CS 773
Features

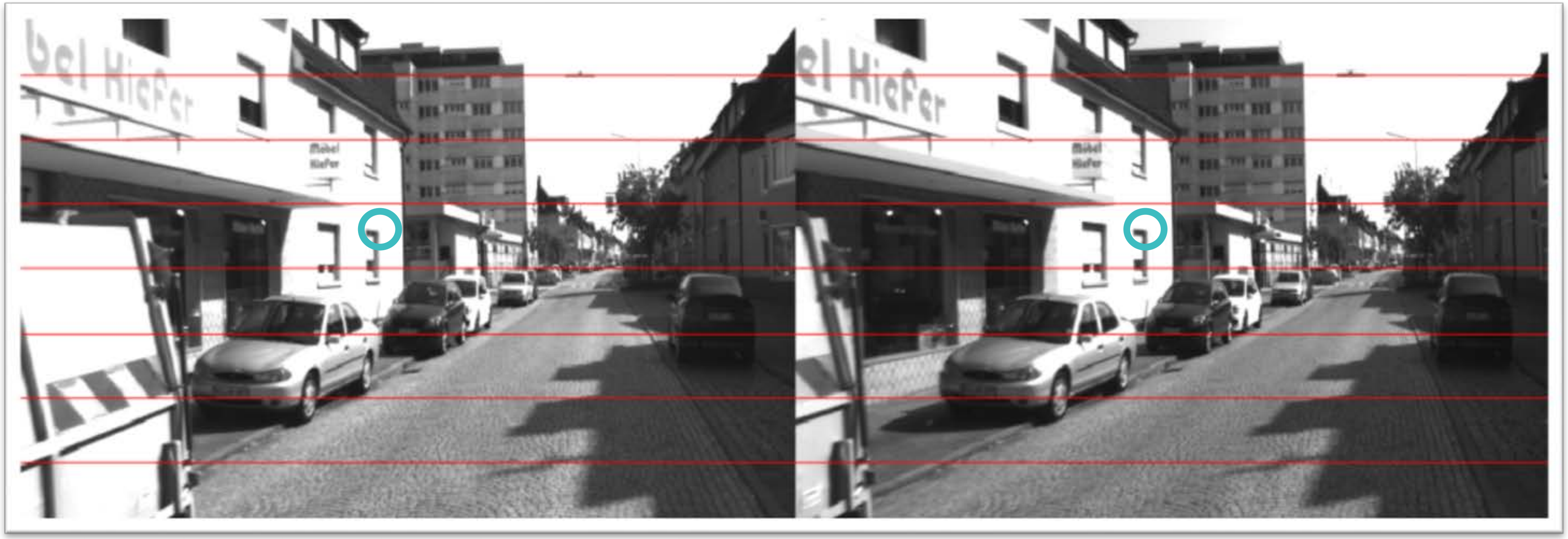
Chia-Yen Chen

Dept. of Computer Science
The University of Auckland
yen@cs.auckland.ac.nz

Features in images

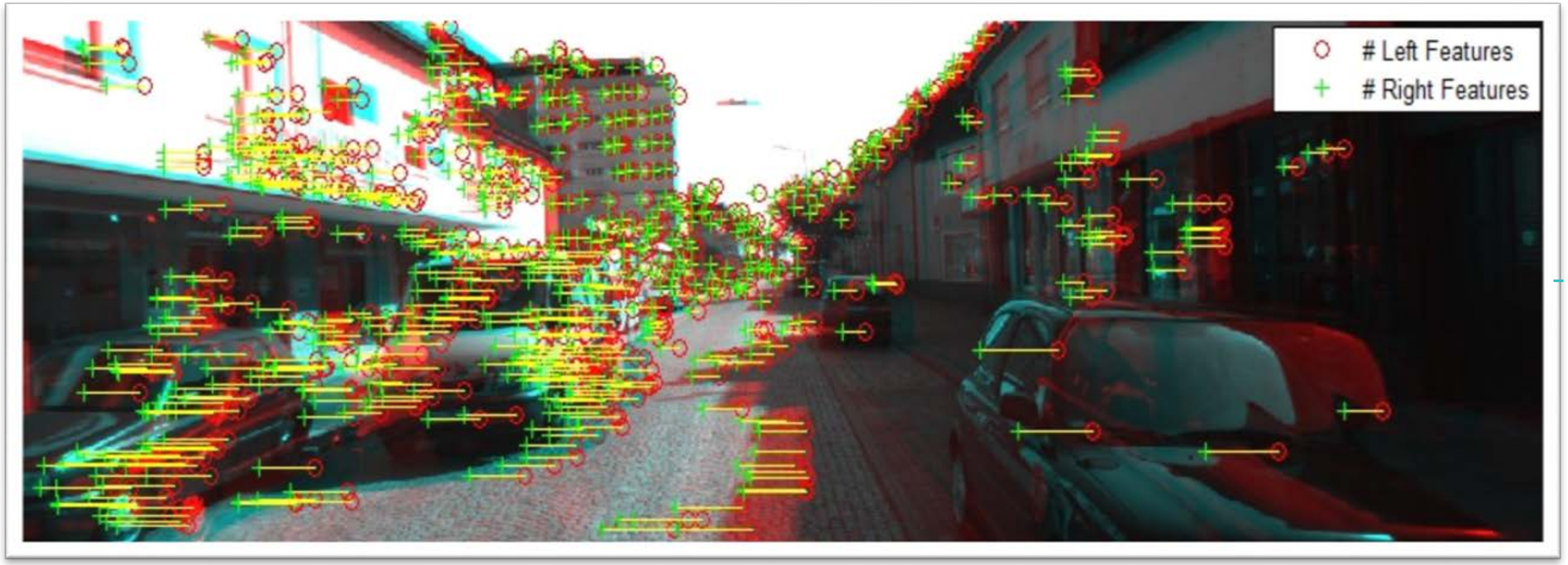
Something interesting in the image

- Spatial structures in the image
  - Point
  - Edge
  - Corner
  - Shape



- Characteristics revealed as the result of some operations
  - Feature detection

Examples of matched feature points

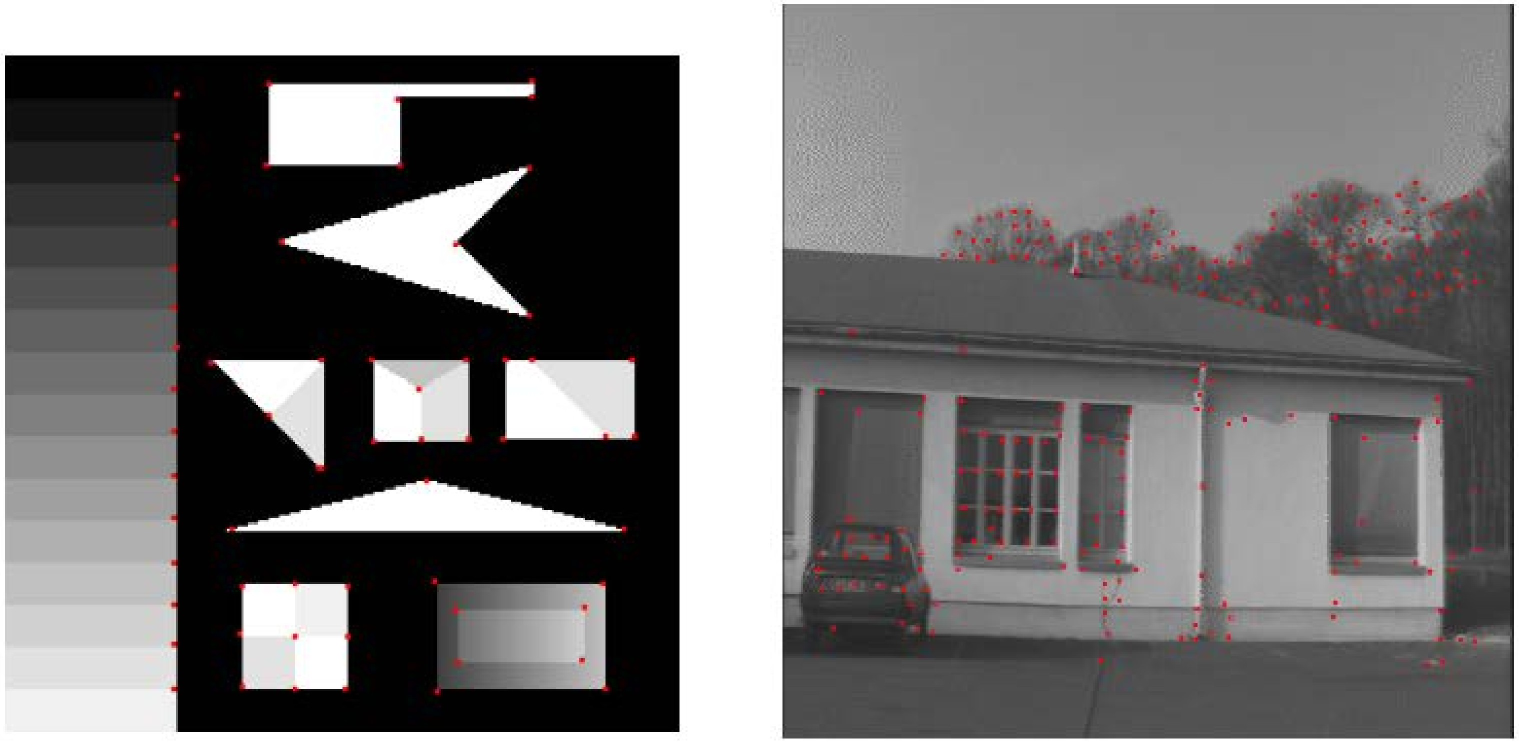


Left/right images

Successive frames

Corners

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



The two figures show an artificial and a real image, respectively, with the corners indicated in red.

Feature Point detection

The corner detectors should satisfy the following criteria:

- All (or most) the true feature points should be detected.
- No false feature points should be detected.
- Feature points should be well localized.
- Feature point detector should be robust with respect to noise.
- Feature point detector should be efficient.

Feature Point detection

Two families of corner detectors

1. Algorithms that work directly with the values of brightness of images (without segmenting the image in advance
   - Usually based on the study of derivatives (orientation, magnitude) of grey level or colour image

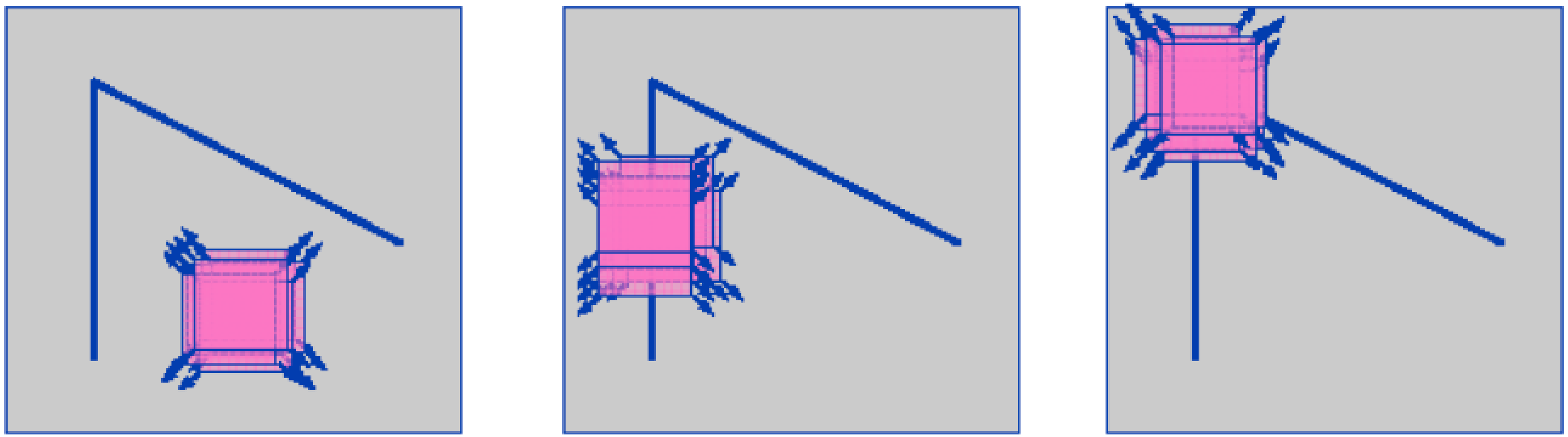
2. Algorithms that extract object boundaries first and analyze its shape afterwards
   - Boundaries often assumed to be extracted by edge-detectors
   - Usually based on the analysis of the curvature of boundaries

Group 2 seems to offer less reliability (use edge detectors for boundary extraction is not working well) and slower solutions.

The Harris-Plessey detector

Basic idea: Look at changes of Intensity in any direction on a small windows over any given point.

- Flat region: no change in all directions
- Edge: no change along the edge direction
- Corner: significant change in most directions



Slides adapted: https://www.coursehero.com/sitemap/schools/21-Penn-State/courses/645562-CSE486/

The Harris-Plessey detector

Change of intensity

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

By Taylor series expansion, first order approximation for 2D function

- $I(x+u, y+v) \approx I(x,y) + uI_x(x,y) + vI_y(x,y)$

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

$$\approx \sum_w \left[\cancel{I(x,y)} + uI_x(x,y) + vI_y(x,y) - \cancel{I(x,y)}\right]^2$$

$$= \sum_w \left[u^2 I_x(x,y)^2 + 2uvI_x(x,y)I_y(x,y) + v^2 I_y(x,y)^2\right]$$

The Harris-Plessey detector

- The following 2x2 symmetric matrix is considered at each image point (pixel) of the image

$$M = \begin{bmatrix} \sum_{w} \left(\frac{\partial I}{\partial x}(x,y) \right)^2 & \sum_{w} \left(\frac{\partial I}{\partial x}(x,y) \frac{\partial I}{\partial y}(x,y) \right) \\ \sum_{w} \left(\frac{\partial I}{\partial y}(x,y) \frac{\partial I}{\partial x}(x,y) \right) & \sum_{w} \left(\frac{\partial I}{\partial y}(x,y) \right)^2 \end{bmatrix}$$

I(x,y): image intensity at point (x,y)

The Harris-Plessey detector

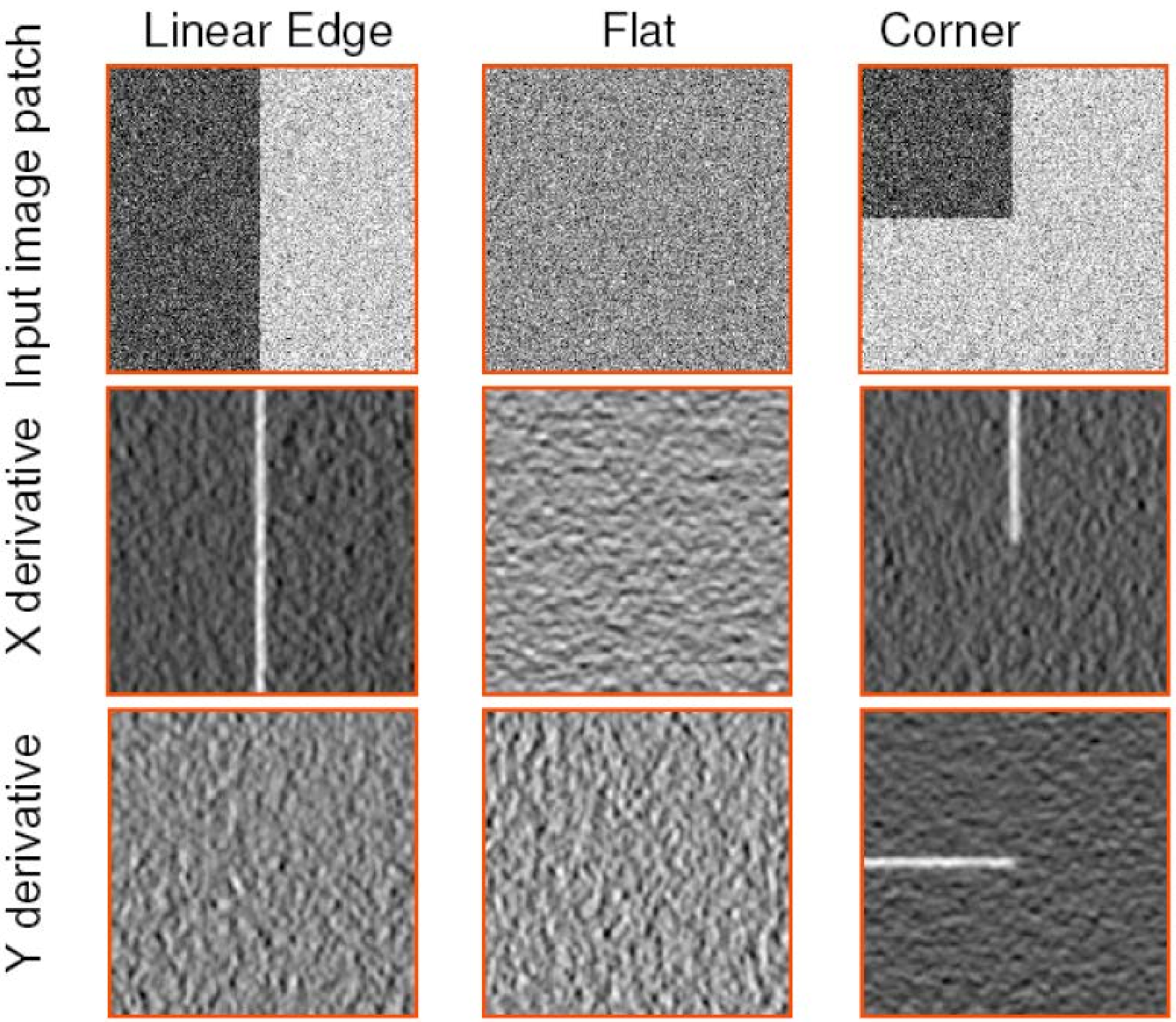
- By evaluating the eigenvalues of the matrix M, we can detect the image feature by following rules:

1. If both eigenvalues are small, the intensity of the windowed image region is approximately constant (homogeneous region).
2. If one eigenvalues is high and the other is low, this indicates an edge.
3. If both eigenvalues are sufficiently large, the point is declared to be a corner.

The Harris-Plessey detector

- 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

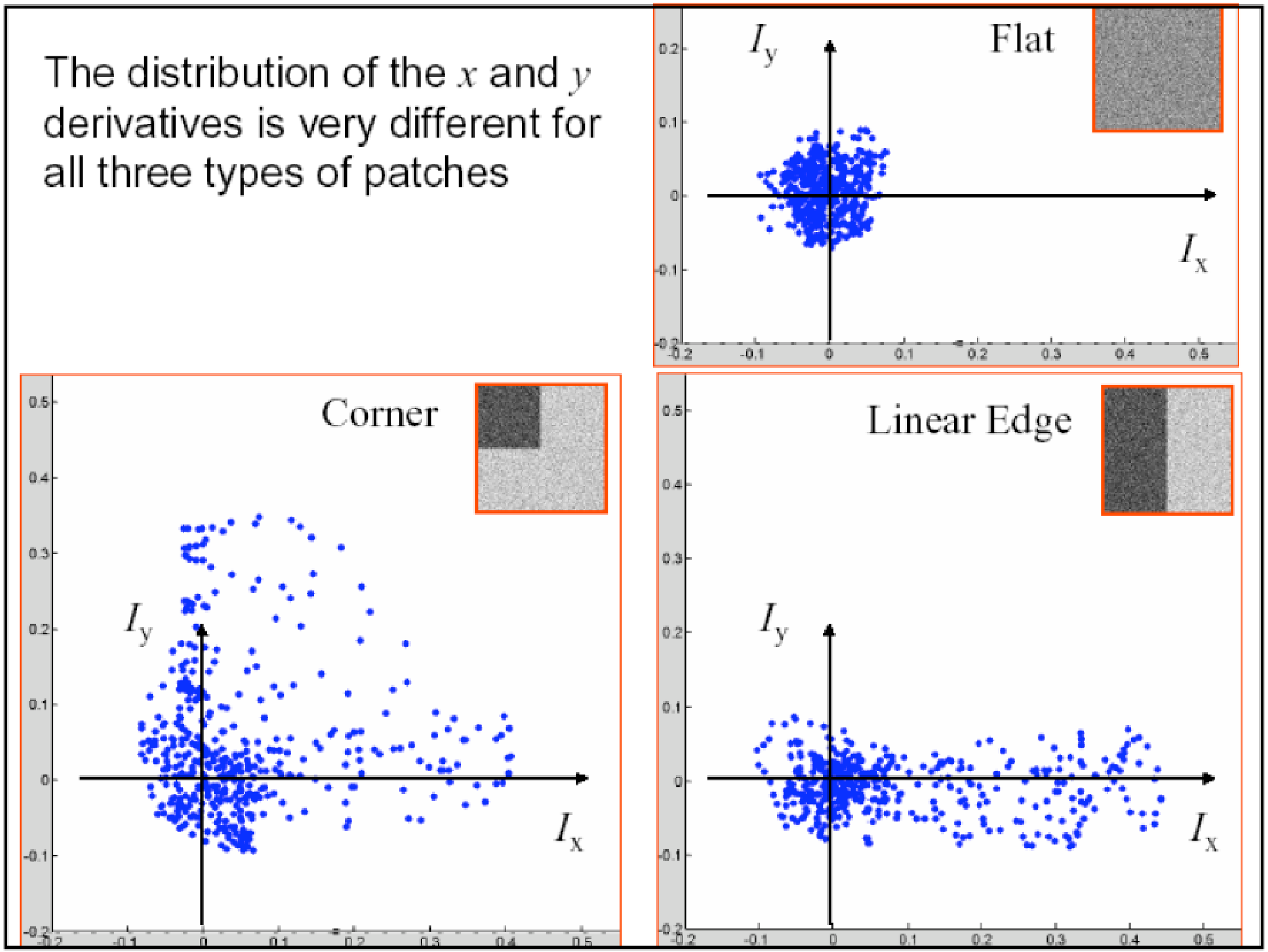
Examples



Slides adapted: https://www.coursehero.com/sitemap/schools/21-Penn-State/courses/645562-CSE486/

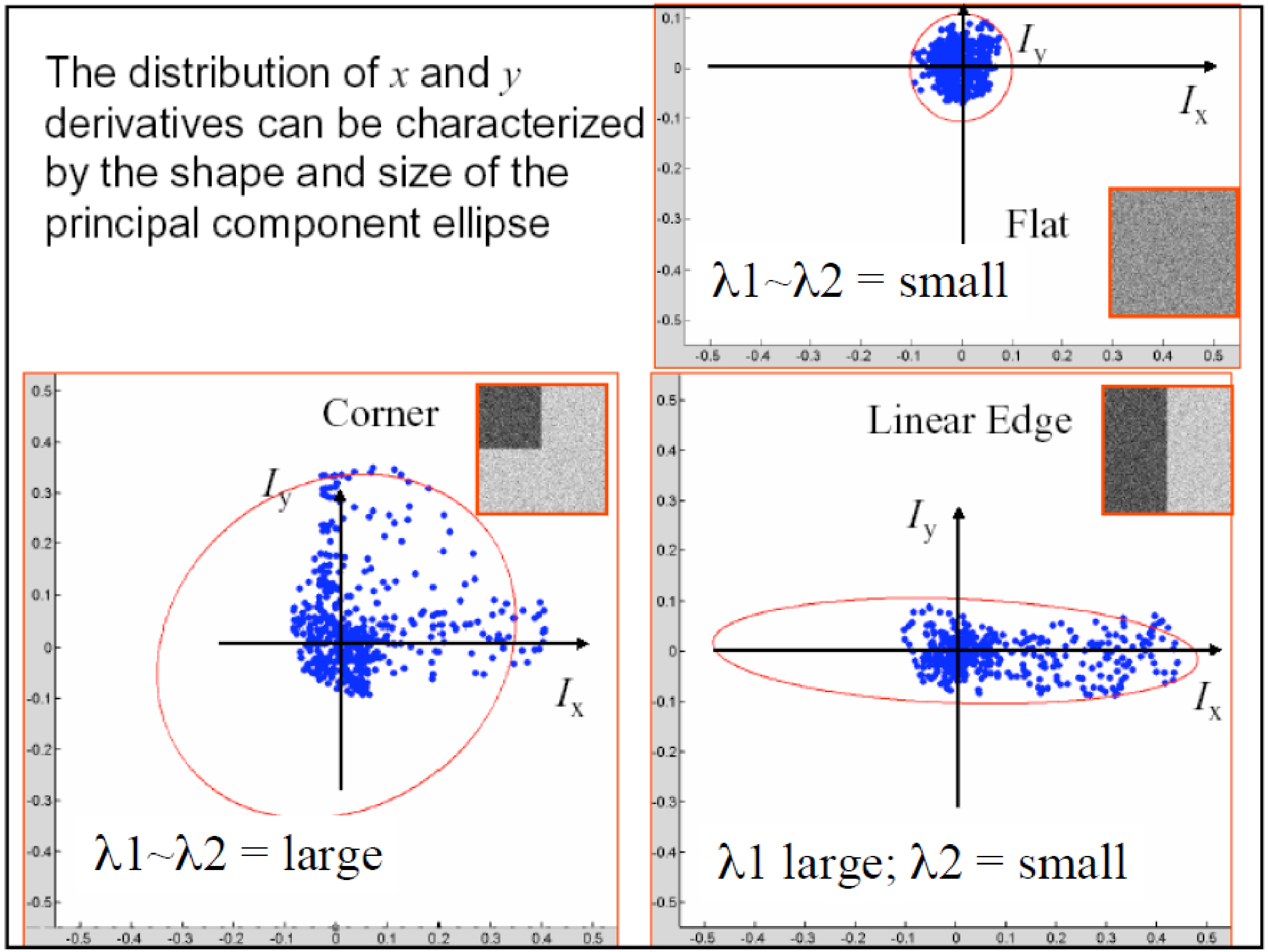
Examples

- Plotting derivatives as 2D points



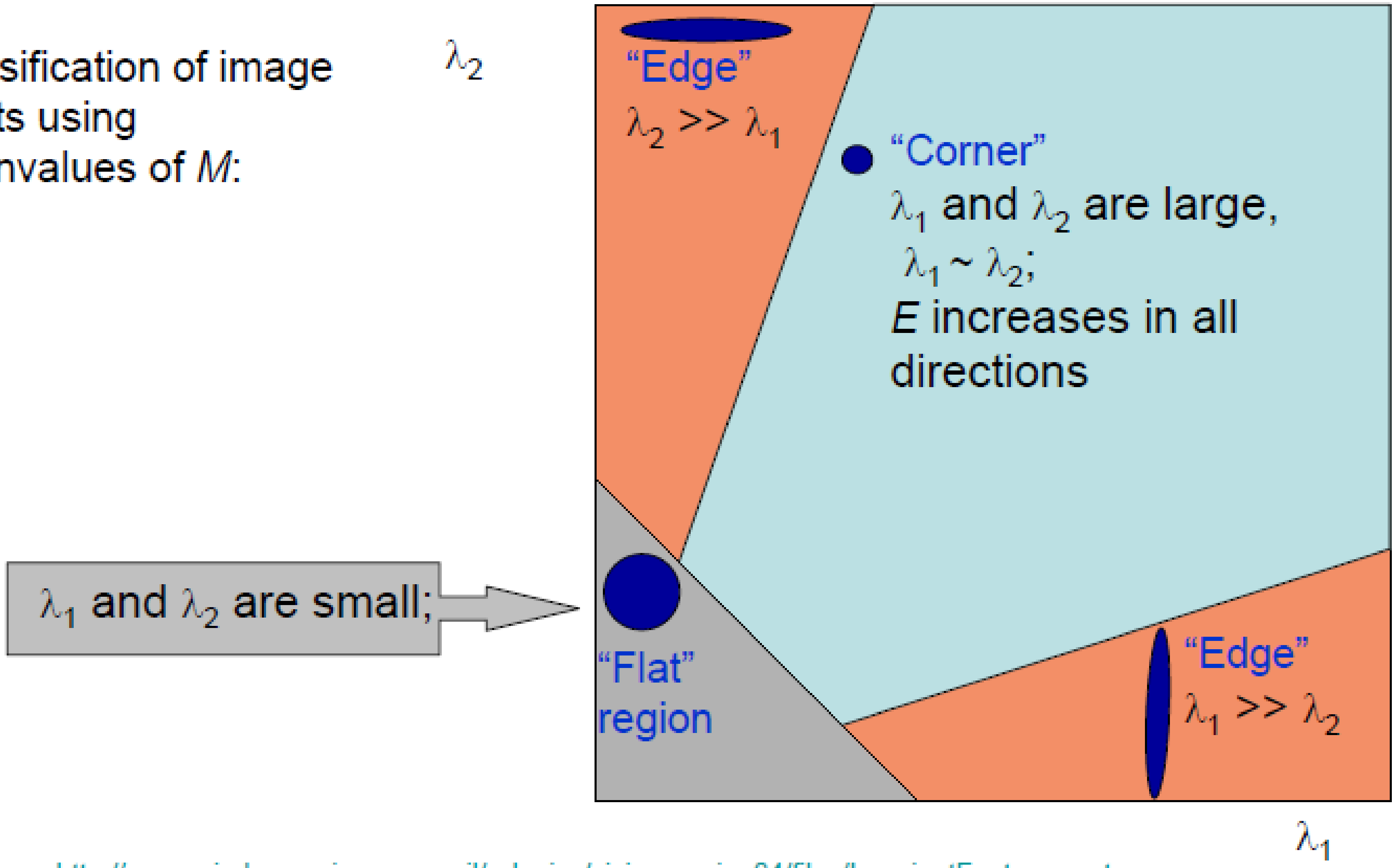
Examples

- Ellipse fitting



The Harris-Plessey detector

Classification of image points using eigenvalues of *M*:



Modified from: http://www.wisdom.weizmann.ac.il/~deniss/vision_spring04/files/InvariantFeatures.ppt

The Harris-Plessey detector

- In the Harris implementation, the corner is calculated as the ratio:

$$R_p = \frac{Trace(M)}{Det(M)}$$

- Thus, a point is marked as a corner if the value of R_p is less than the threshold and is the local minimum.
- Deemed unstable

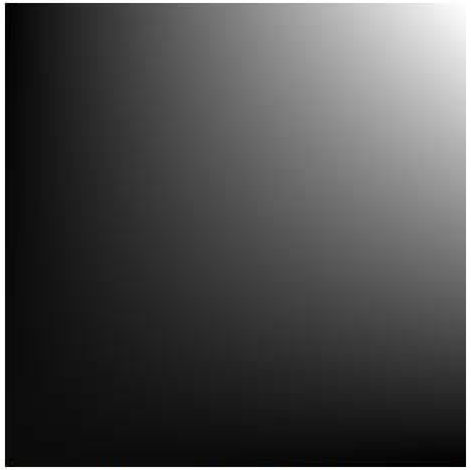
A good (corner) point should have a *large intensity change* in *all directions*, i.e. *R* should be large positive

The Harris-Plessey detector

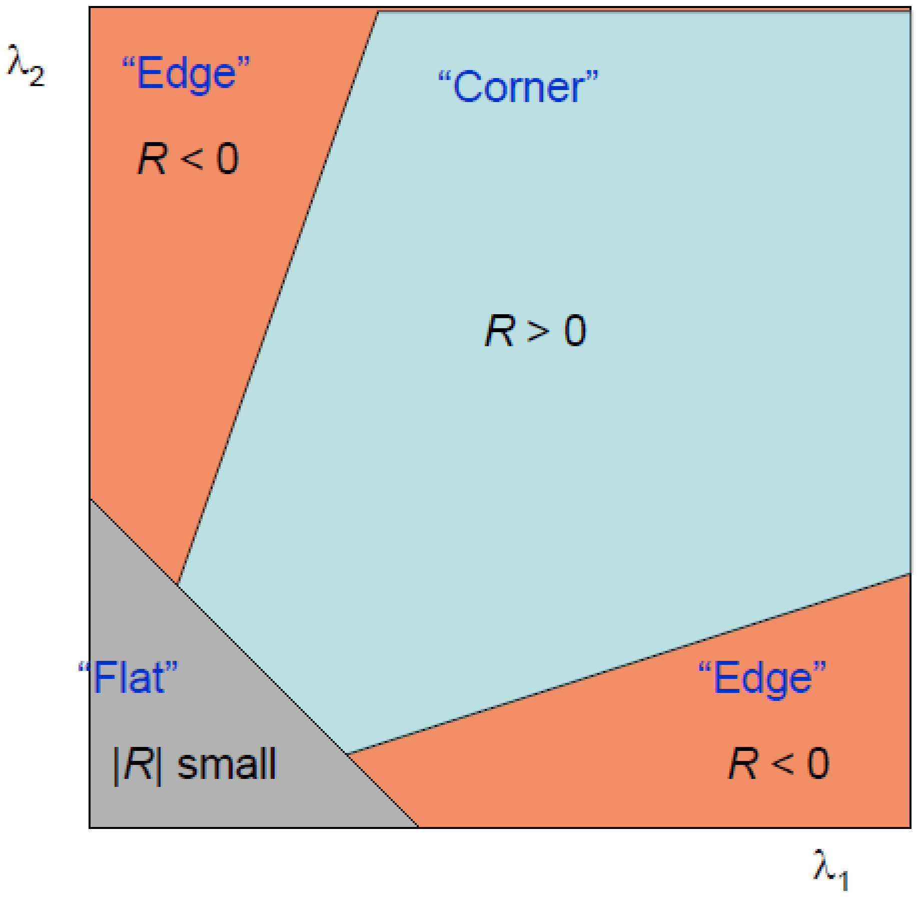
Measure of corner response:

- $R = Det(M) - k(Trace(M))^2$
- $Det(M) = \lambda_1 \lambda_2$
- $Trace(M) = \lambda_1 + \lambda_2$
- k – empirical constant, k = 0.04-0.15

- R depends only on eigenvalues of M
- R is large for a corner
- R is negative with large magnitude fc
- $|R|$ is small for a flat region



$R=\lambda_1*\lambda_2-0.04(\lambda_1+\lambda_2)^2$



Modified from: http://www.wisdom.weizmann.ac.il/~deniss/vision_spring04/files/InvariantFeatures.ppt

The 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

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

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

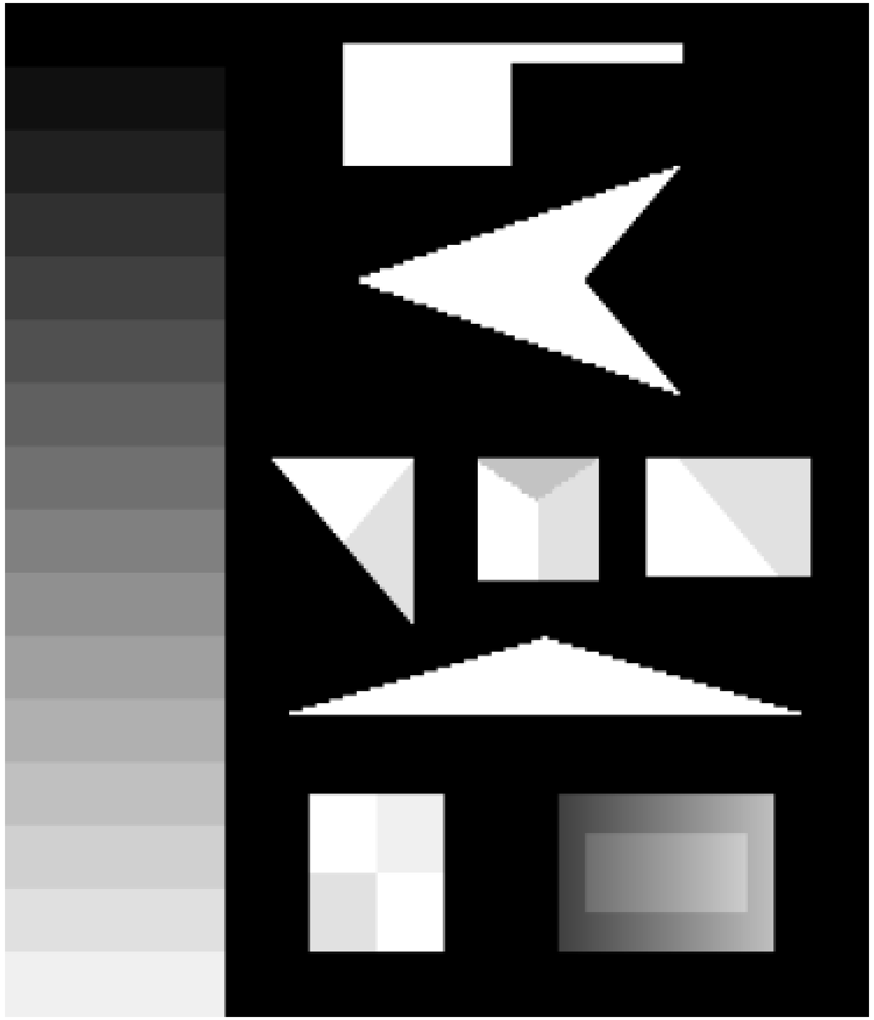
$$H(x,y) = \begin{bmatrix} S_{x2}(x,y) & S_{xy}(x,y) \\ S_{xy}(x,y) & S_{y2}(x,y) \end{bmatrix}$$

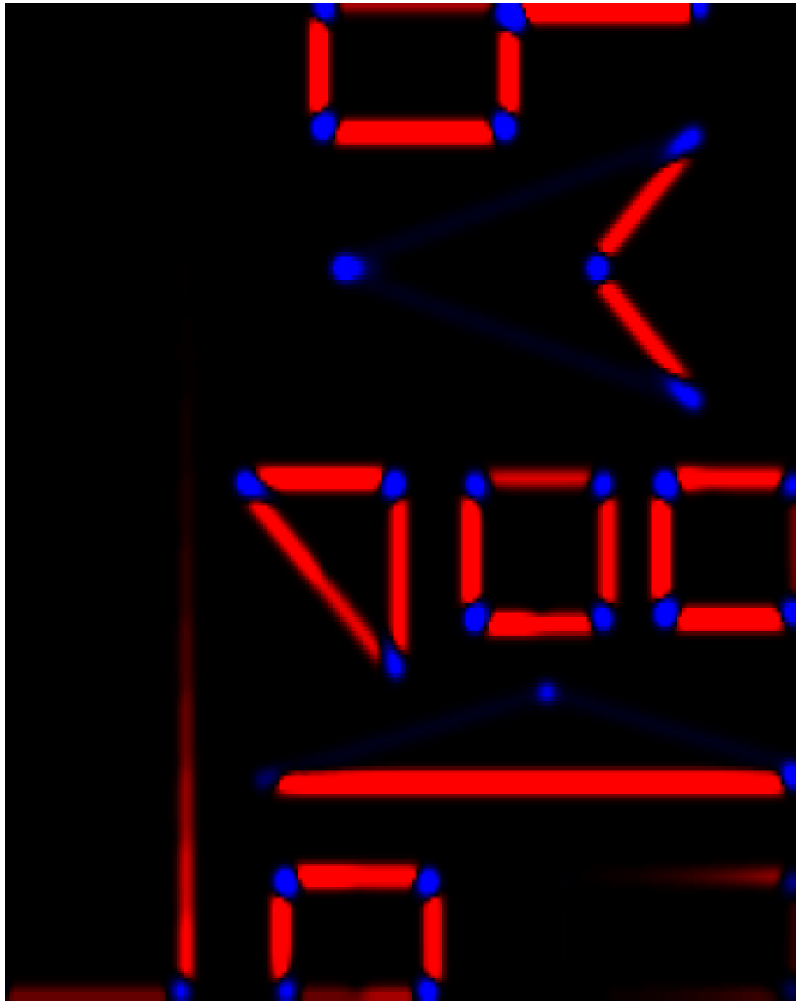
5. Compute the response of the detector at each pixel

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

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

Examples

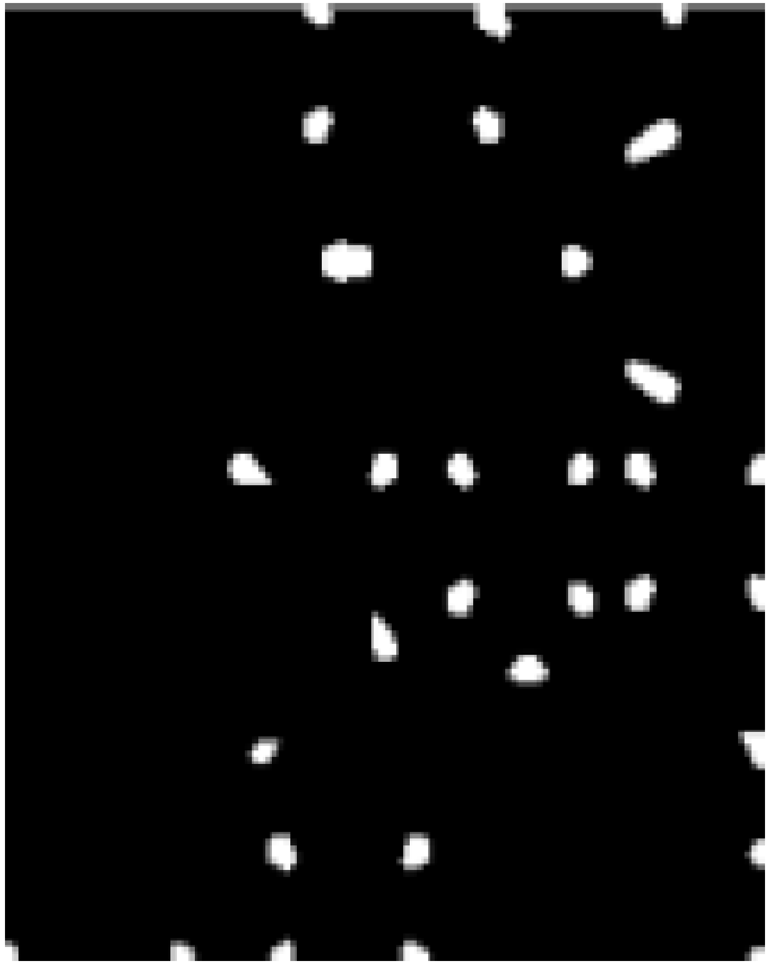




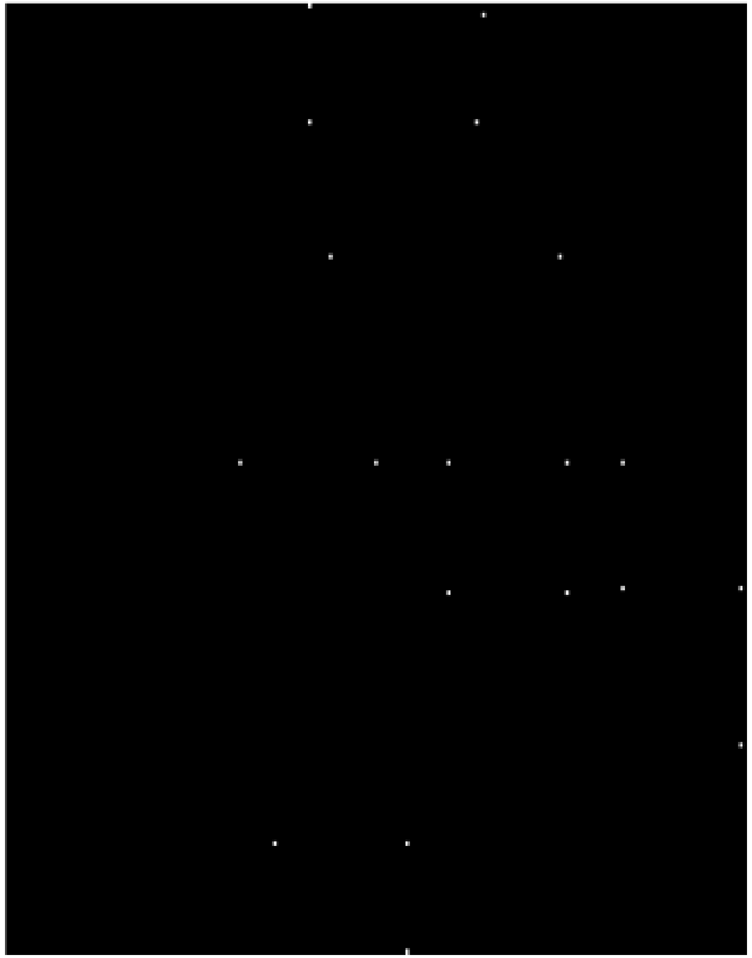
Coded *R: negative R in red; positive R in blue*

Examples

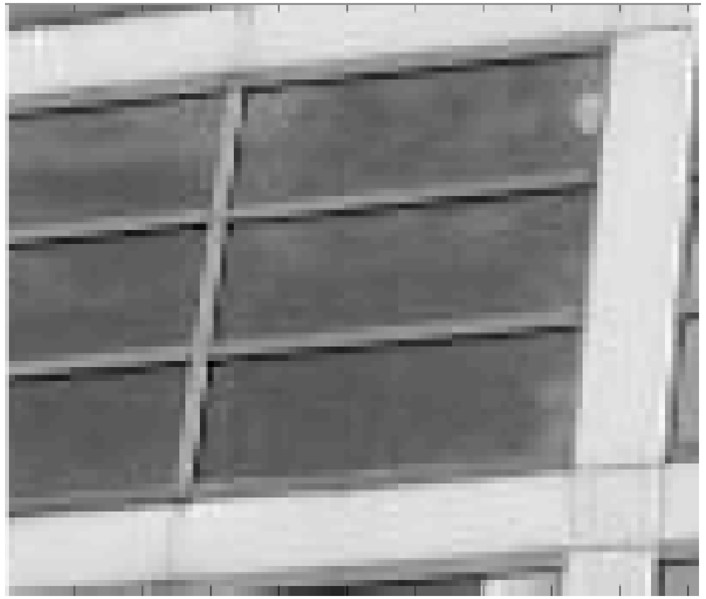
Find points with large corner response: *R*>threshold

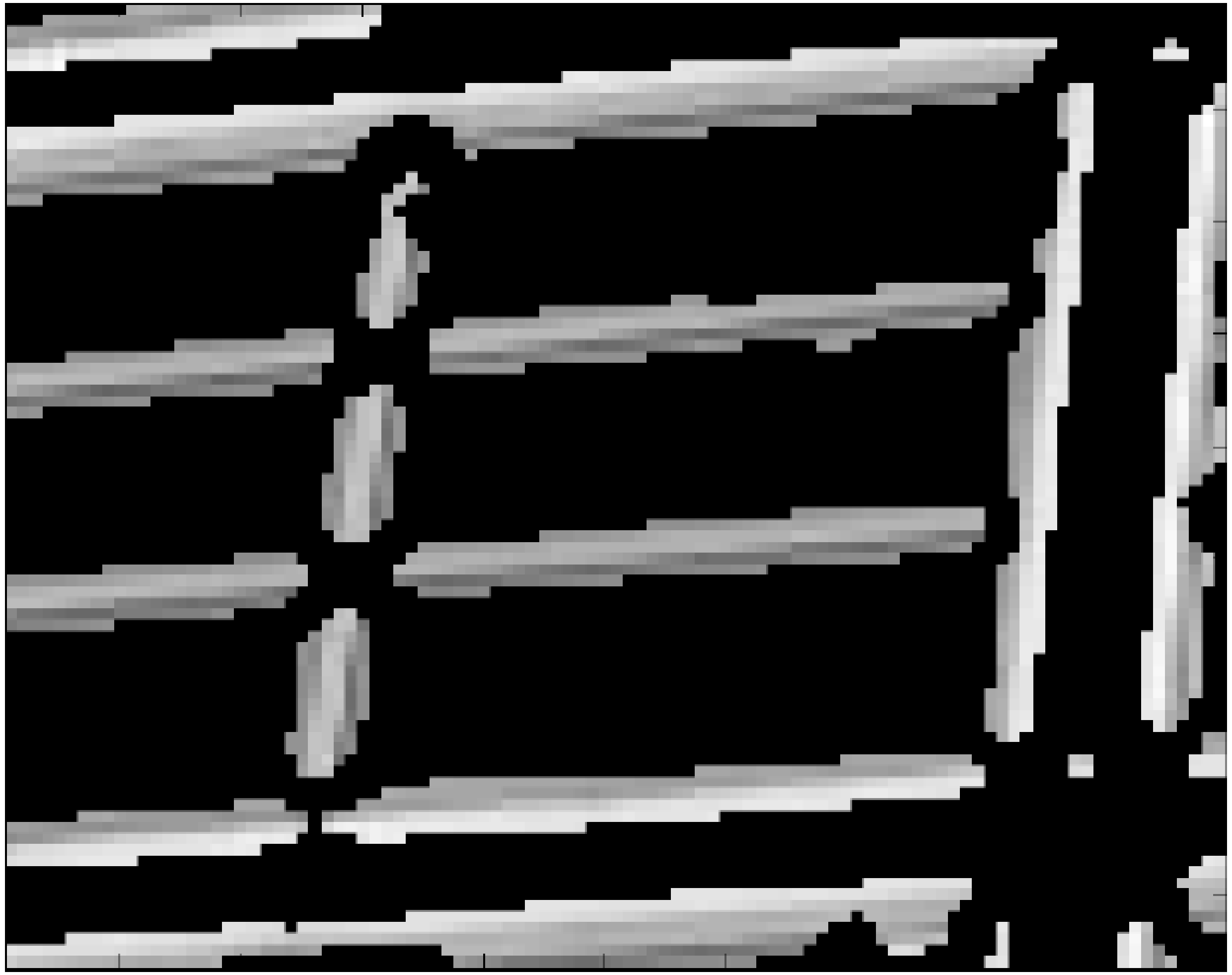


Take only the points of local maxima of *R*



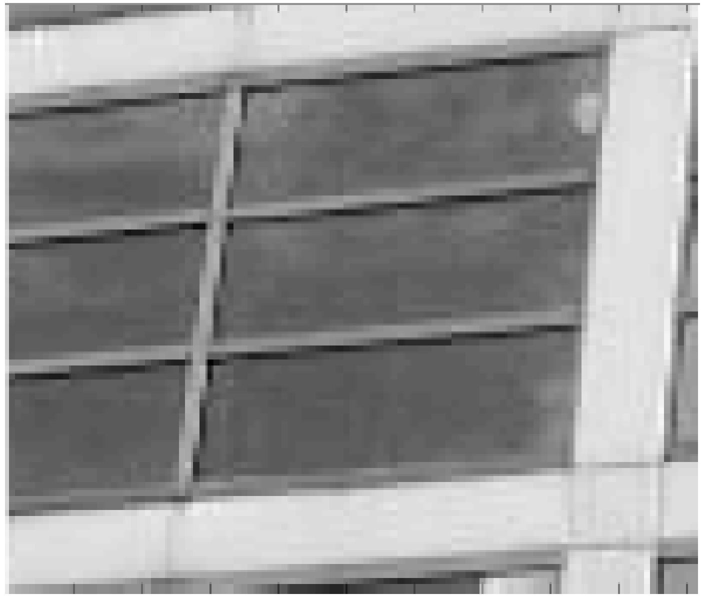
Examples

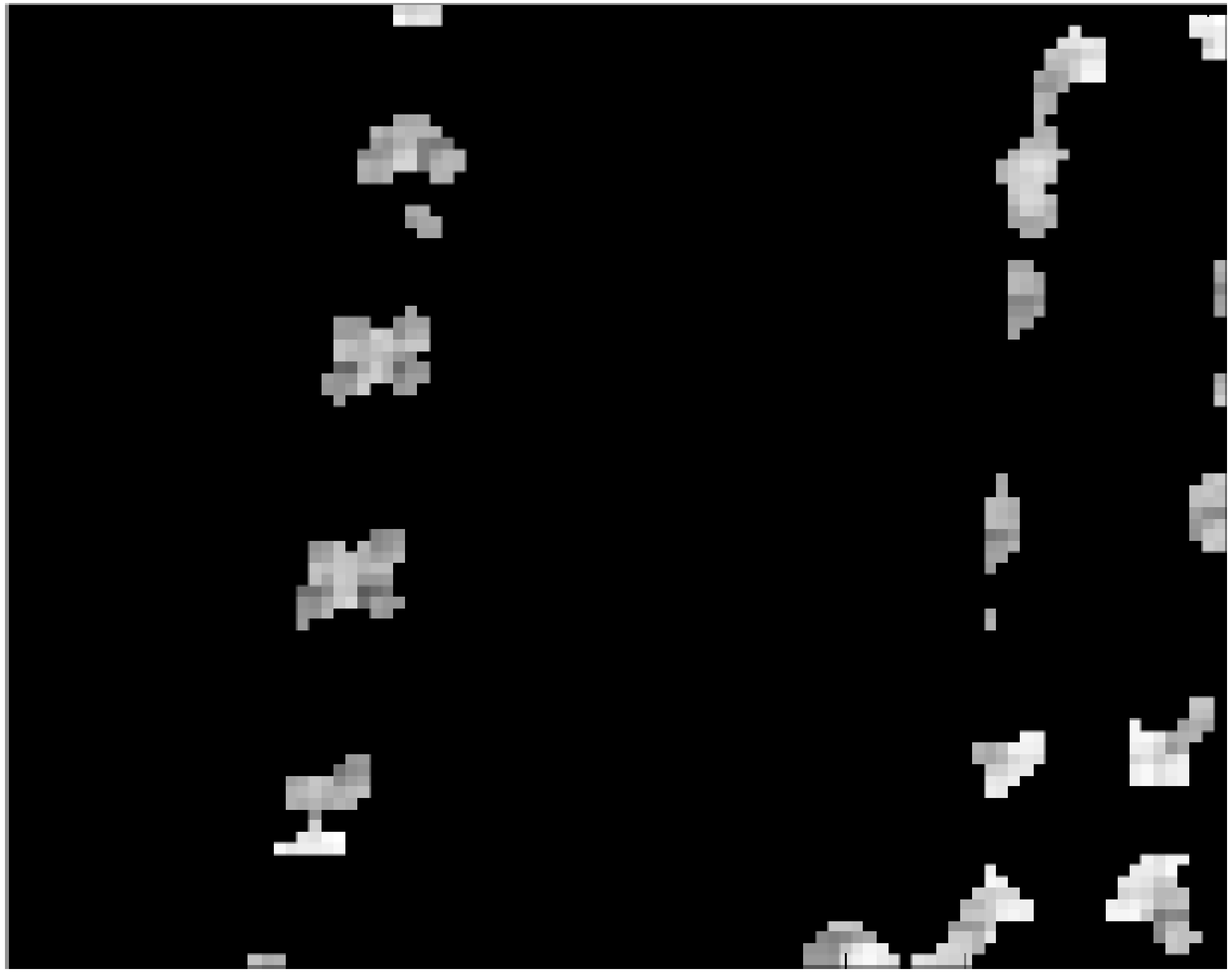




Threshold: R < -10000
(edges)

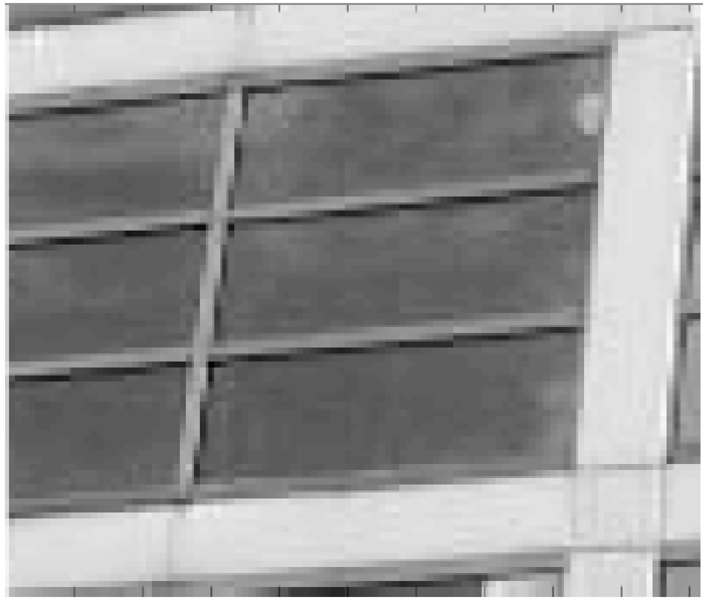
Examples

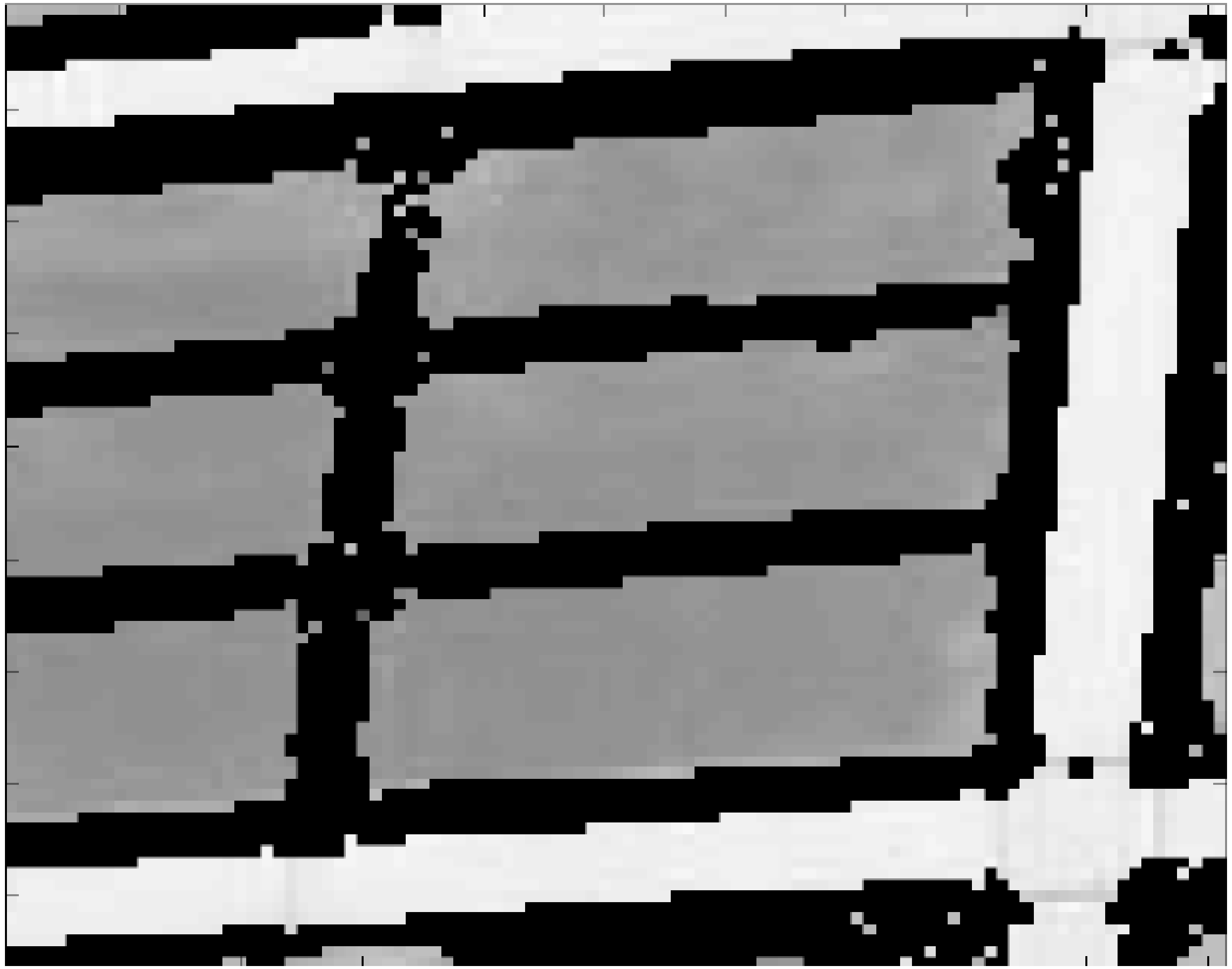




Threshold: > 10000
(corners)

Examples





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