# Interest Points: Corners

Slides adapted from Alyosha Efros, Lana Lazebnik, Richard Szeliski

# Interest Points (or Image Features)

- Story so far…
  - Not all pixels are created equal
  - Some are more “interesting” than others
- Previously…
  - Pixels that lie on edges are interesting
- Today: Corners

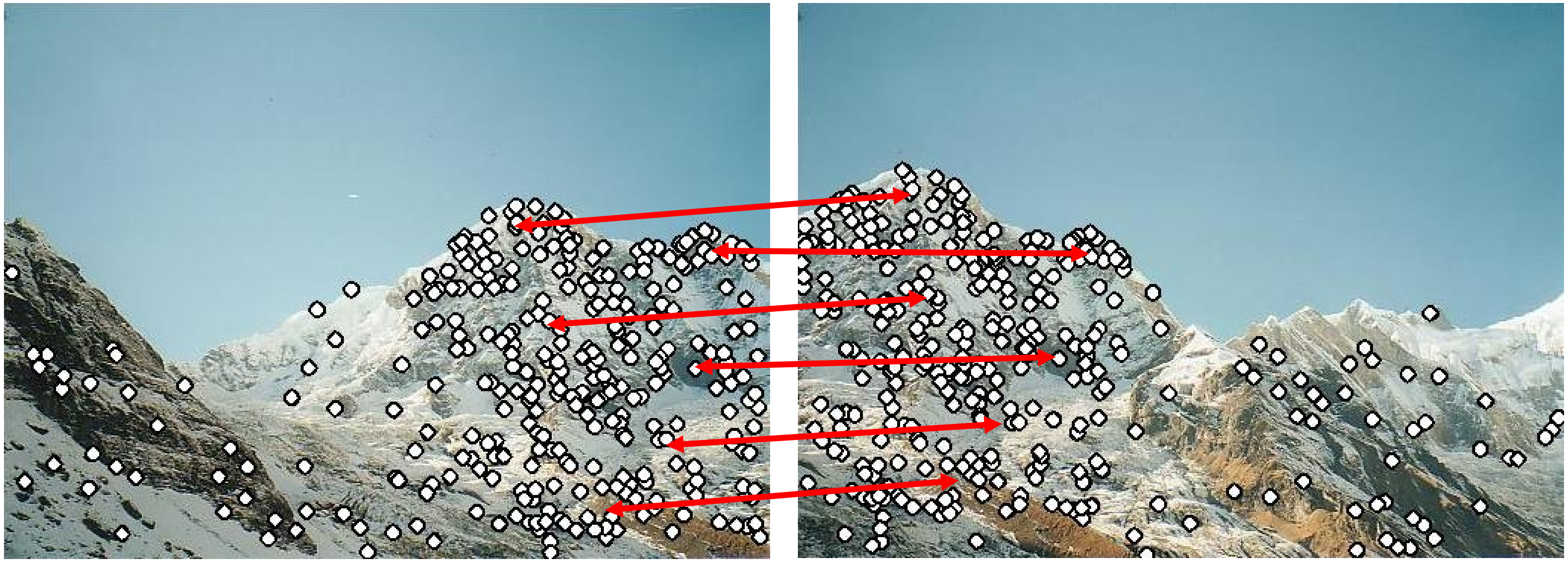
# Corners

# Image Feature Motivation

- Panorama stitching
  - We have two images – how do we combine them?


# Image Feature Motivation

- Panorama stitching
  - We have two images – how do we combine them?



Step 1: find features

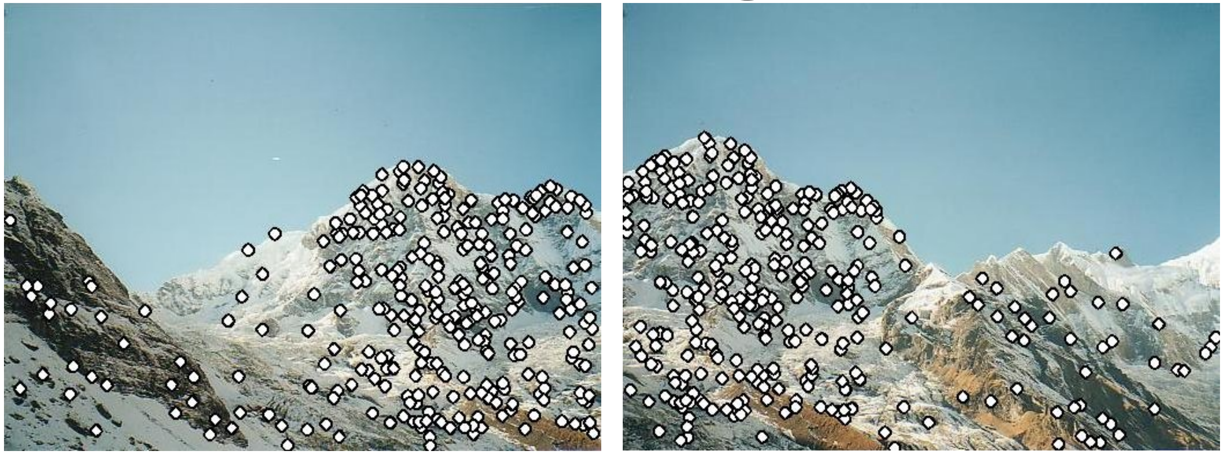
Step 2: match features (we'll cover later)

# Image Feature Motivation

- Panorama stitching
  - We have two images – how do we combine them?

Step 3: align images (we'll cover later)

# Characteristics of good features



- Repeatability
  - The same feature can be found in several images despite geometric and photometric transformations
- Saliency
  - Each feature has a distinctive description
- Compactness and efficiency
  - Many fewer features than image pixels
- Locality
  - A feature occupies a relatively small area of the image; robust to clutter and occlusion

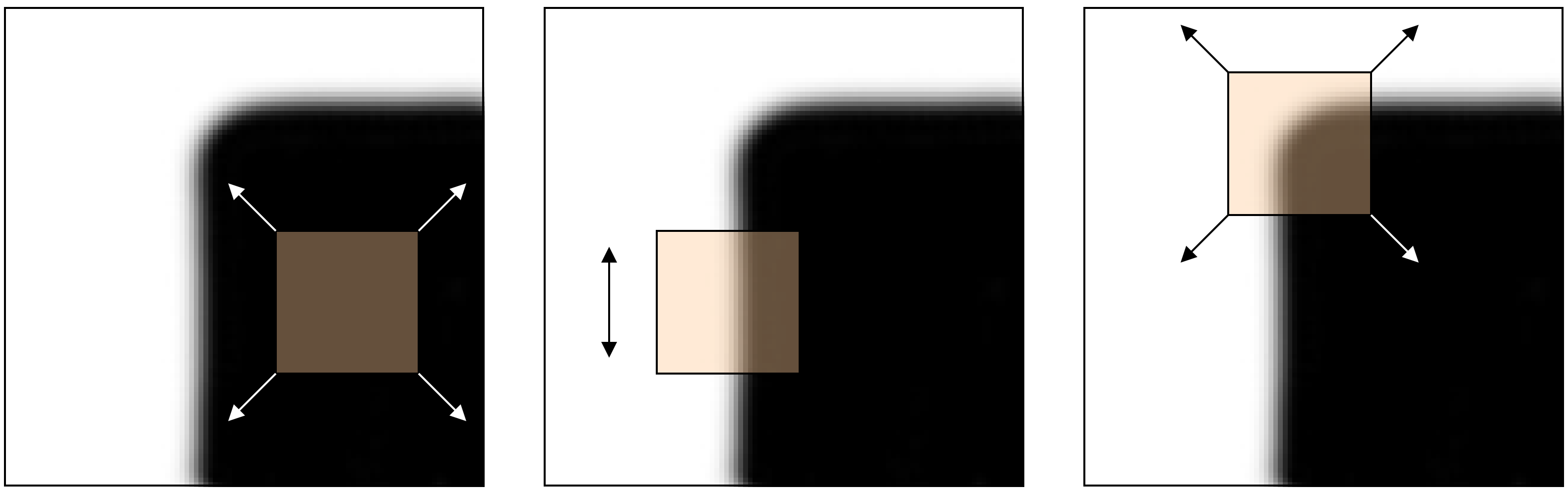
Corners form the basis form many modern image features

# Applications

- Feature points are used for:
  - Image alignment
  - 3D reconstruction
  - Motion tracking
  - Robot navigation
  - Indexing and database retrieval
  - Object recognition

# Corner Detection: Basic Idea

- We should easily recognize the point by looking through a small window
- Shifting a window in *any direction* should give *a large change* in intensity



"flat" region: no change in all directions

"edge": no change along the edge direction

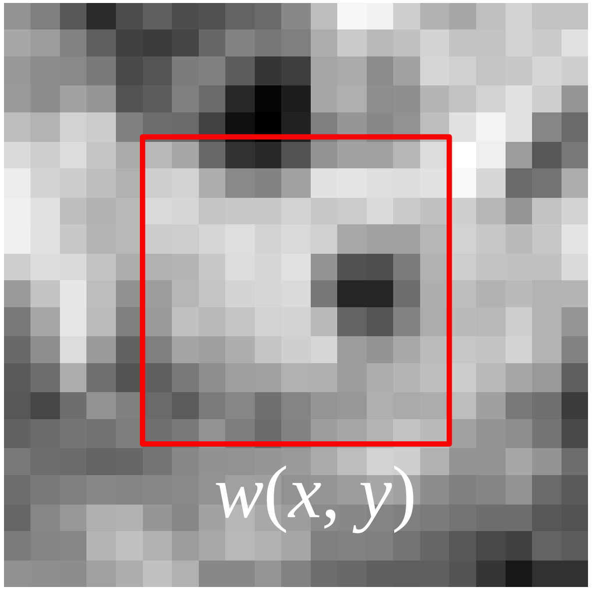
"corner": significant change in all directions

# Corner Detection: Mathematics

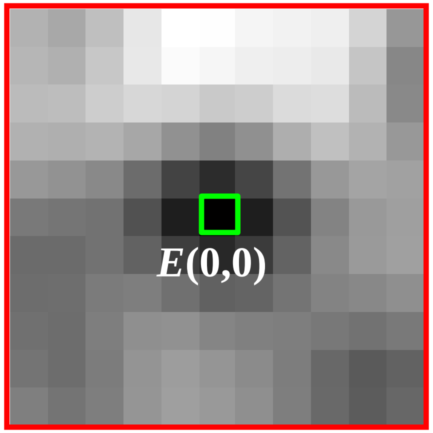
Change in appearance of window $w(x,y)$ for the shift $[u,v]$:

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

$I(x, y)$




$E(u, v)$

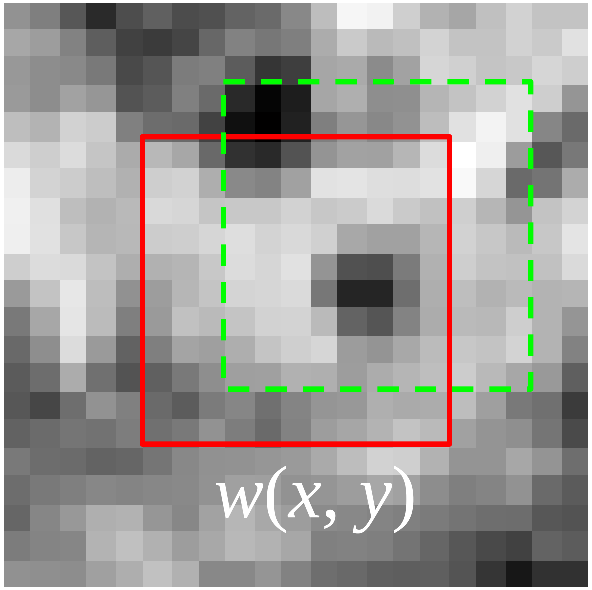


# Corner Detection: Mathematics

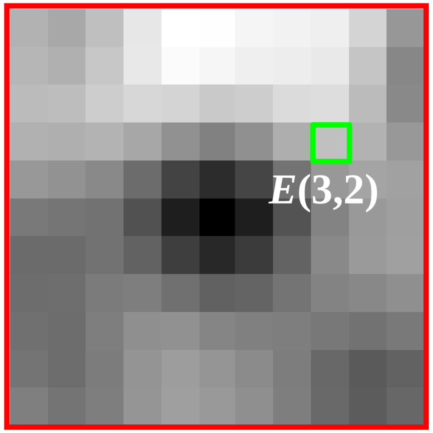
Change in appearance of window $w(x,y)$ for the shift $[u,v]$:

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

$I(x, y)$




$E(u, v)$



# Corner Detector: Mathematics

Change in appearance for the shift [*u,v*]:

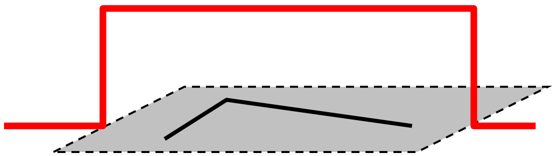
$$E(u,v) = \sum_{x,y} w(x,y)\left[I(x+u,y+v) - I(x,y)\right]^2$$

Window function

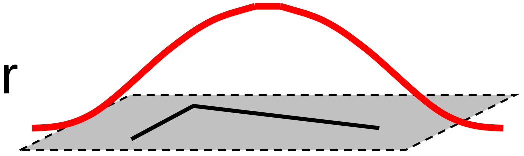
Shifted intensity

Intensity

Window function $w(x,y)$ =



or



1 in window, 0 outside

Gaussian

*However, we are not actually going slide this window around for every (u,v) at each pixel (x,y) to calculate E. We will solve this analytically.*

# Harris Detector: Mathematics

- First-order Taylor approximation for small motions [*u*, *v*]:

$$I(x+u, y+v) \approx I(x,y) + I_x u + I_y v$$

- Let's plug this into *E*(*u*,*v*):

$$\begin{aligned}
E(u,v) &= \sum_{(x,y)\in W} [I(x+u, y+v) - I(x,y)]^2 \\
&\approx \sum_{(x,y)\in W} [I(x,y) + I_x u + I_y v - I(x,y)]^2 \\
&= \sum_{(x,y)\in W} [I_x u + I_y v]^2 = \sum_{(x,y)\in W} I_x^2 u^2 + 2 I_x I_y uv + I_y^2 v^2
\end{aligned}$$

# Harris Detector: Mathematics

The bilinear approximation simplifies to

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

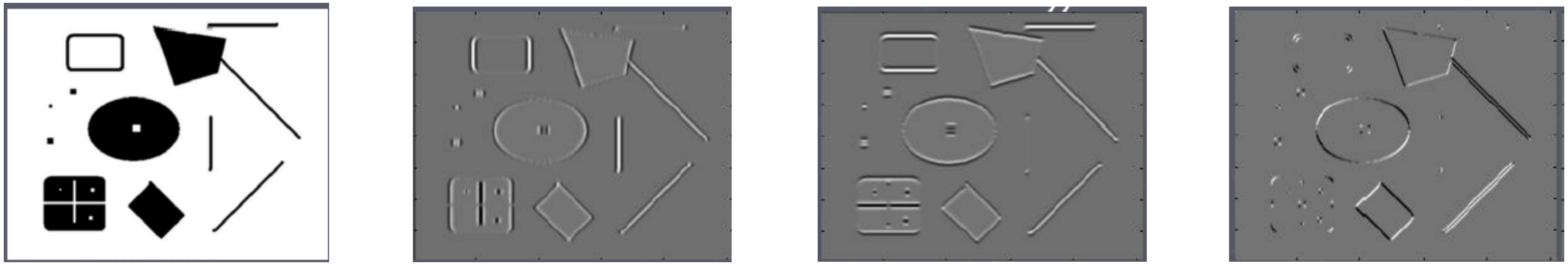
where $M$ is a ***second moment matrix*** computed from image derivatives:

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

# **Corners** as distinctive interest points

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

2 x 2 matrix of image derivatives (averaged in neighborhood of a point).



Notation:

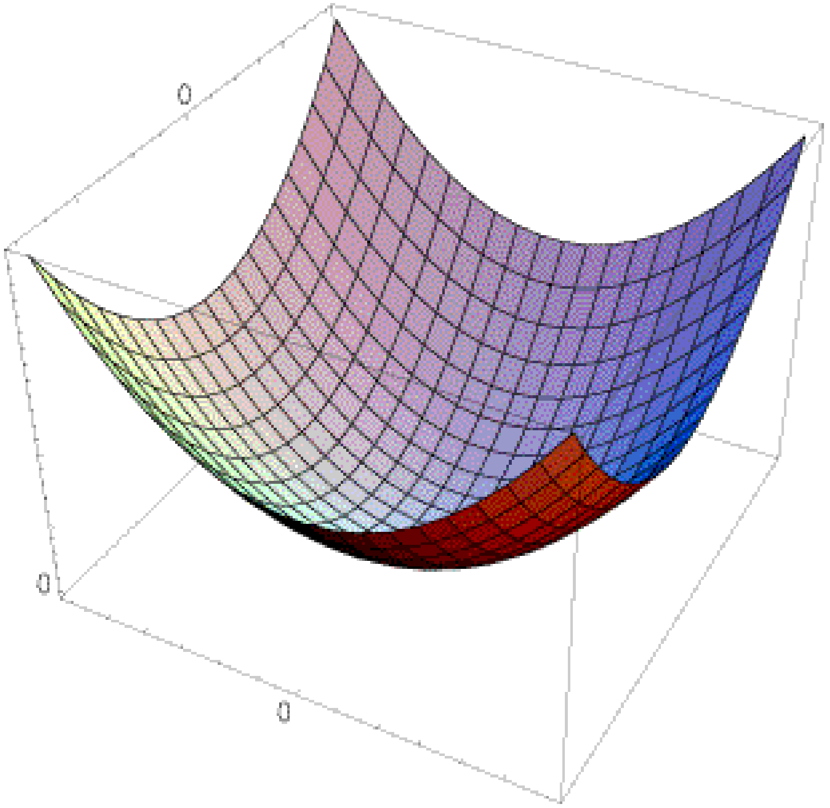
$$I_x \Leftrightarrow \frac{\partial I}{\partial x} \qquad I_y \Leftrightarrow \frac{\partial I}{\partial y} \qquad I_x I_y \Leftrightarrow \frac{\partial I}{\partial x}\frac{\partial I}{\partial y}$$

# Interpreting the 2nd Moment Matrix

The surface $E(u,v)$ is locally approximated by a quadratic form. Let's try to understand its shape.

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

$$M = \sum \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$



# Interpreting the 2nd moment matrix

First, consider the axis-aligned case (gradients are either horizontal or vertical)

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

If either λ is close to 0, then this is **not** a corner, so look for locations where both are large.

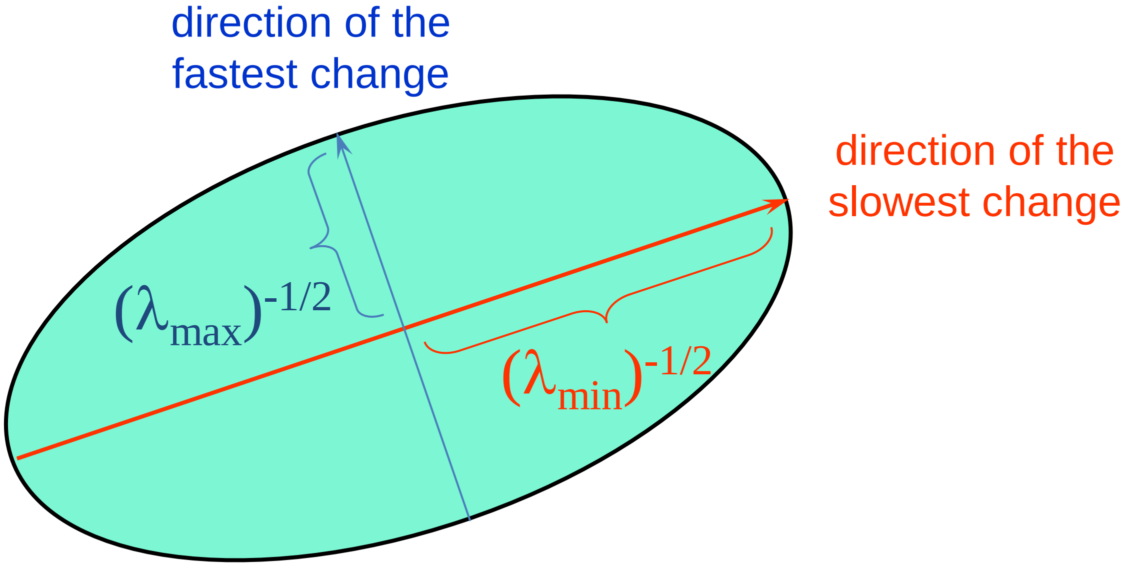
# General Case

Since M is symmetric, we have $M = R^{-1}\begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R$

We can visualize *M* as an ellipse with axis lengths determined by the eigenvalues and orientation determined by *R*

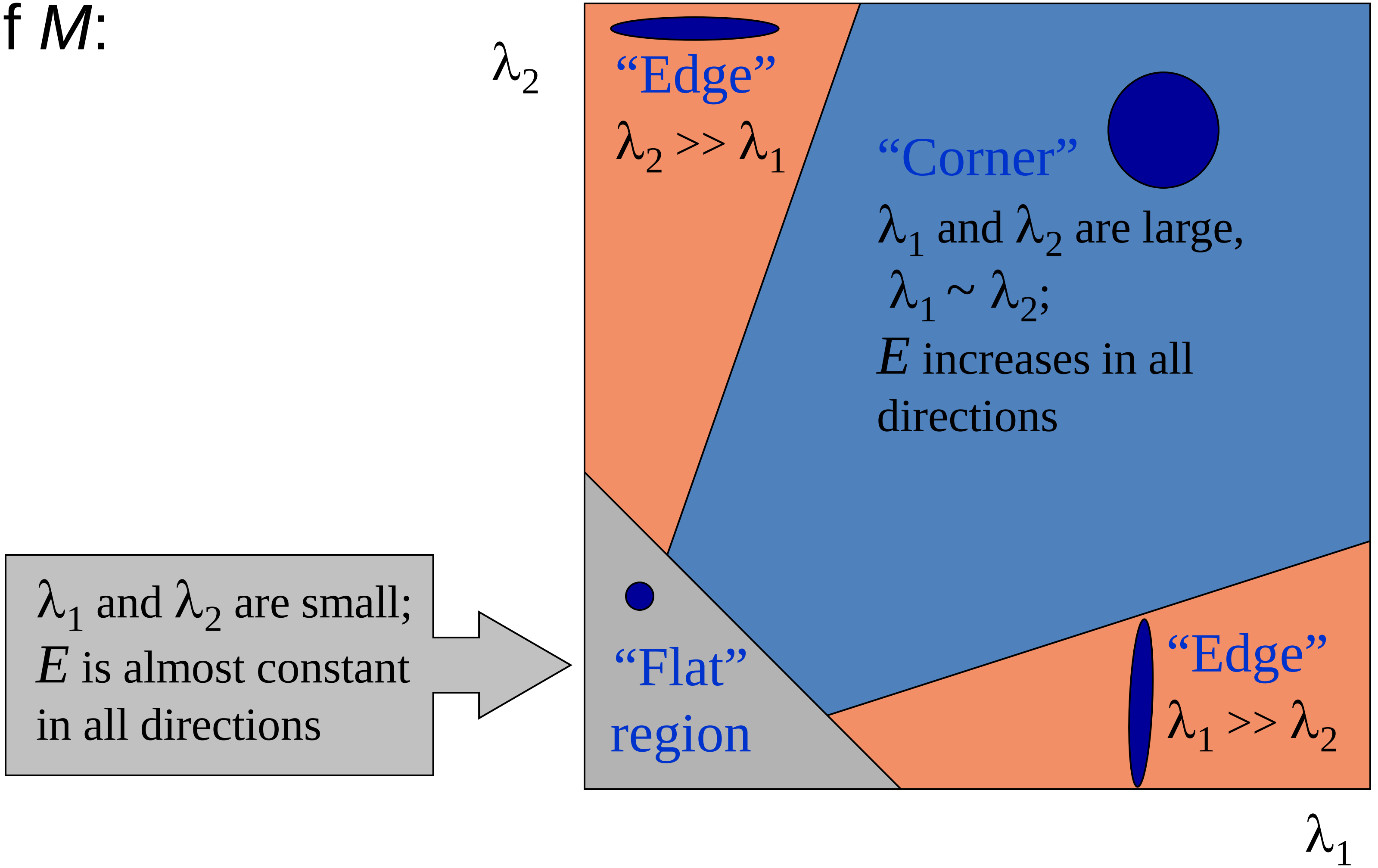
Ellipse equation:

$$[u \;\; v] \; M \begin{bmatrix} u \\ v \end{bmatrix} = \text{const}$$



# Interpreting the eigenvalues

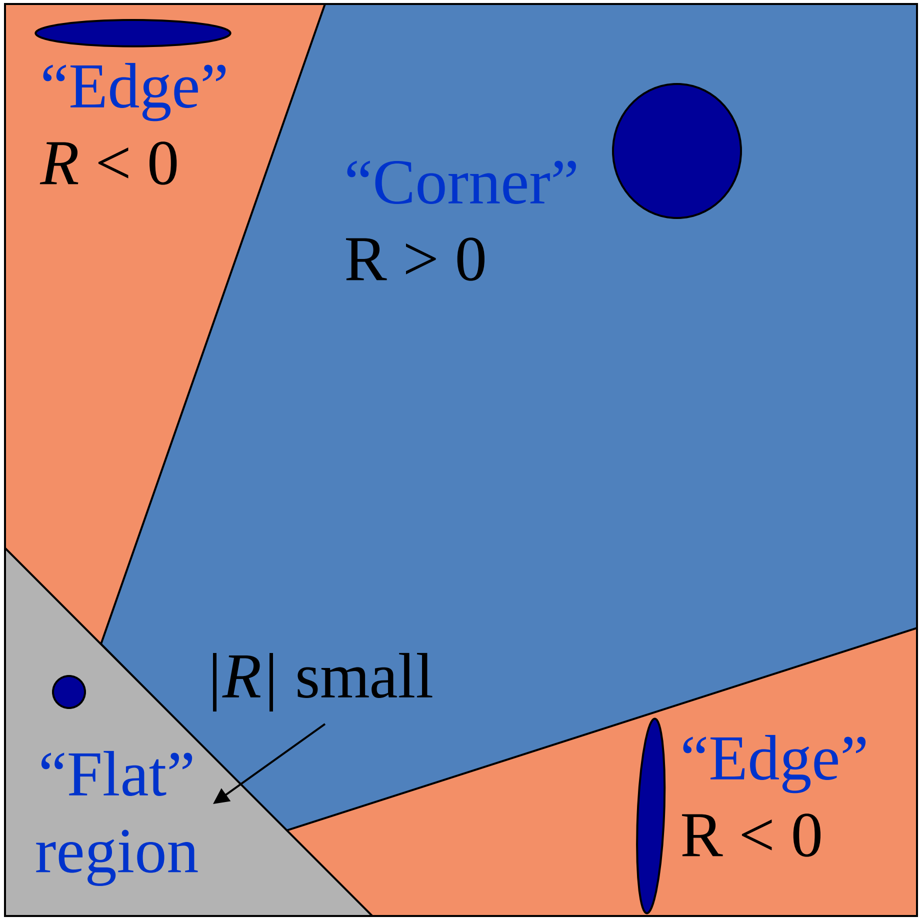
Classification of image points using eigenvalues of *M*:



# Corner response function

$$R = \det(M) - \alpha \,\mathrm{trace}(M)^2 = \lambda_1\lambda_2 - \alpha(\lambda_1 + \lambda_2)^2$$

$\alpha$: constant (0.04 to 0.06)



# Harris Corner Detector: Steps

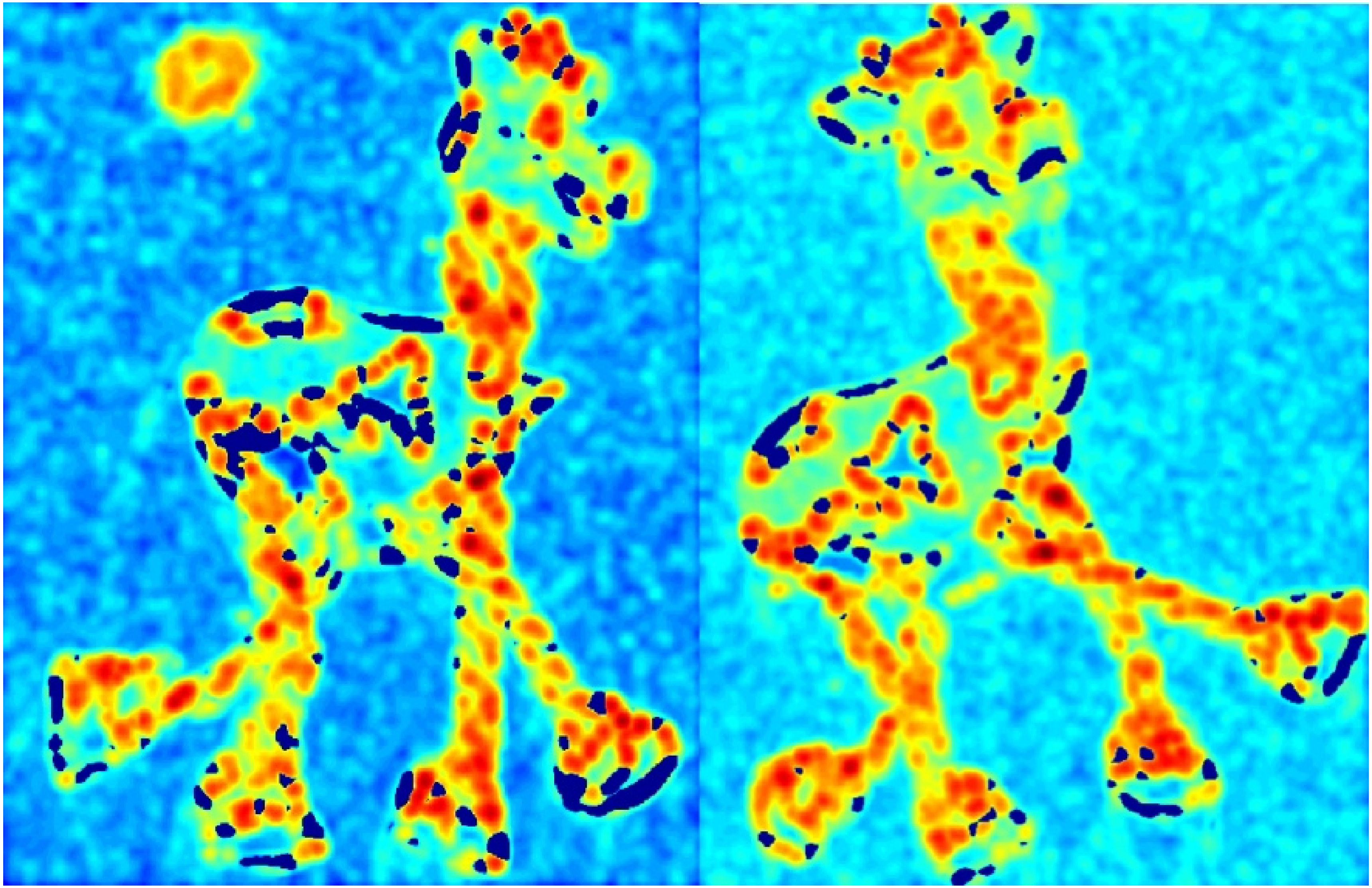
1. Compute Gaussian derivatives at each pixel
2. Compute second moment matrix *M* in a Gaussian window around each pixel
3. Compute corner response function *R*
4. Threshold *R*
5. Find local maxima of response function (nonmaximum suppression)

C.Harris and M.Stephens. [“A Combined Corner and Edge Detector.”] *Proceedings of the 4th Alvey Vision Conference*: pages 147—151, 1988.

# Harris Detector: Steps

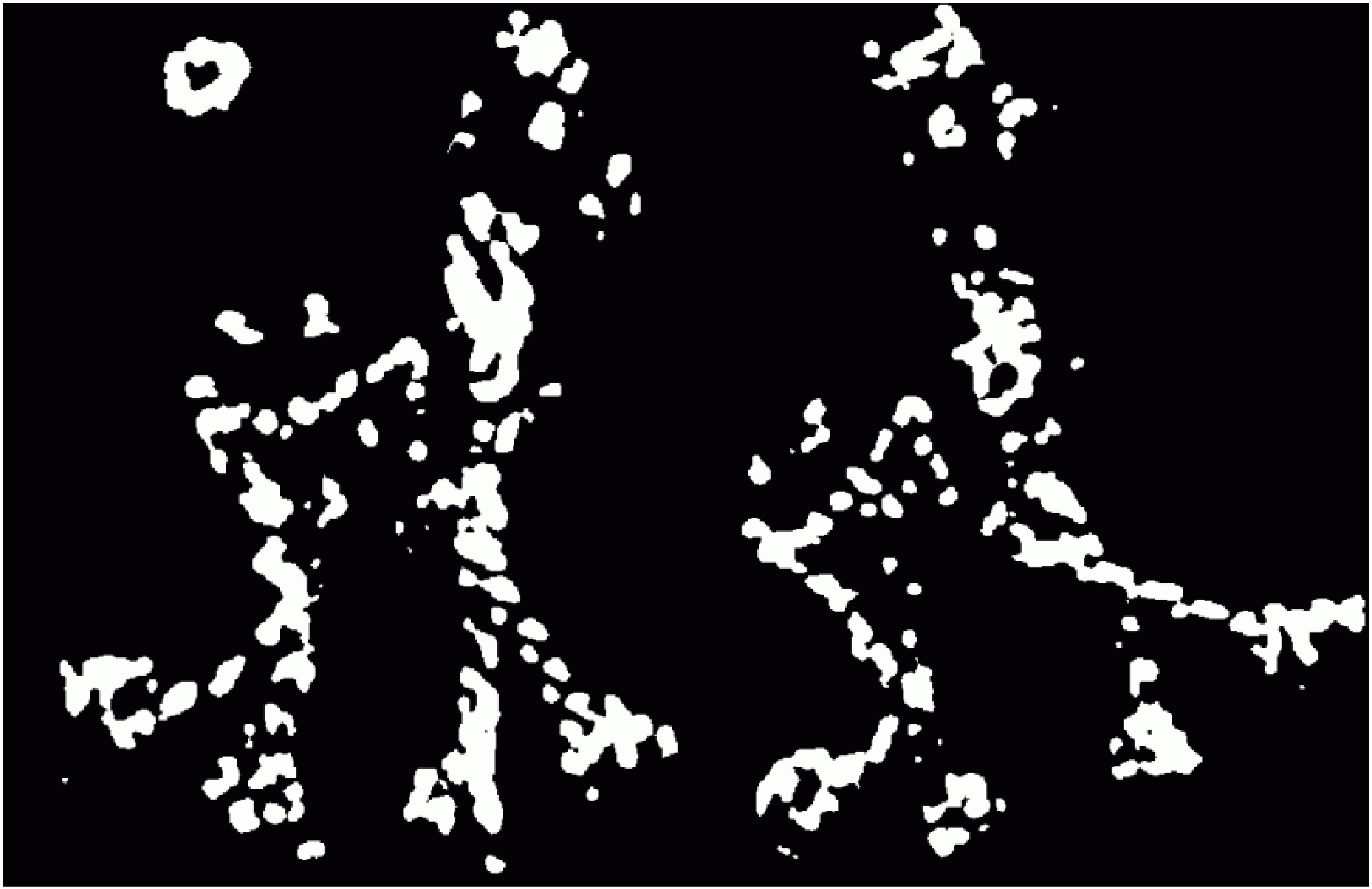
# Harris Detector: Steps

Compute corner response $R$



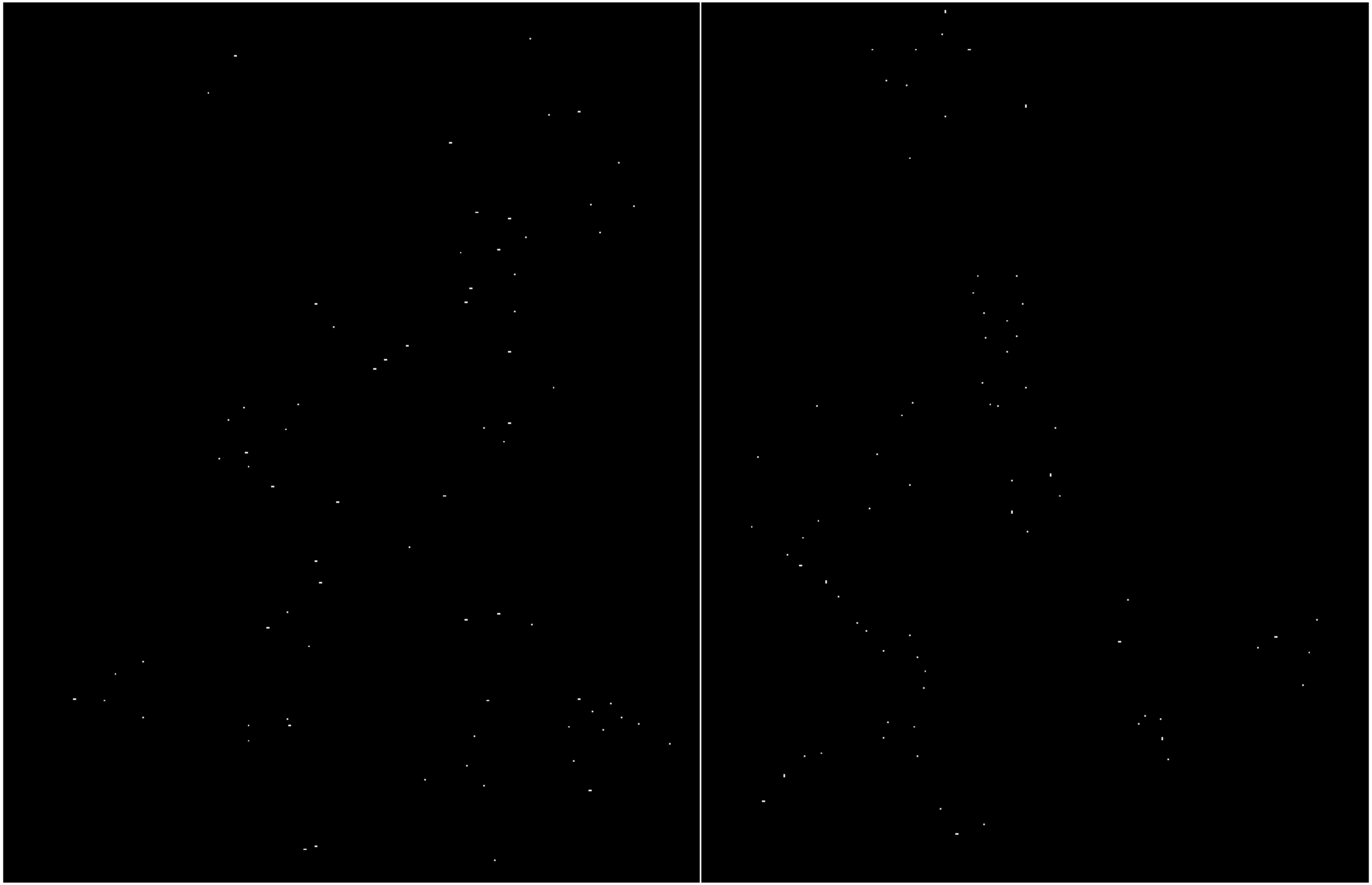
# Harris Detector: Steps

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



# Harris Detector: Steps

Take only the points of local maxima of $R$

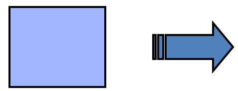


# Harris Detector: Results

# Invariance and covariance

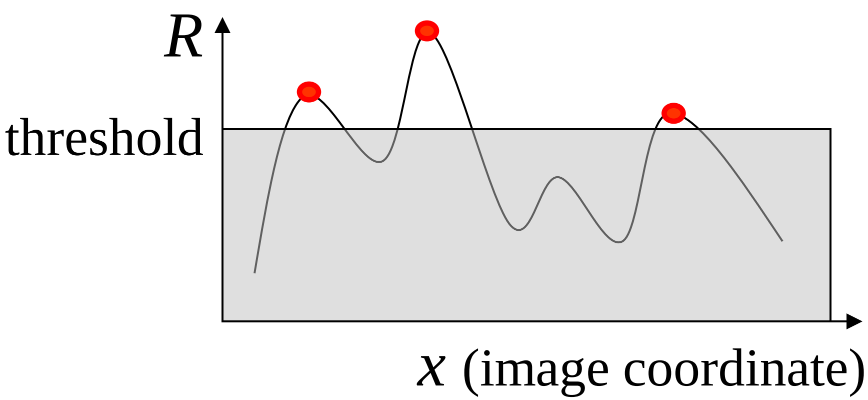
- We want corner locations to be *invariant* to photometric transformations and *covariant* to geometric transformations
  - **Invariance:** image is transformed and corner locations do not change
  - **Covariance:** if we have two transformed versions of the same image, features should be detected in corresponding locations

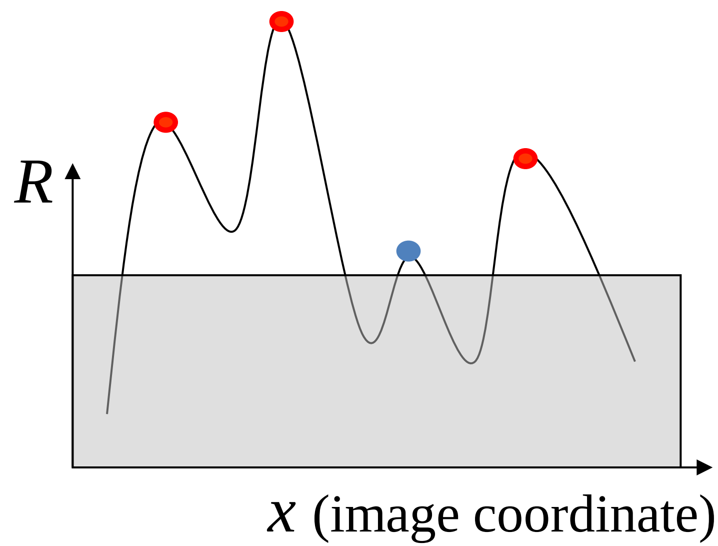
# Harris Detector: Affine Intensity Change



$I \rightarrow a\,I + b$

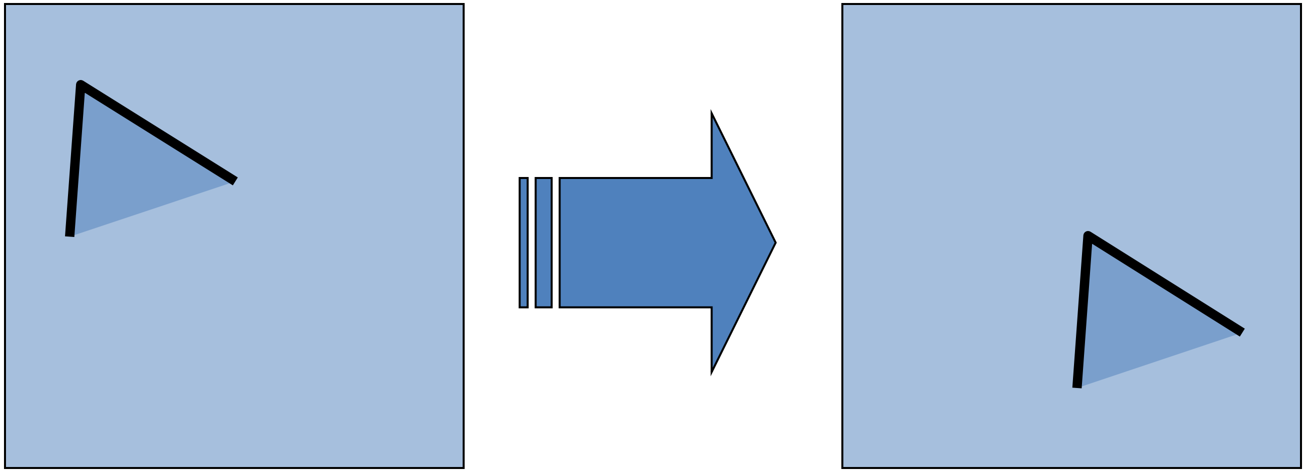
- ✓ Only derivatives are used => invariance to intensity shift $I \rightarrow I + b$
- ✓ Intensity scale: $I \rightarrow a\,I$




*Partially invariant* to affine intensity change

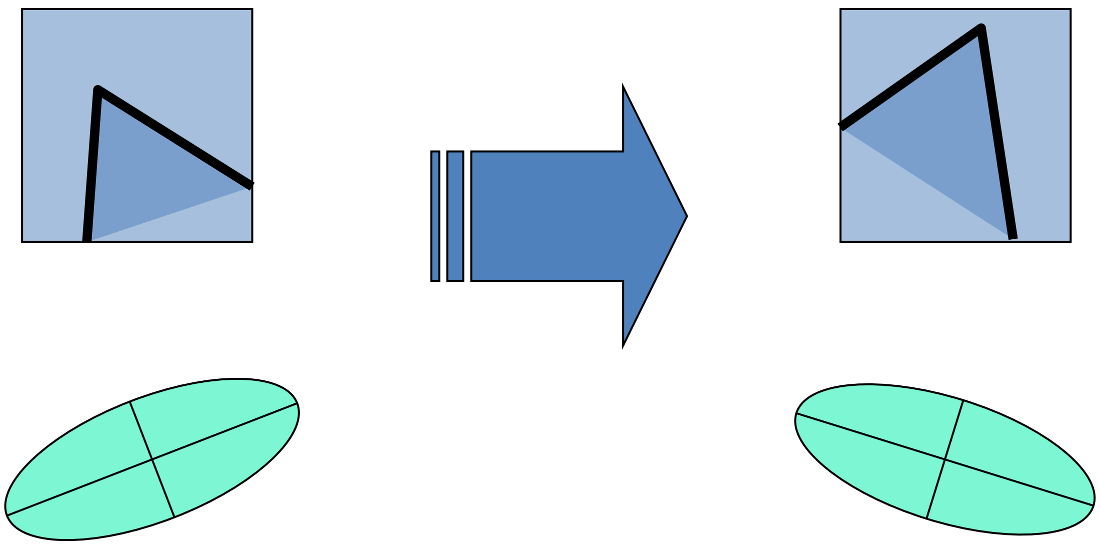
# Harris Detector: Translation



- Derivatives and window function are shift-invariant

Corner location is covariant to image translation

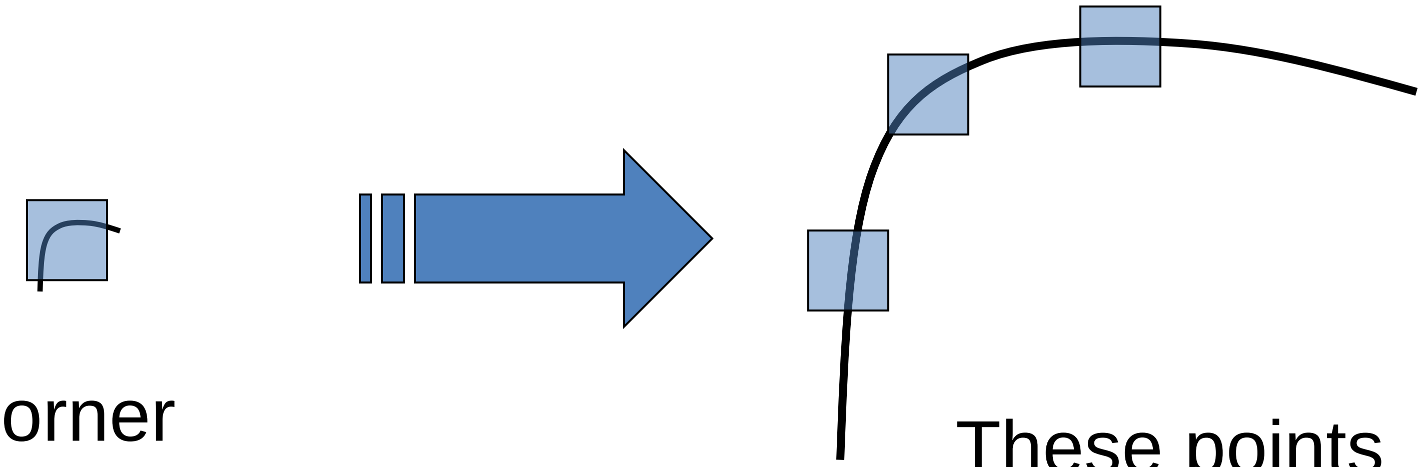
# Harris Detector: Rotation



$2^{nd}$ moment Ellipse rotates but its shape (i.e. eigenvalues) remains the same

*Corner response R* is covariant to image rotation

# Harris Detector: Scale Change

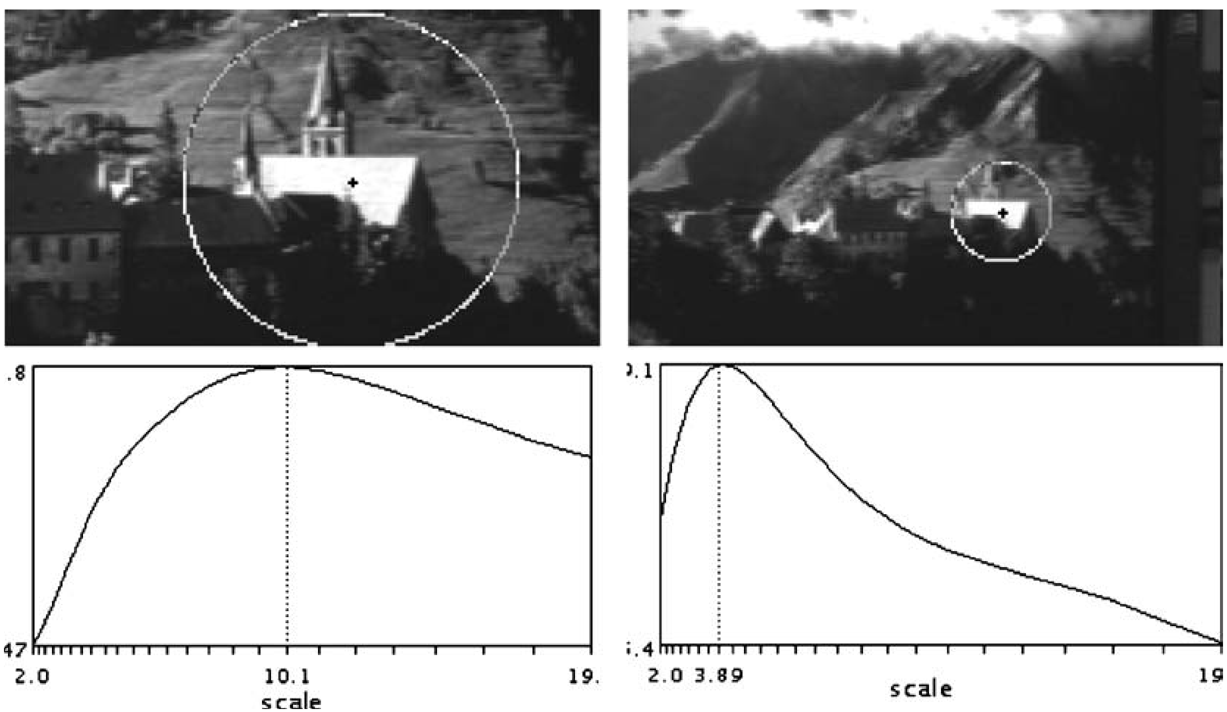



These points will be classified as edges

*Not covariant* to scaling!

# Scale-invariant feature detection

- Goal: independently detect corresponding regions in scaled versions of the same image
- Need *scale selection* mechanism for finding characteristic region size that is *covariant* with the image transformation



# Next Time - Scale-invariant features: Blobs / Image Regions