Review

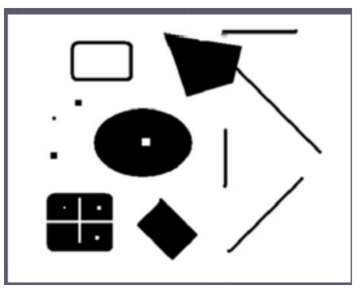
ECE P 596

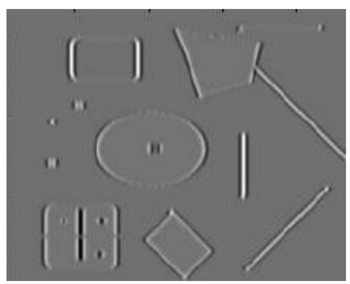
Linda Shapiro

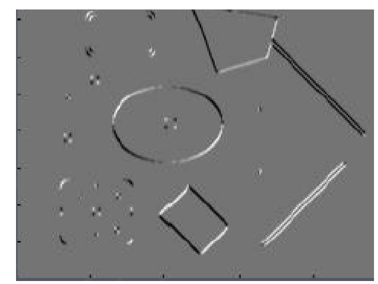
Second Moment Matrix or Harris Matrix

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

2 x 2 matrix of image derivatives smoothed by Gaussian weights.







Notation:

$$I_x \Leftrightarrow \frac{\partial I}{\partial x} \qquad I_y \Leftrightarrow \frac{\partial I}{\partial y} \qquad I_x I_y \Leftrightarrow \frac{\partial I}{\partial x}\frac{\partial I}{\partial y}$$

The math

To compute the eigenvalues:

1. Compute the Harris matrix over a window.

$$H = \sum_{(u,v)} w(u,v) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

Typically Gaussian weights

What does this equation mean in practice?

$$\begin{bmatrix} \Sigma\text{smoothed } I_x^2 & \Sigma\text{smoothed } I_x I_y \\ \Sigma\text{smoothed } I_x I_y & \Sigma\text{smoothed } I_y^2 \end{bmatrix}$$

2. Compute response from that.

$$I_x = \frac{\partial f}{\partial x}, I_y = \frac{\partial f}{\partial y}$$

This is how people write it for technical papers

This is how you DO it.

You just smooth with Gaussian as you add up the derivatives.

Corner Response Function

- Computing eigenvalues are expensive
- Harris corner detector used the following alternative

$$R = det(M) - \alpha \cdot trace(M)^2$$

Reminder:

$$det\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right) = ad - bc \qquad trace\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right) = a + d$$

Harris detector: Steps (modified to simplify)

1. Compute derivatives I_x^2, I_y^2 and I_xI_y at each pixel and smooth them with a Gaussian as you sum them to
2. Compute the Harris matrix H in a window around each pixel
3. Compute corner response function *R*
4. Threshold *R*
5. Find local maxima of response function (nonmaximum suppression)

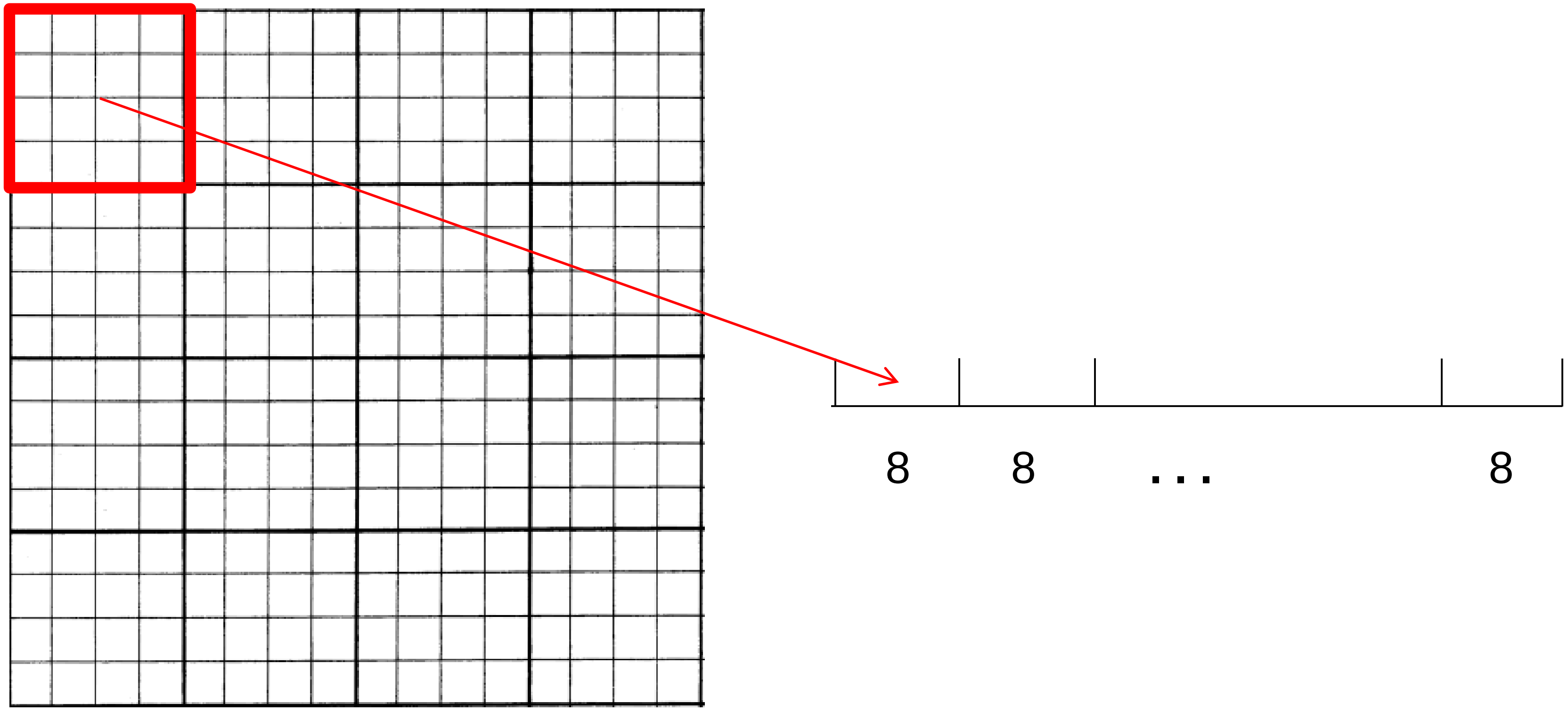
C.Harris and M.Stephens. *Proceedings of the 4th Alvey Vision Conference*: pages 147—151, 1988.

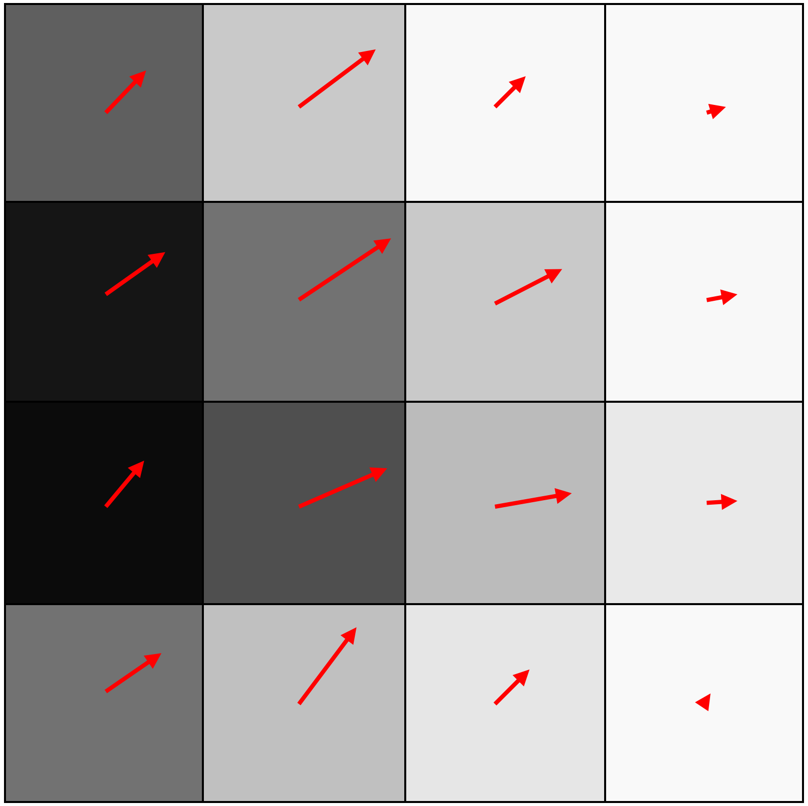
Harris Detector: Results

SIFT descriptor

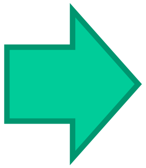
Full version

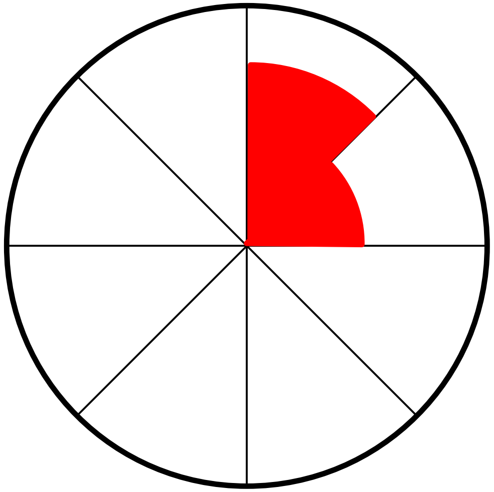
- Divide the 16x16 window into a 4x4 grid of cells
- Compute an orientation histogram for each cell
- 16 cells * 8 orientations = 128 dimensional descriptor





Orientations in each of the 16 pixels of the cell





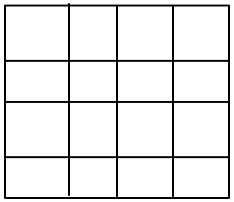
The orientations all ended up in two bins: 11 in one bin, 5 in the other. (rough count)

5 11 0 0 0 0 0 0

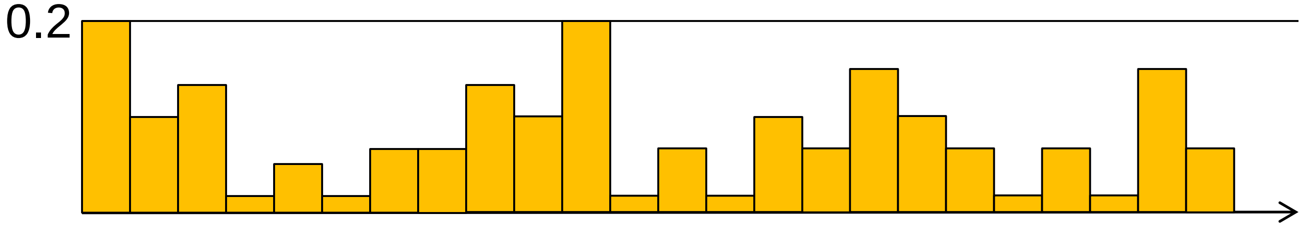
SIFT descriptor

Full version

- Start with a 16x16 window (256 pixels)
- Divide the 16x16 window into a 4x4 grid of cells (16 cells)
- Compute an orientation histogram for each cell
- 16 cells * 8 orientations = 128 dimensional descriptor
- Threshold normalize the descriptor:



$$\sum_i d_i^2 = 1 \quad \text{such that: } d_i < 0.2$$



Adapted from slide by David Lowe

Matching with Features

- Detect feature points in both images
- Find corresponding pairs
- Use these matching pairs to align images - the required mapping is called a homography.