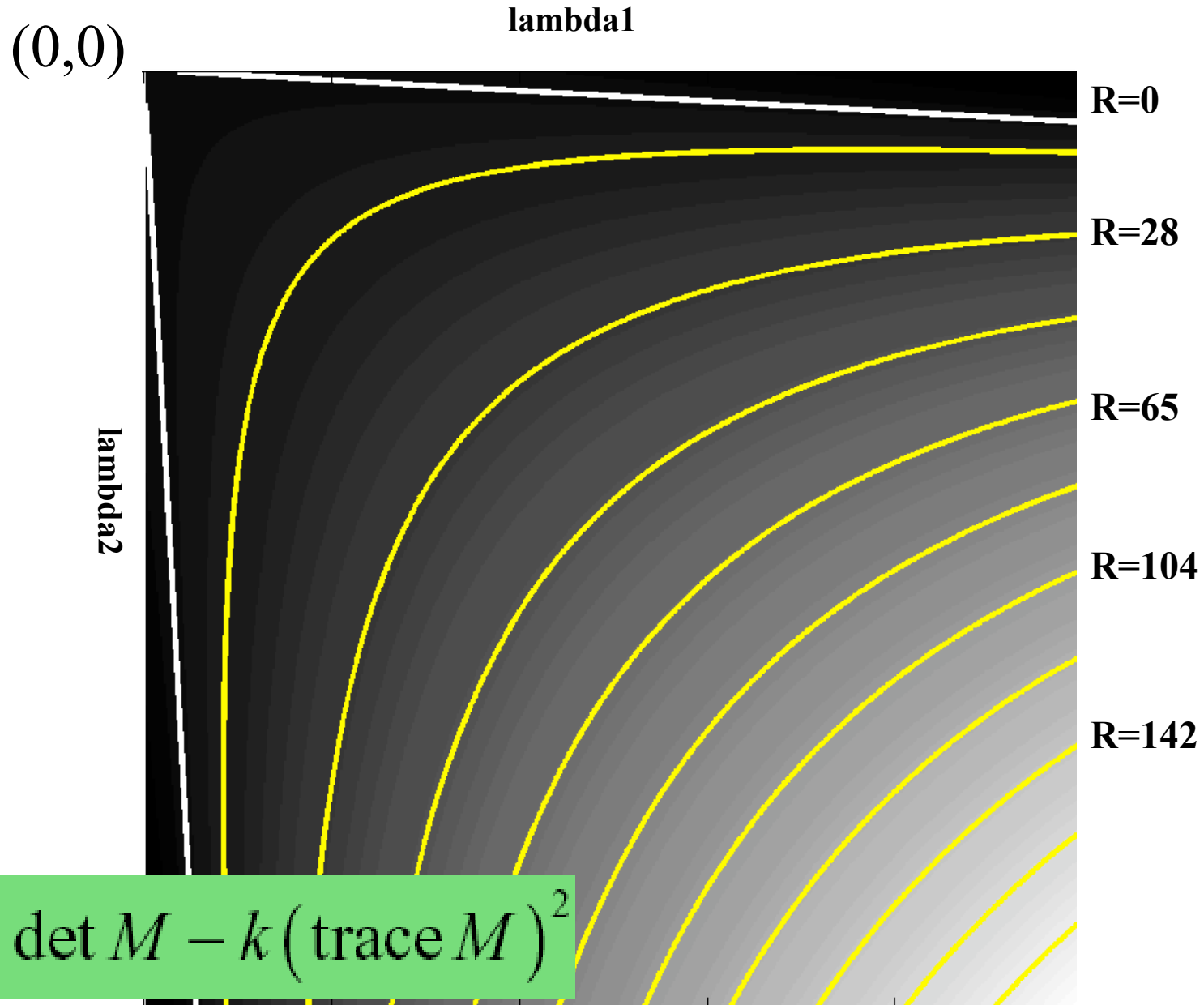
$$R = \det M - k (\text{trace } M)^2$$

$$\det M = \lambda_1 \lambda_2$$

$$\text{trace } M = \lambda_1 + \lambda_2$$

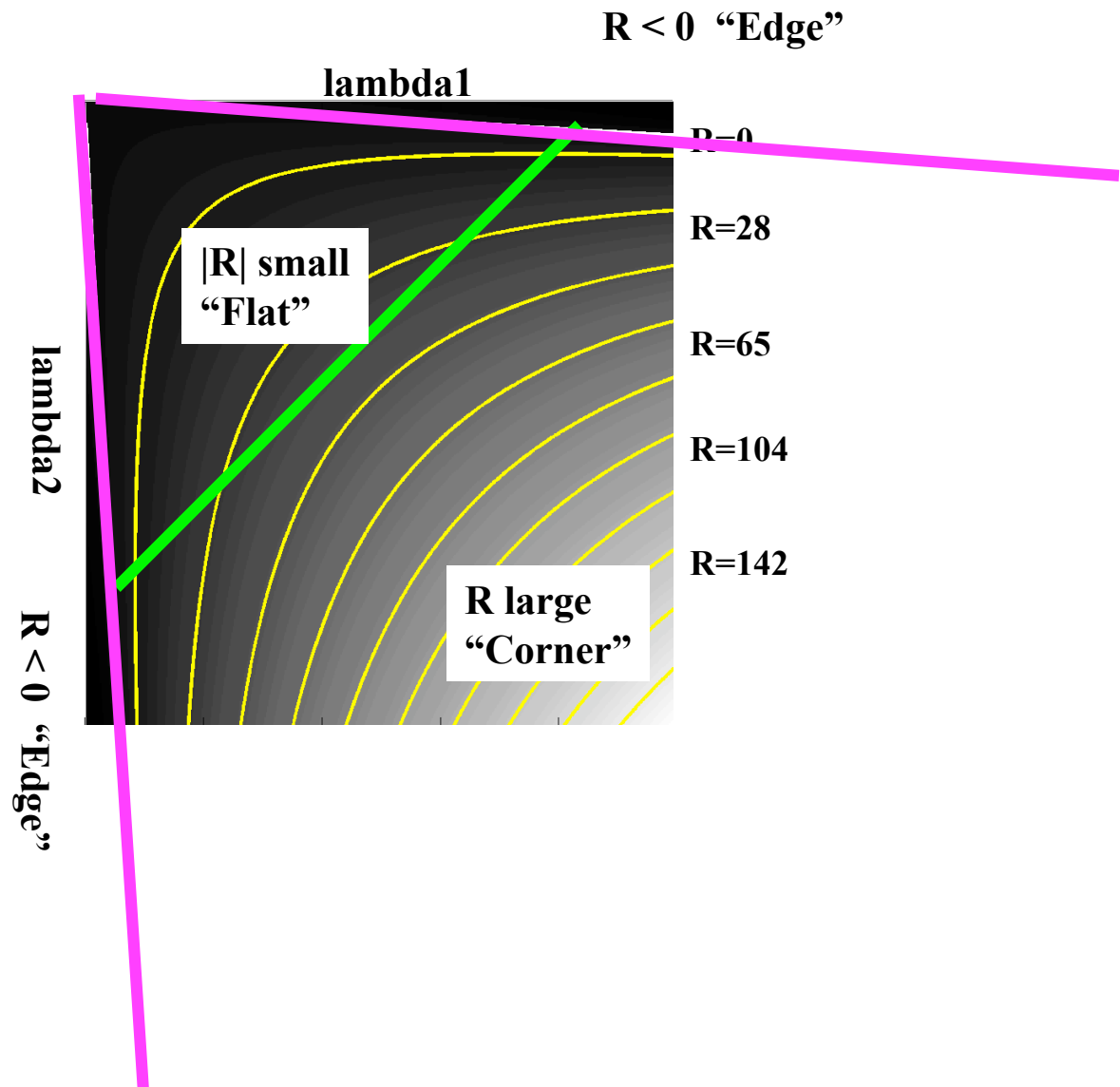
(k is an empirically determined constant; $k = 0.04 - 0.06$)

Corner Response Map

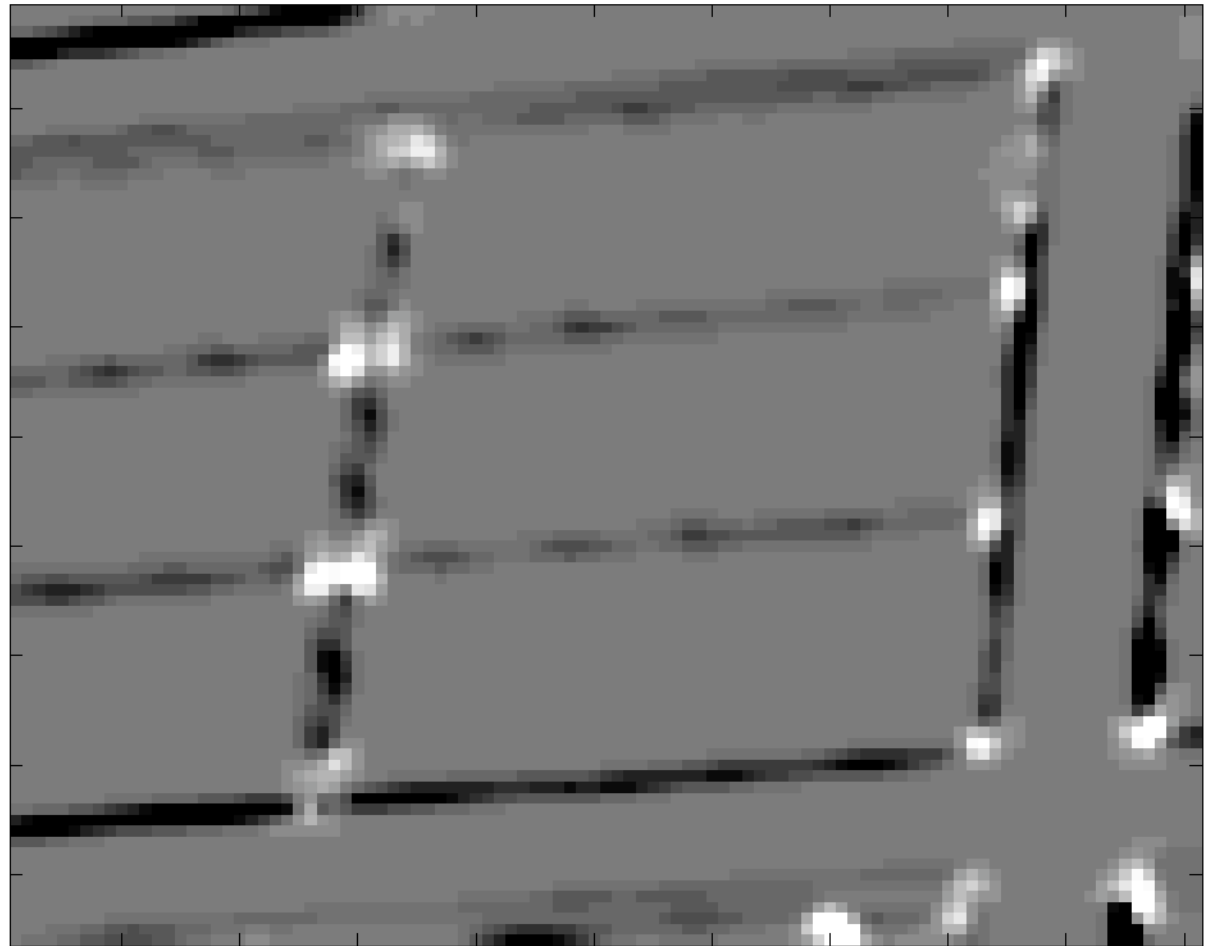


Corner Response Map

- 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



Corner Response Example

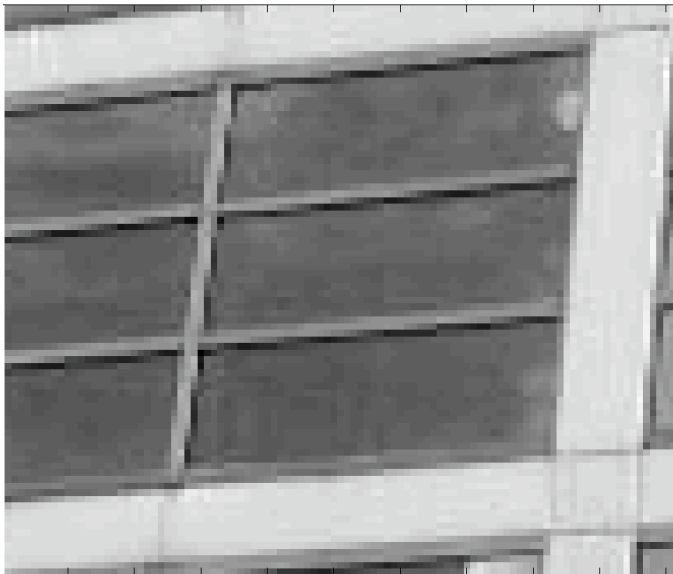


Harris R score.

I_x , I_y computed using Sobel operator

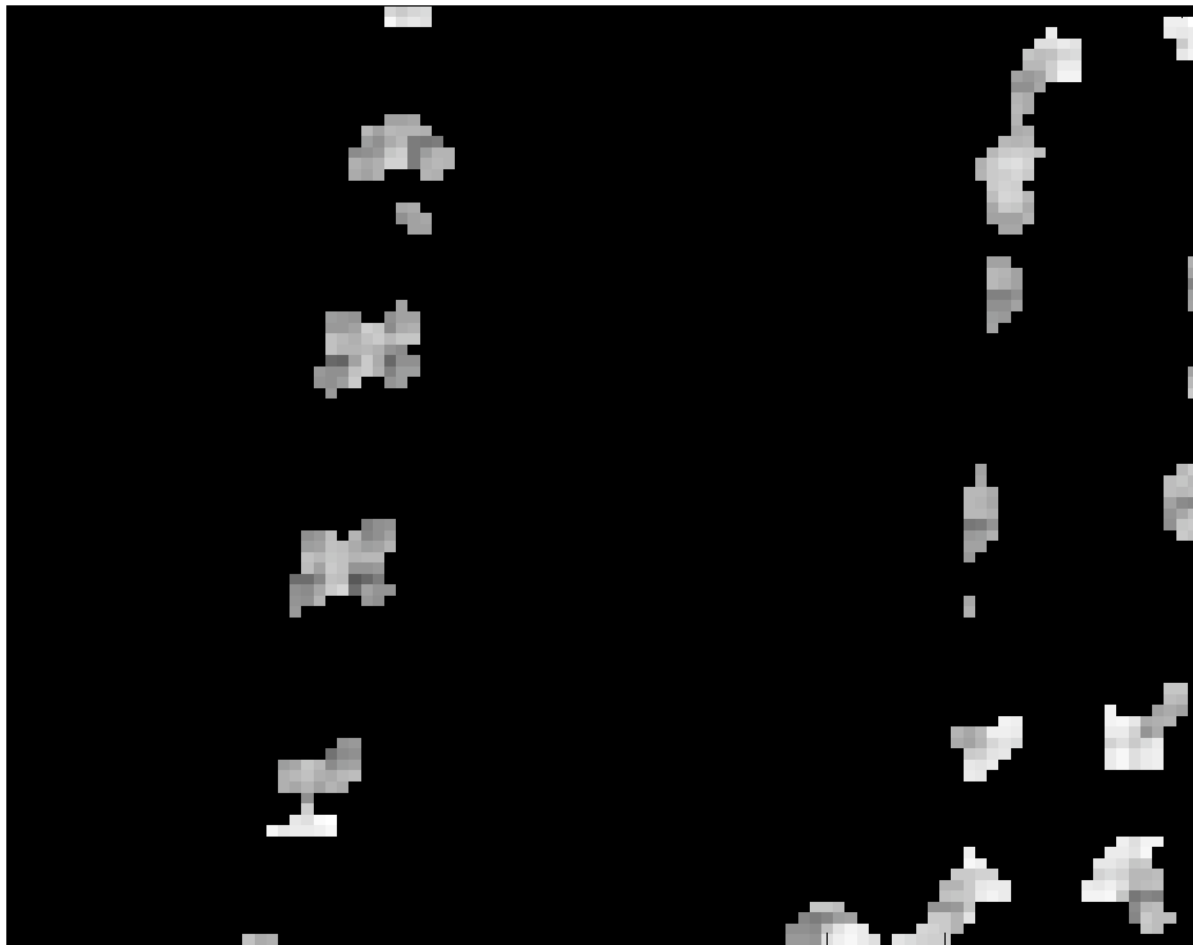
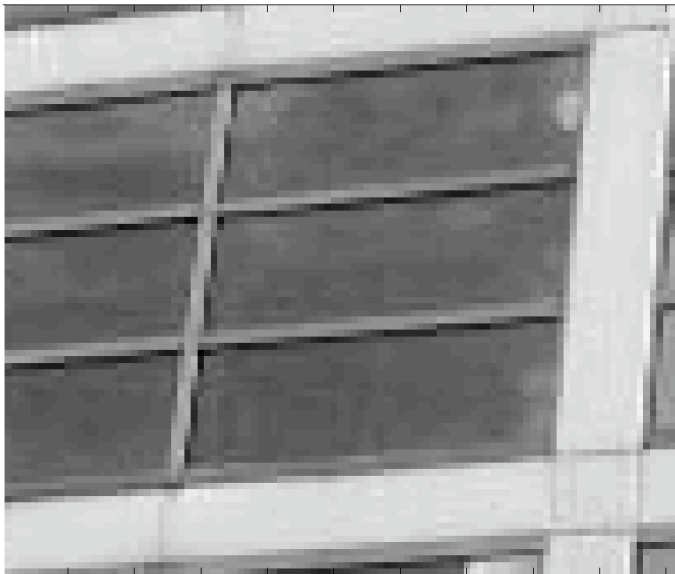
Windowing function w = Gaussian, $\sigma=1$

Corner Response Example



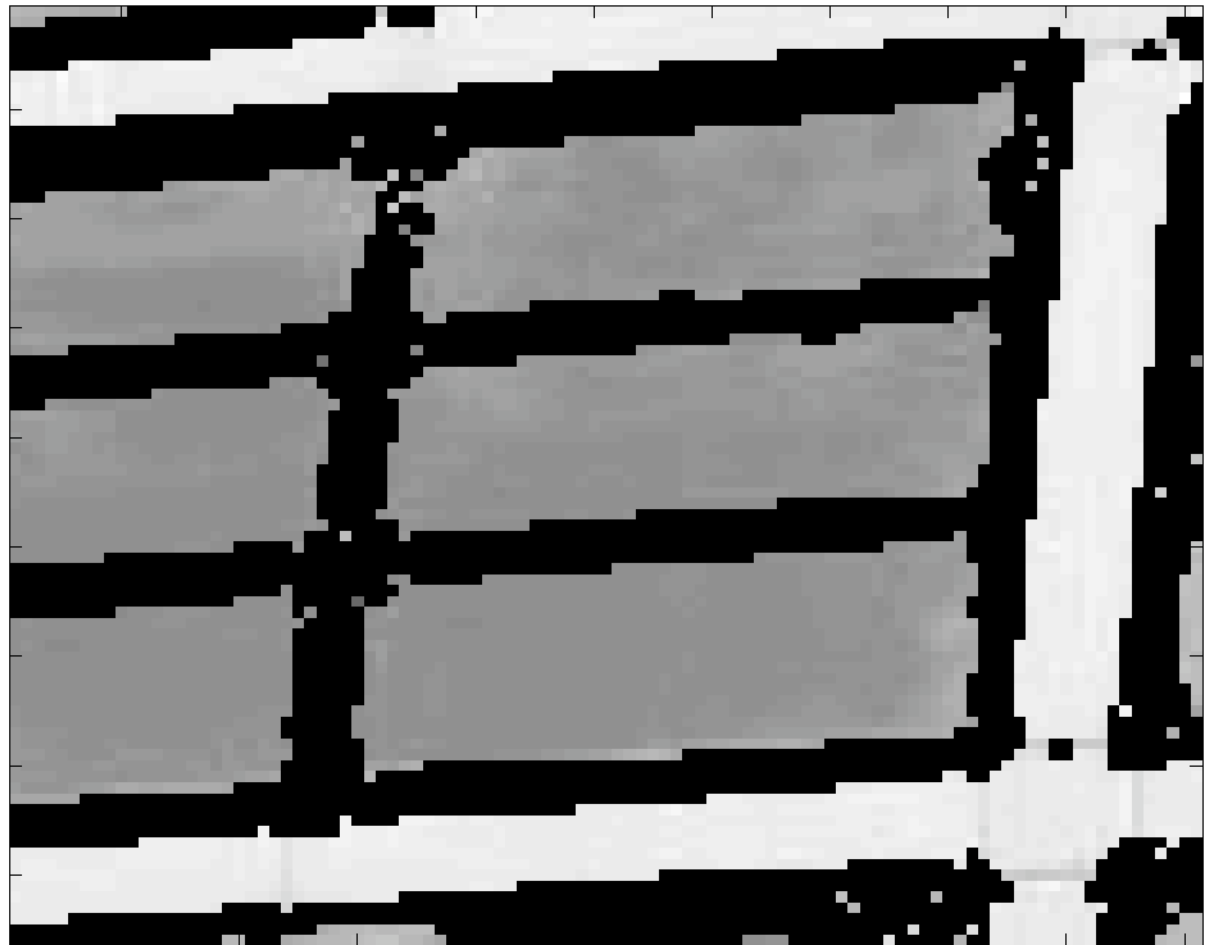
Threshold: $R < -10000$
(edges)

Corner Response Example



Threshold: > 10000
(corners)

Corner Response Example



Threshold: $-10000 < R < 10000$
(neither edges nor corners)

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

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

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

$$S_{x2} = G_{\sigma^2} * I_{x2} \quad S_{y2} = G_{\sigma^2} * I_{y2} \quad S_{xy} = G_{\sigma^2} * I_{xy}$$

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

$$R = \text{Det}(H) - k(\text{Trace}(H))^2$$

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