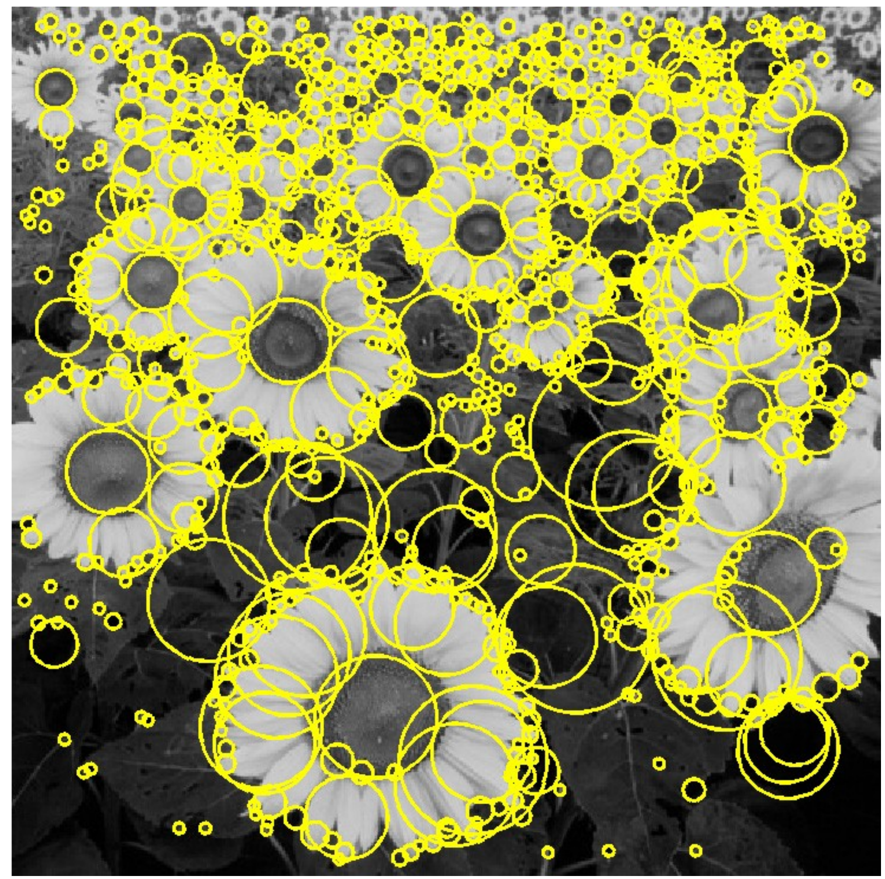
Corner Detection

Feature extraction: Corners and blobs

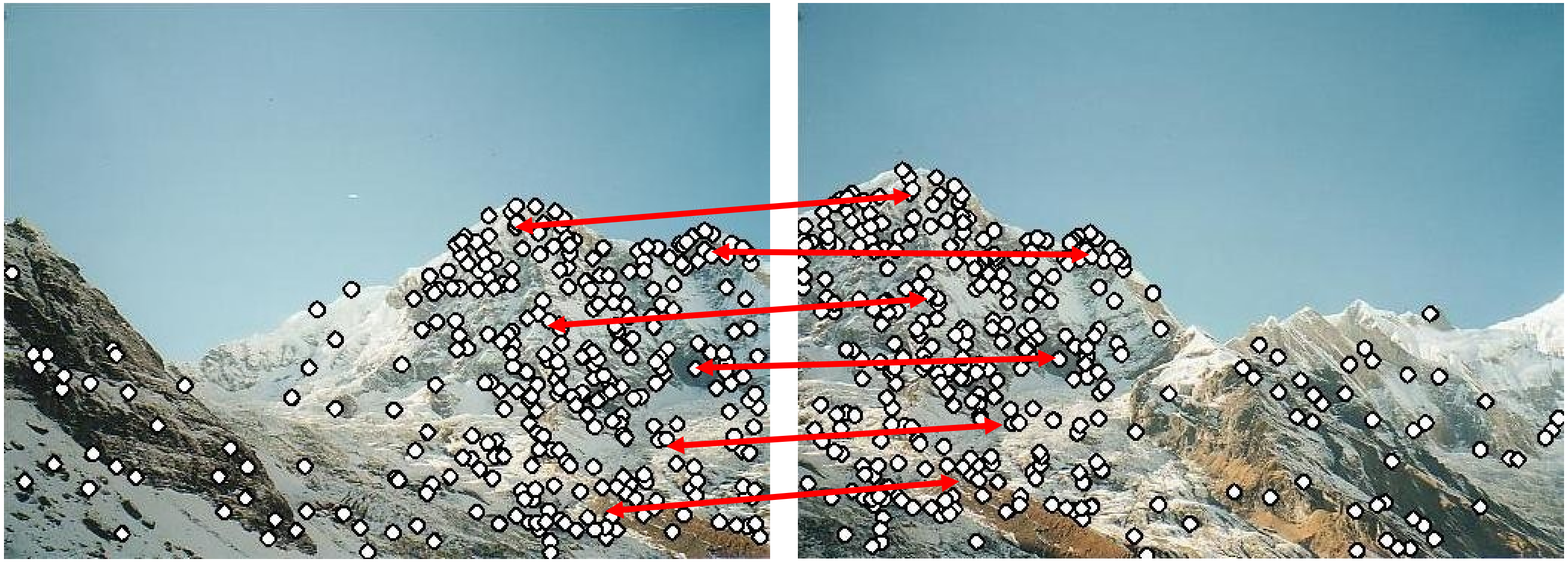


Features

- Motivation: panorama stitching
  - We have two images – how do we combine them?

Feature Extraction

- Motivation: panorama stitching



Step 1: extract features

Step 2: match features

Feature Matching

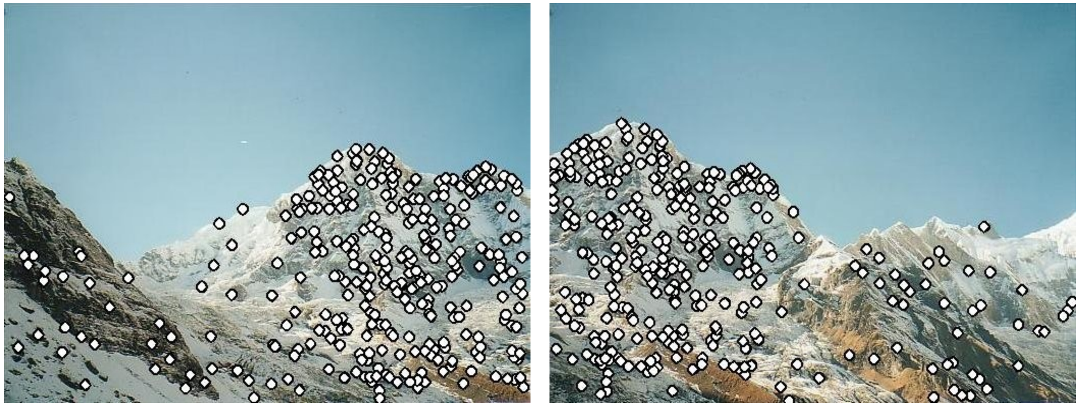
- Motivation: panorama stitching

Step 1: extract features

Step 2: match features

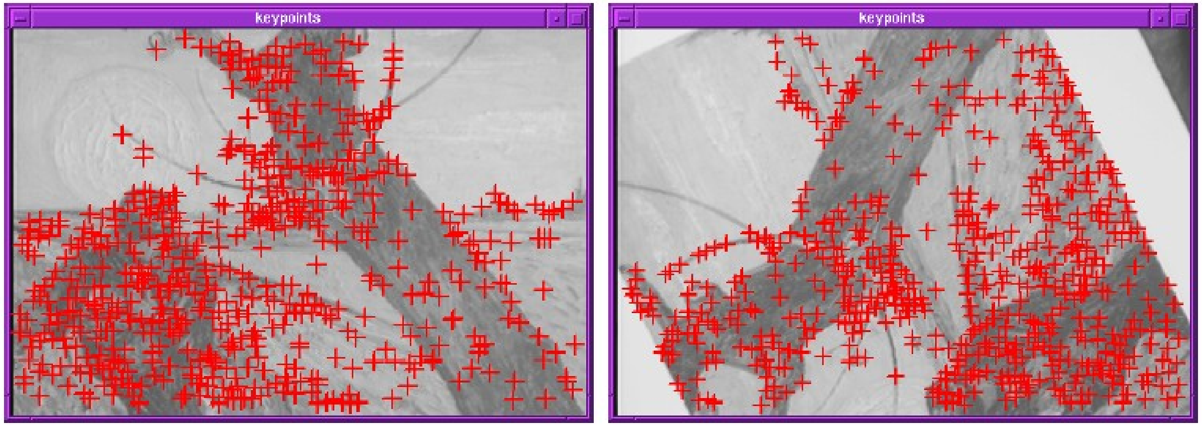
Step 3: align images

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

Finding Corners

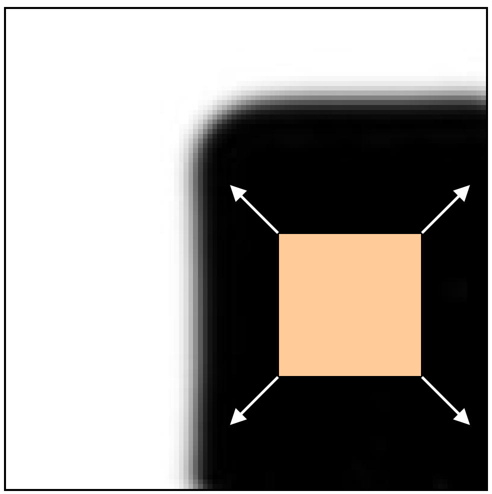


- Key property: in the region around a corner, image gradient has two or more dominant directions
- Corners are repeatable and distinctive

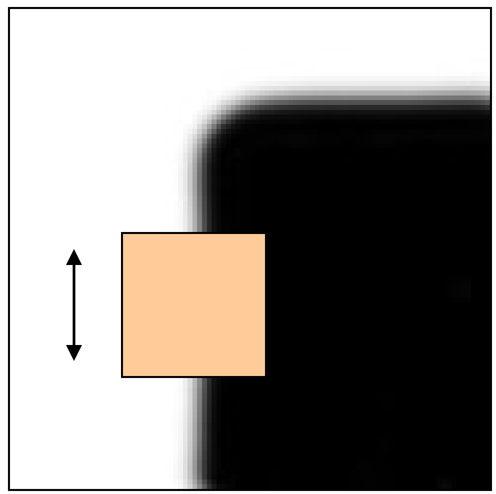
C.Harris and M.Stephens. ["A Combined Corner and Edge Detector."]()
Proceedings of the 4th Alvey Vision Conference: pages 147—151,1988

Concept

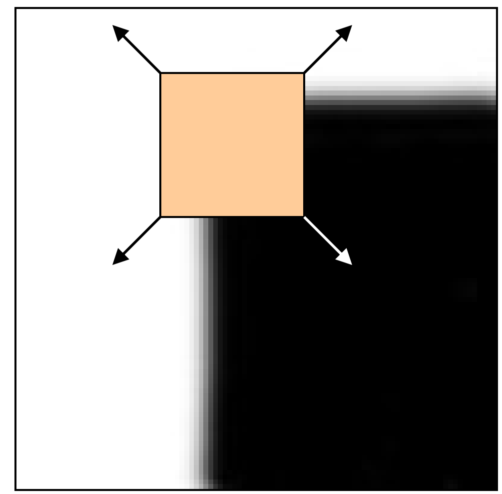
- We should easily recognize the point by looking through a small window
- Shifting a window in *any direction* should give *a large change* in intensity



"flat" region:
no change in all directions



"edge":
no change along the edge direction



"corner":
significant change in all directions

Harris Detector: Mathematics

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

Sum of squared difference (SSD function)

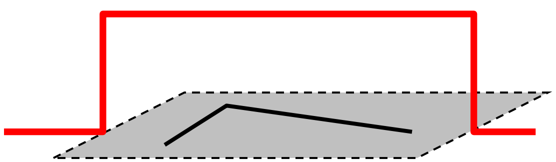
$$E(u,v) = \sum_{x,y} w(x,y)\left[I(x+u,y+v) - I(x,y)\right]^2$$

Window function

Shifted intensity

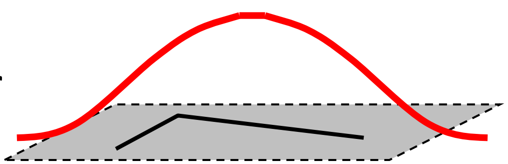
Intensity

Window function $w(x,y) =$



1 in window, 0 outside

or



Gaussian

Harris Detector: Mathematics

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

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

By Taylor expansion of *I(x+u,y+v)* :

$$I(x+u,y+v) \approx I(x,y) + I_u(x,y)u + I_v(x,y)v$$

$$E(u,v) \approx \sum_{x,y} w(x,y)(I_u(x,y)u + I_v(x,y)v)^2$$

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 2×2 matrix computed from image derivatives:

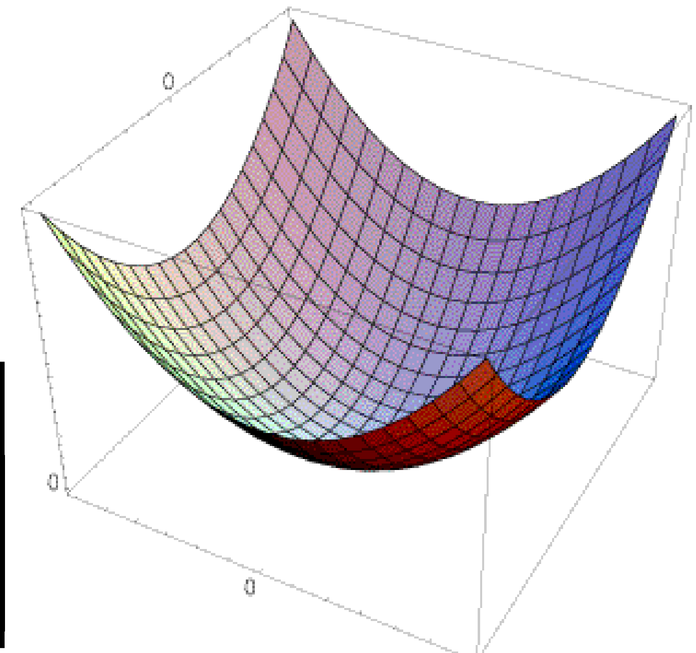
$$\boxed{M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}}$$

$$M = \begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} = \sum \begin{bmatrix} I_x \\ I_y \end{bmatrix} [I_x \; I_y] = \sum \nabla I (\nabla I)^T$$

Interpreting the second moment matrix

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

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



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

Interpreting the second moment matrix

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

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

In a small region we can omit *w*.

$$M = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} = R^{-1} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R$$

Filter

First-order derivative filters

$$[1 \quad -1] \longrightarrow \frac{\partial f}{\partial x} \approx f(x+1, y) - f(x, y) \qquad \begin{bmatrix} 1 \\ -1 \end{bmatrix} \longrightarrow \frac{\partial f}{\partial y} \approx f(x, y+1) - f(x, y)$$

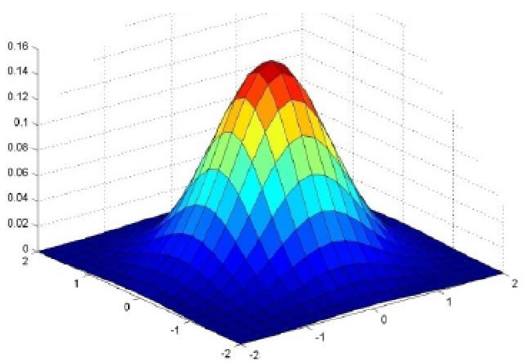
Second-order derivative filters

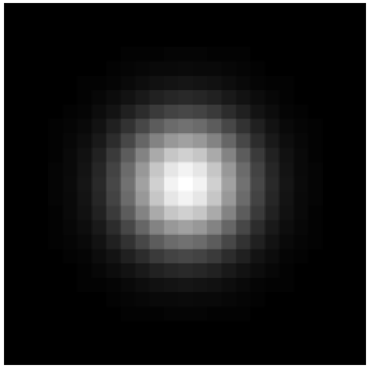
$$[1 \quad -2 \quad 1] \longrightarrow \frac{\partial^2 f}{\partial x^2} \approx f(x+1, y) - 2f(x, y) + f(x-1, y)$$

$$\begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} \longrightarrow \frac{\partial^2 f}{\partial y^2} \approx f(x, y+1) - 2f(x, y) + f(x, y-1)$$

Laplacian: $$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \Rightarrow [1 \quad -2 \quad 1] + \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Gaussian filter





0.003	0.013	0.022	0.013	0.003
0.013	0.059	0.097	0.059	0.013
0.022	0.097	0.159	0.097	0.022
0.013	0.059	0.097	0.059	0.013
0.003	0.013	0.022	0.013	0.003

5 x 5, σ = 1

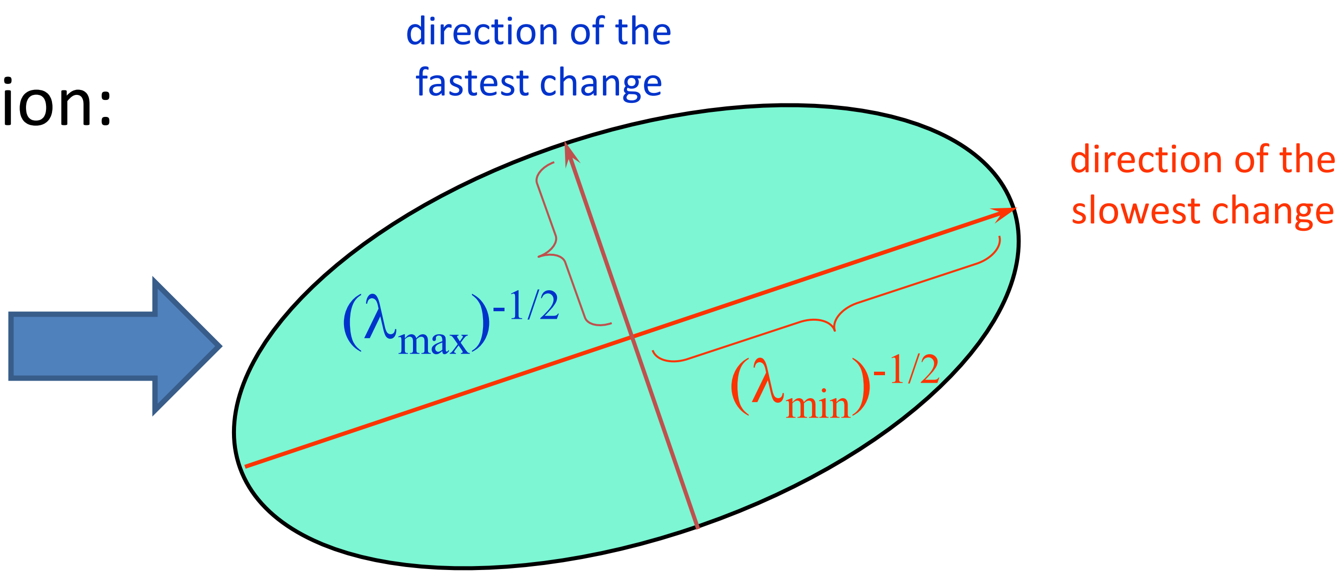
General Case

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*

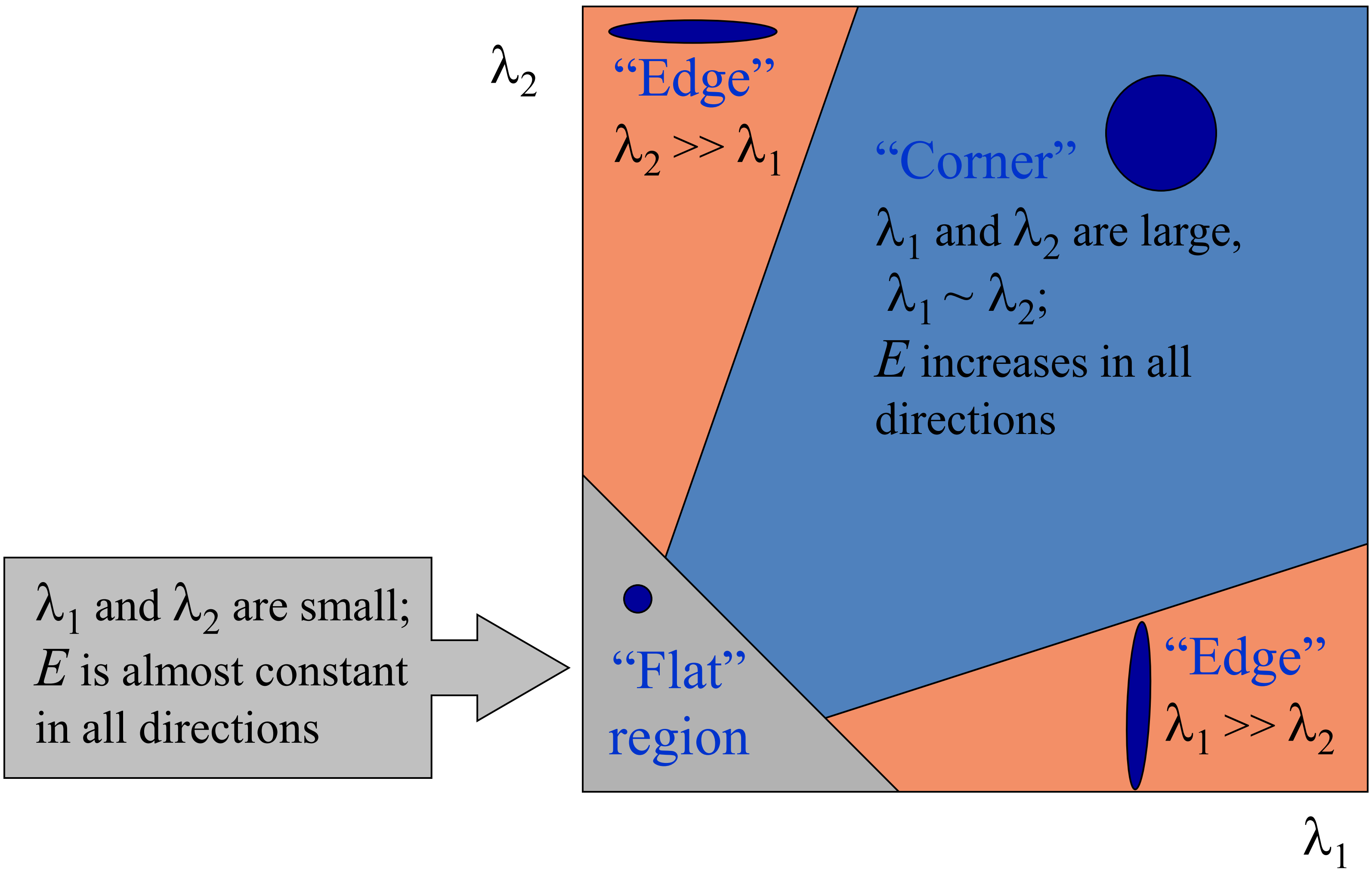
Ellipse equation:

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



Interpreting The Eigenvalues

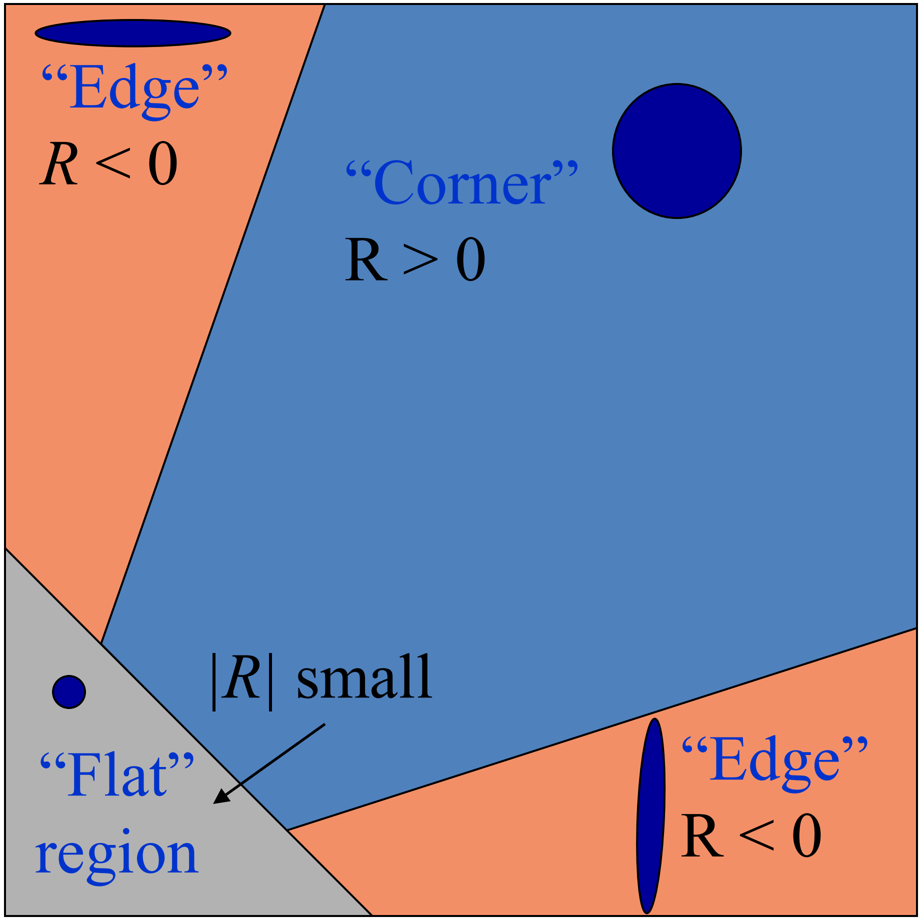
Classification of image points using eigenvalues of *M*:



Corner Response Function

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

α: constant (0.04 to 0.06)



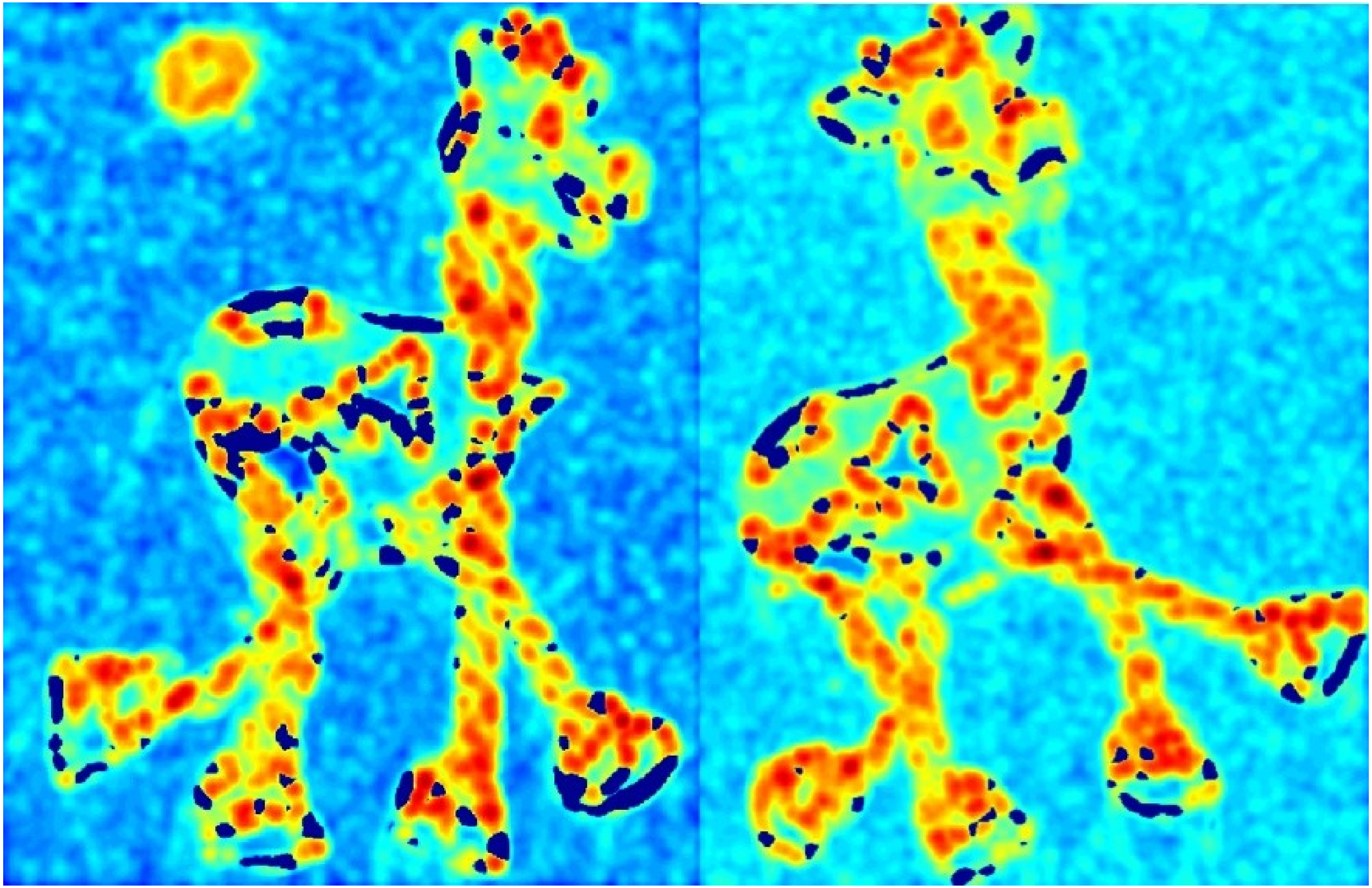
Harris Detector Algorithm

- Filter image with Gaussian to reduce noise
- Compute magnitude of the x and y gradients at each pixel
- Construct M in a window around each pixel (Harris uses a Gaussian window)
- Compute λs of M
- Compute $R = \det M - k\left(\operatorname{trace} M\right)^2$
- If R> T a corner is detected; retain point of local maxima

Harris Detector: Steps

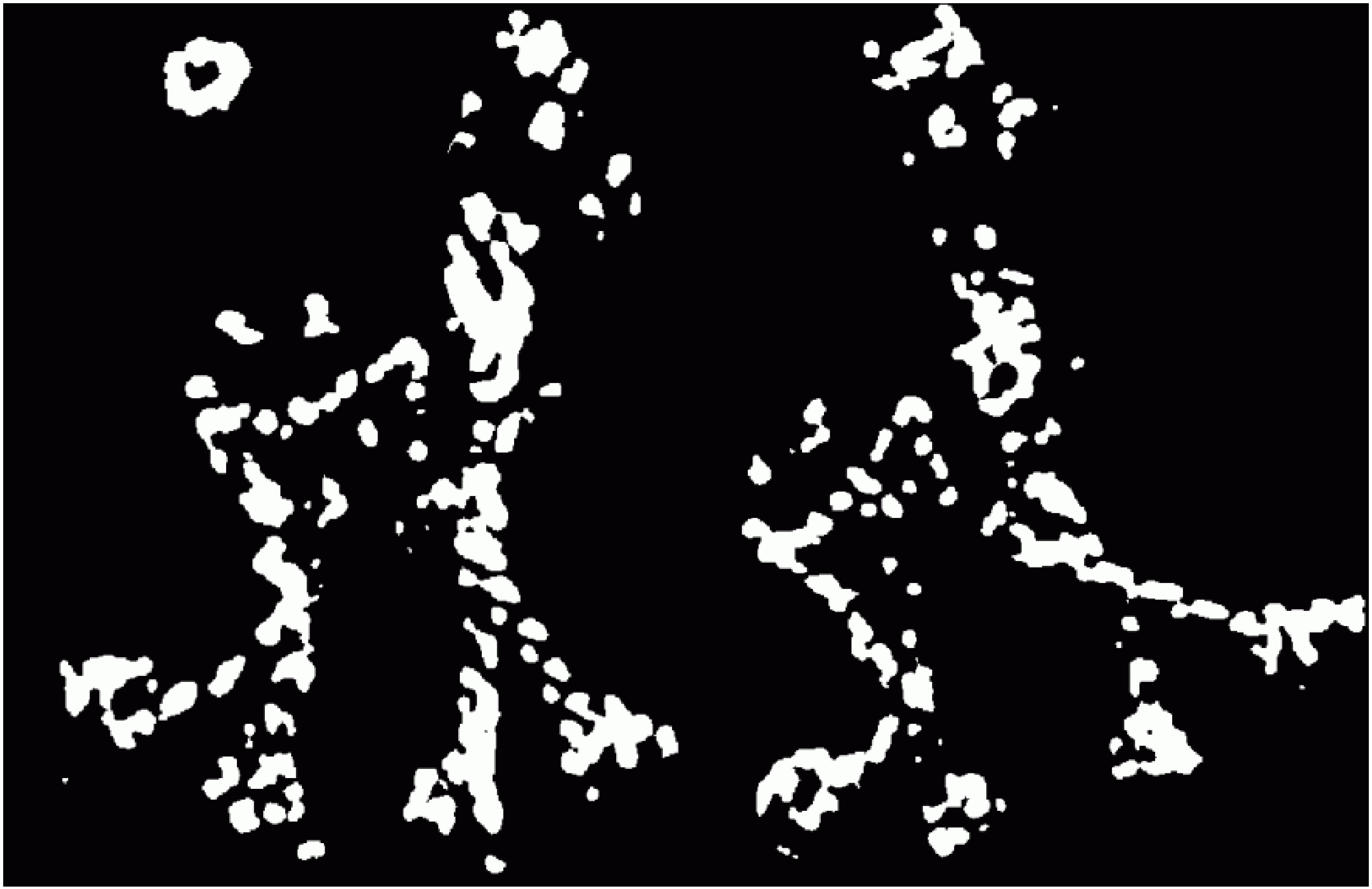
Harris Detector: Steps

Compute corner response R



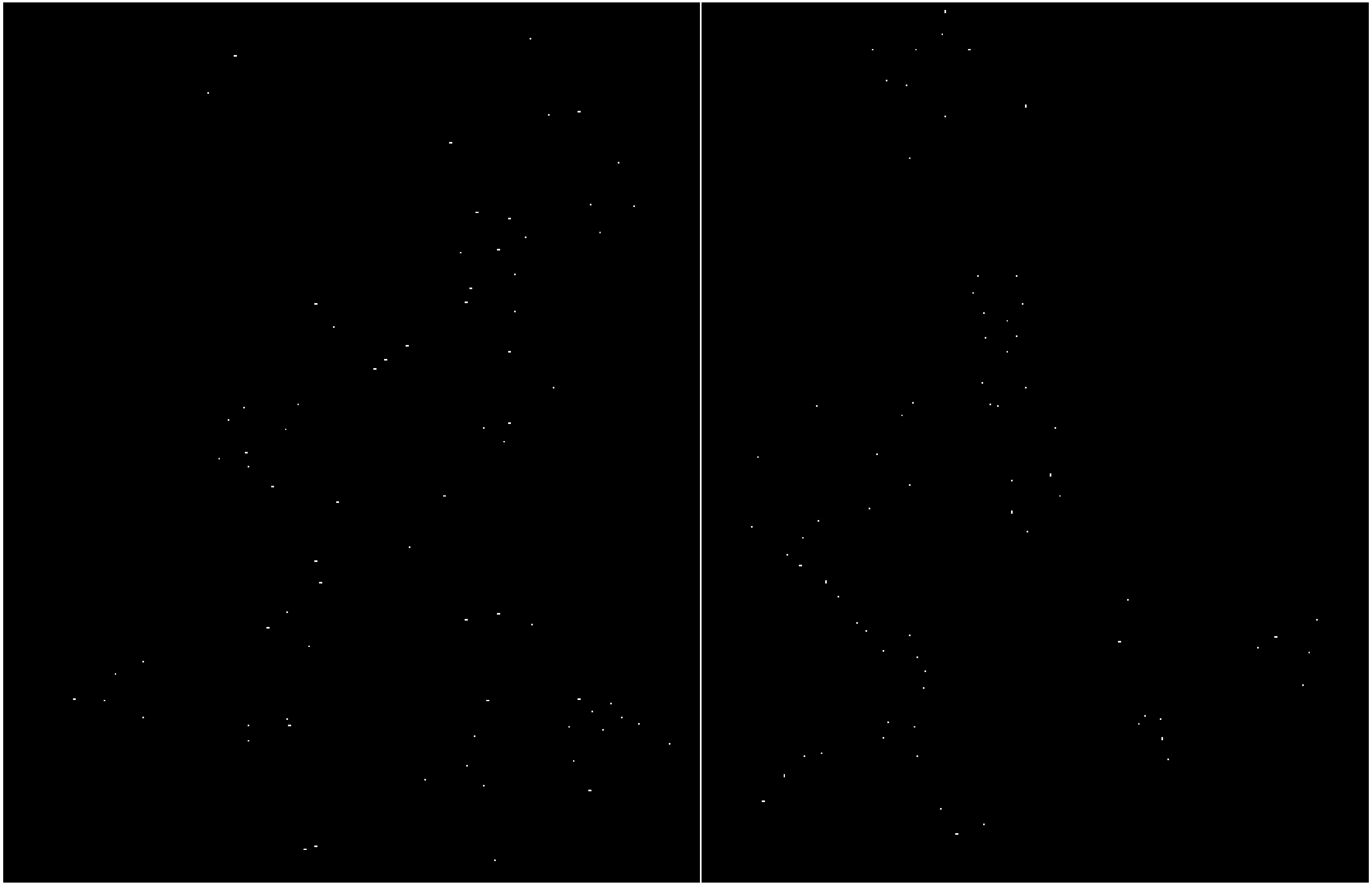
Harris Detector: Steps

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



Harris Detector: Steps

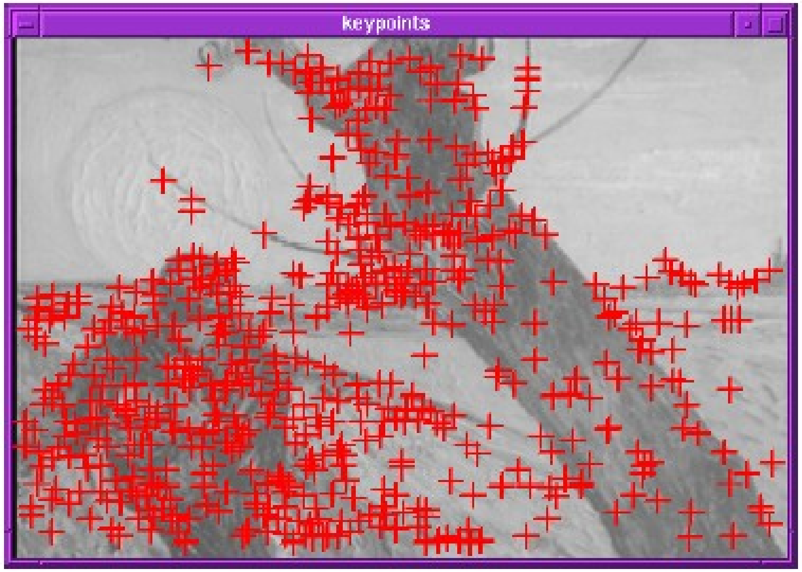
Take only the points of local maxima of R

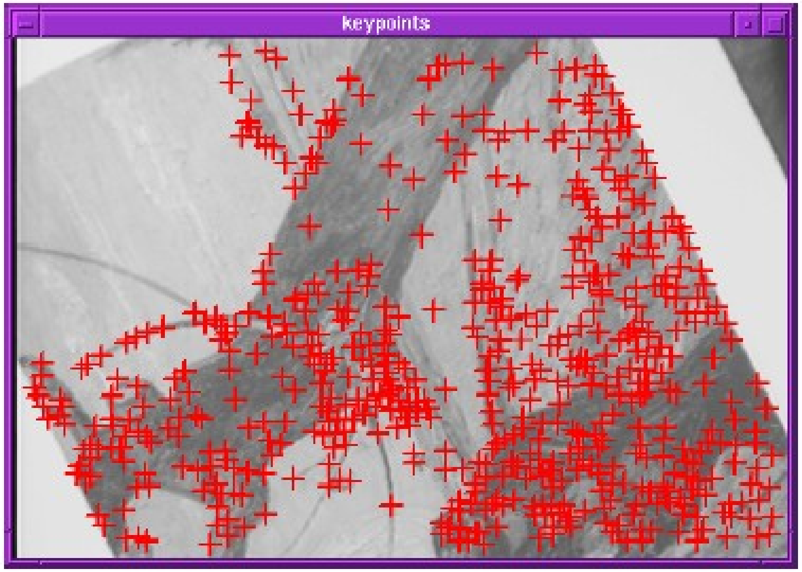


Harris Detector: Steps

Invariance

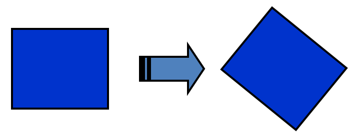
- We want features to be detected despite geometric or photometric changes in the image: if we have two transformed versions of the same image, features should be detected in corresponding locations.

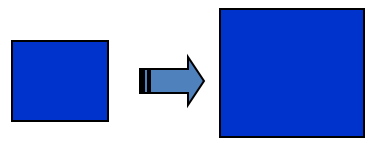




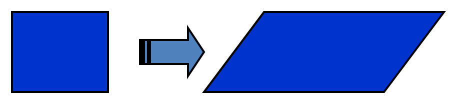
Models of Image Change

- Geometric
  - **Rotation**



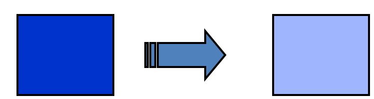


  - **Scale**



  - **Affine**
    valid for: orthographic camera, locally planar object

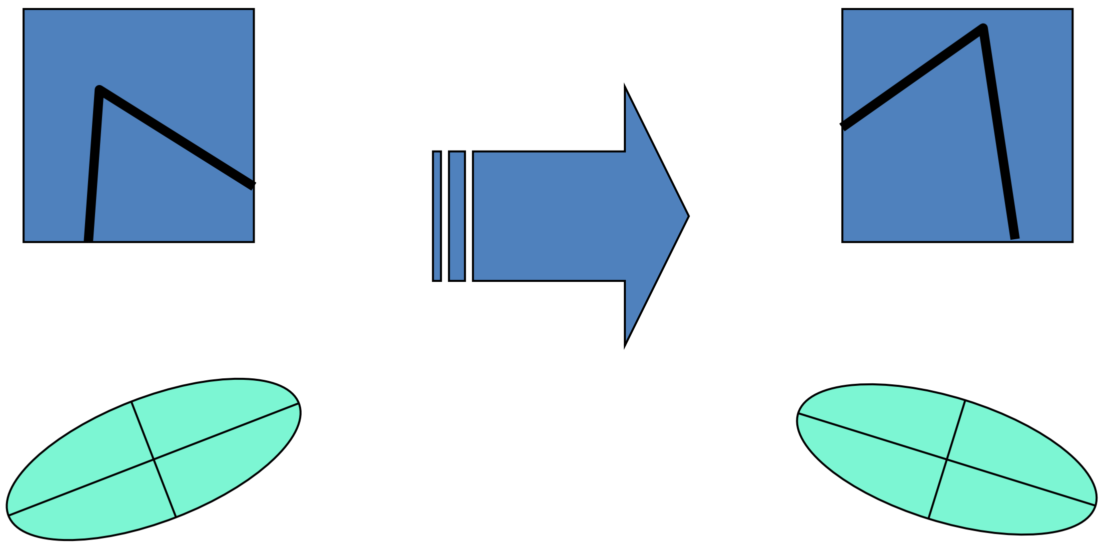
- Photometric



  - **Affine intensity change** ($I \rightarrow a\, I + b$)

Harris Detector: Invariance Properties

- Rotation



Ellipse rotates but its shape (i.e. eigenvalues) remains the same

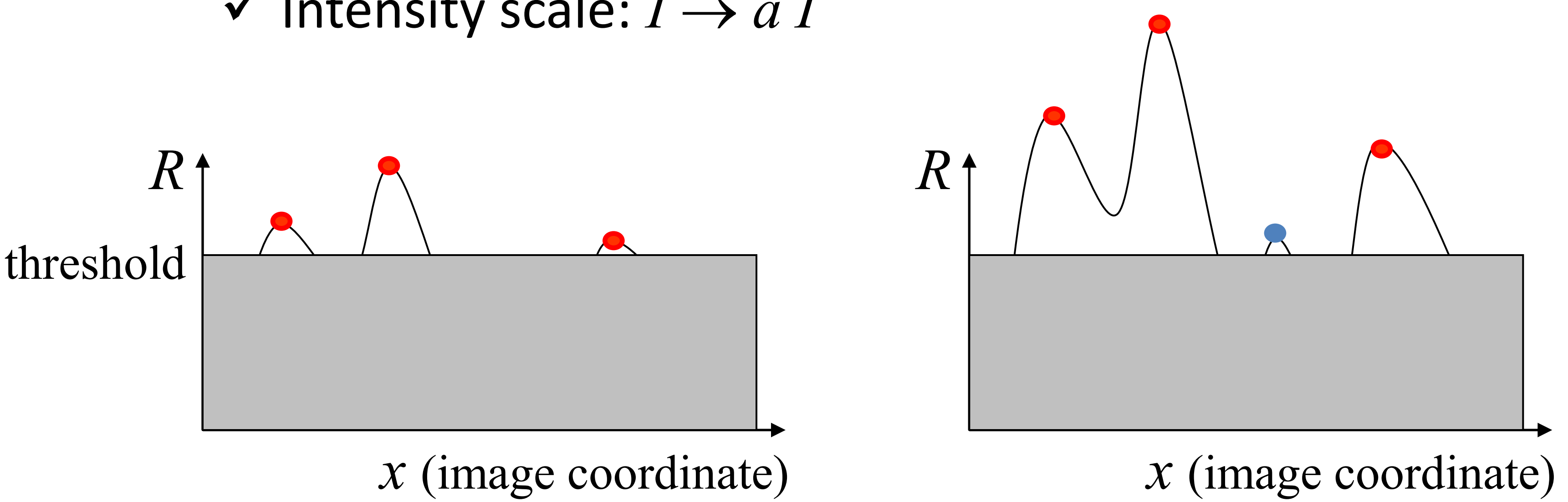
Corner response R is invariant to image rotation

Harris Detector: Invariance Properties

- Affine intensity change

✓ Only derivatives are used => invariance to intensity shift $I \rightarrow I + b$

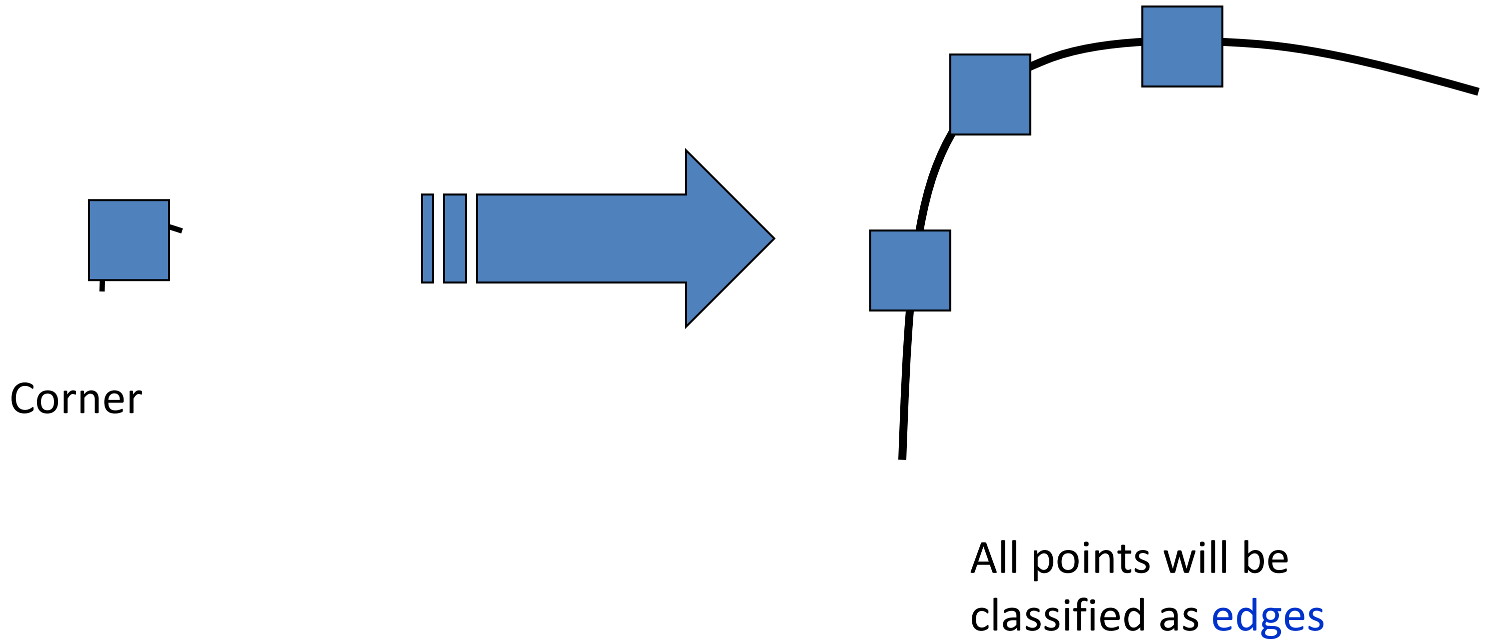
✓ Intensity scale: $I \rightarrow a\,I$



Partially invariant to affine intensity change

Harris Detector: Invariance Properties

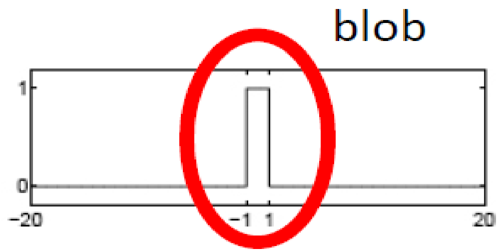
- Scaling

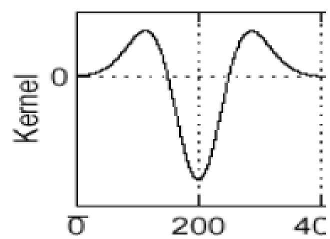


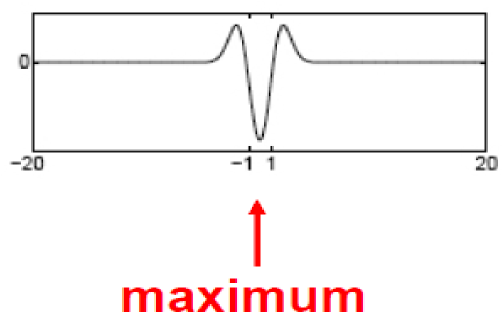
Not invariant to scaling

Blob Detection

- Blob = superposition of nearby edges



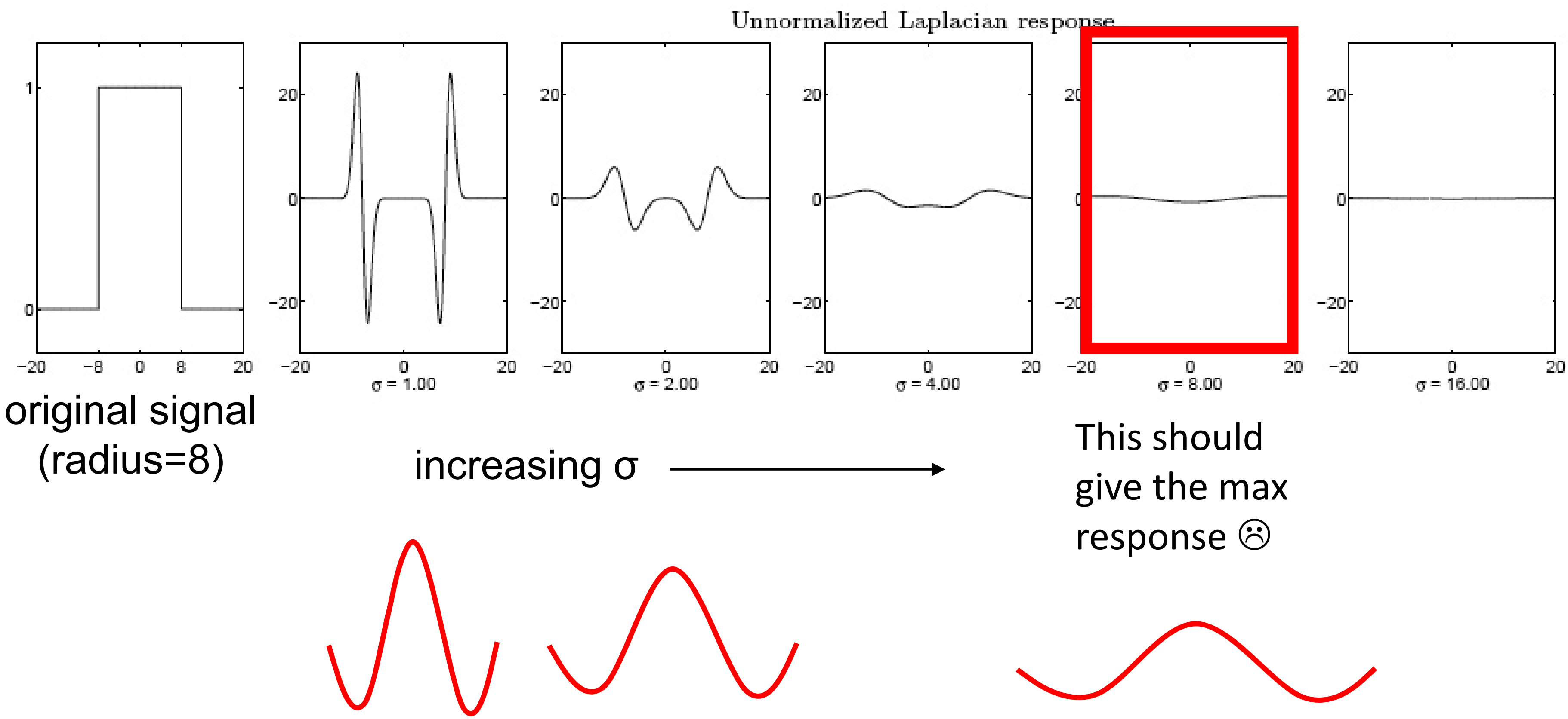




Scale selection

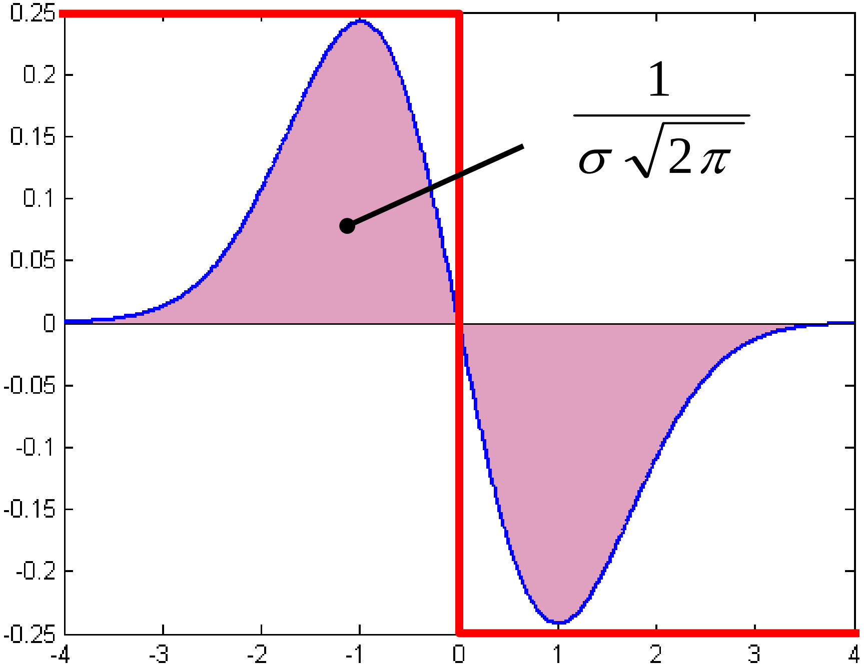
- We want to find the characteristic scale of the blob by convolving it with Laplacians at several scales and looking for the maximum response

- However, Laplacian response decays as scale increases:



Scale normalization

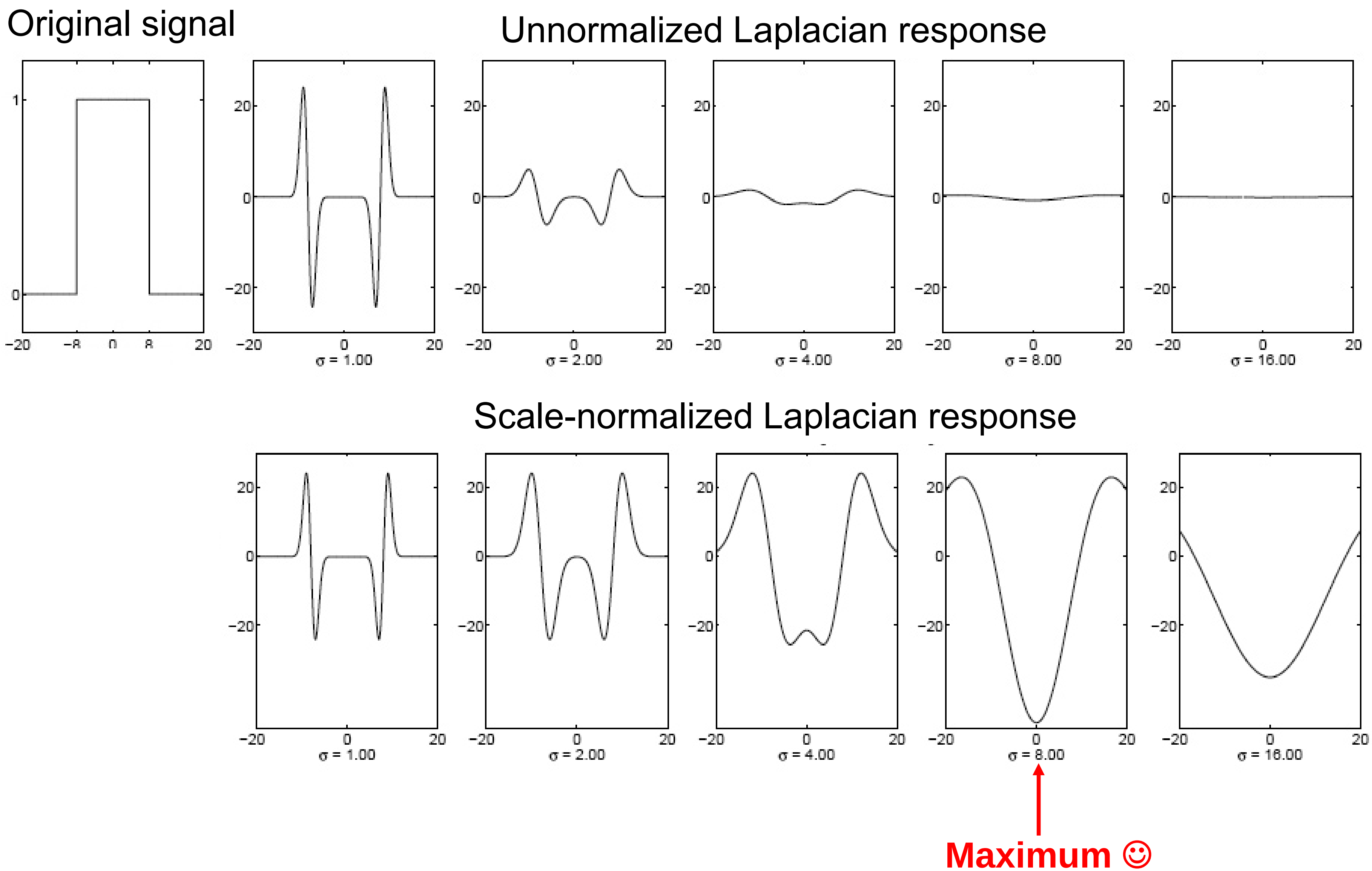
- The response of a derivative of Gaussian filter to a perfect step edge decreases as σ increases



Scale normalization

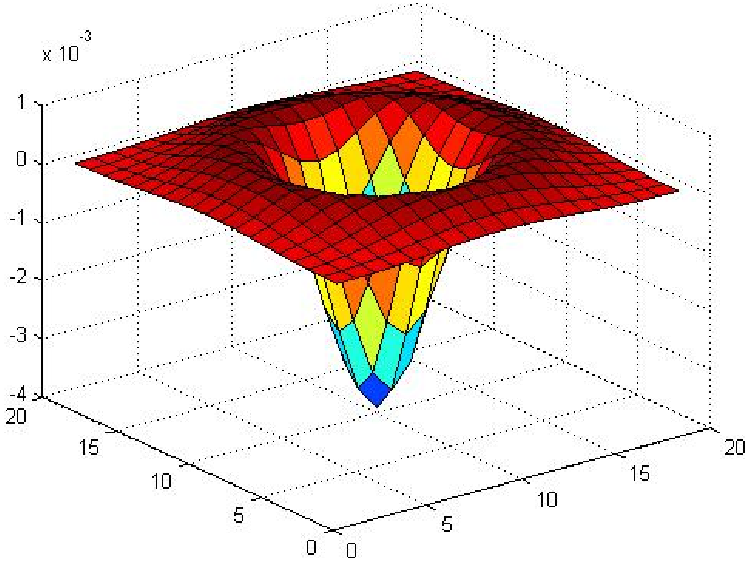
- To keep response the same (scale-invariant), must multiply Gaussian derivative by σ
- Laplacian is the second Gaussian derivative, so it must be multiplied by σ^2

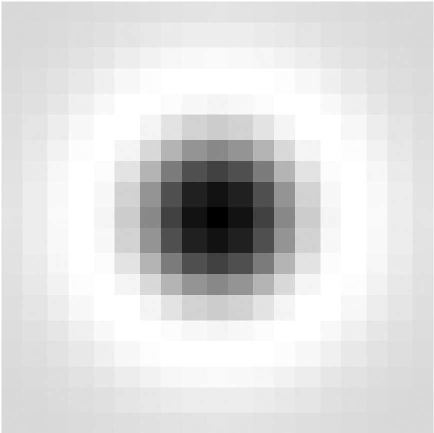
Effect of scale normalization



Blob detection in 2D

- Laplacian of Gaussian: Circularly symmetric operator for blob detection in 2D

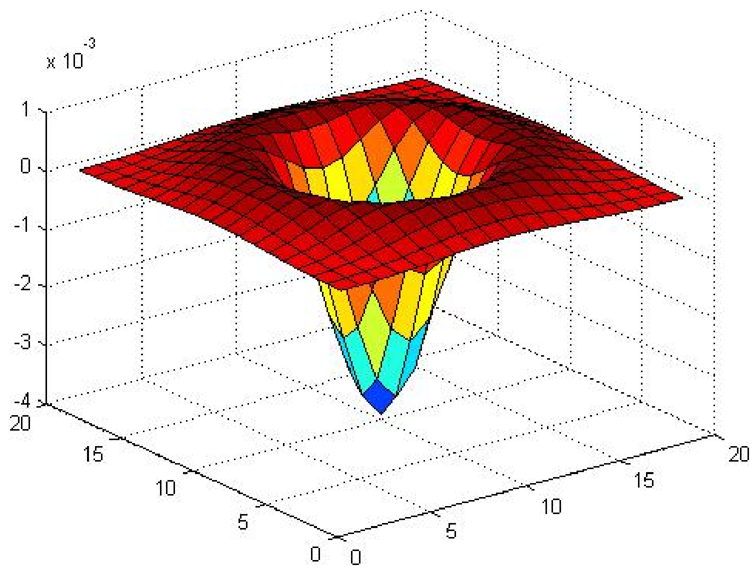


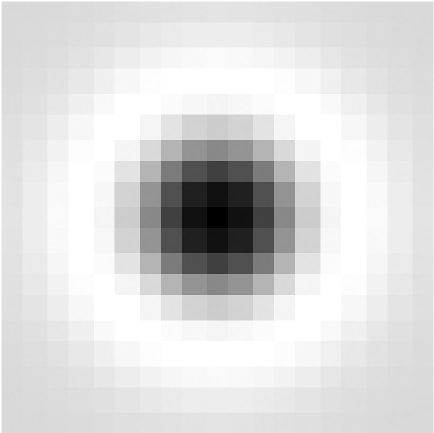


$$\nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}$$

Blob detection in 2D

- Laplacian of Gaussian: Circularly symmetric operator for blob detection in 2D

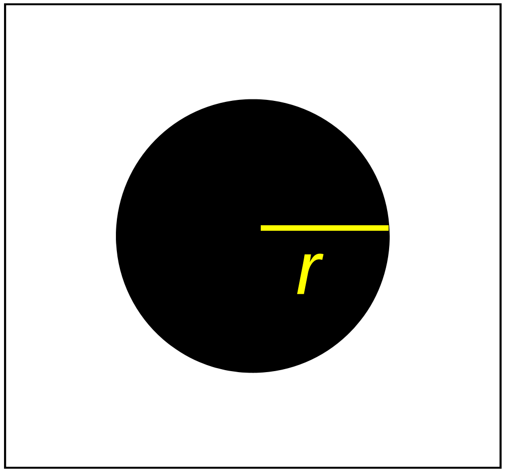




Scale-normalized: $$\nabla^2_{\text{norm}} g = \sigma^2 \left(\frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2} \right)$$

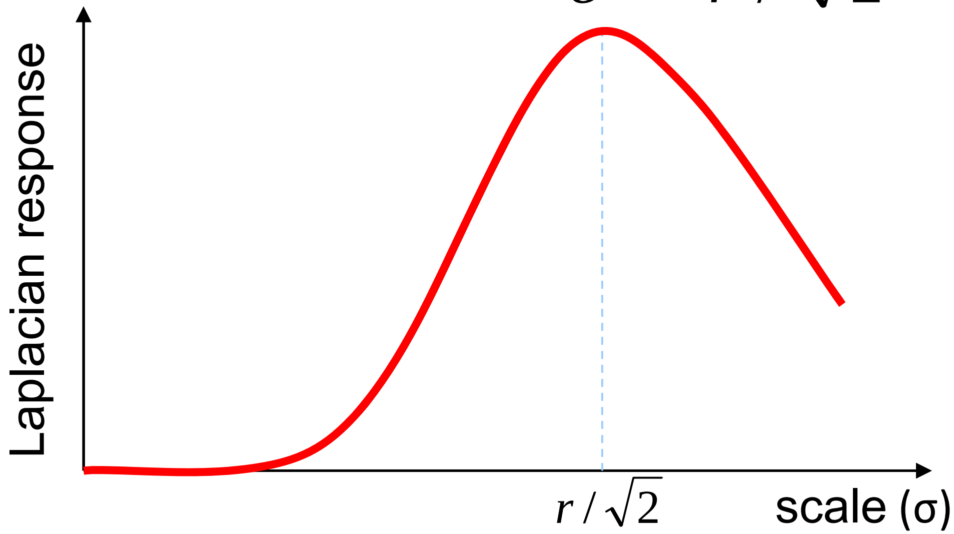
Scale selection

- For a binary circle of radius r, the Laplacian achieves a maximum at



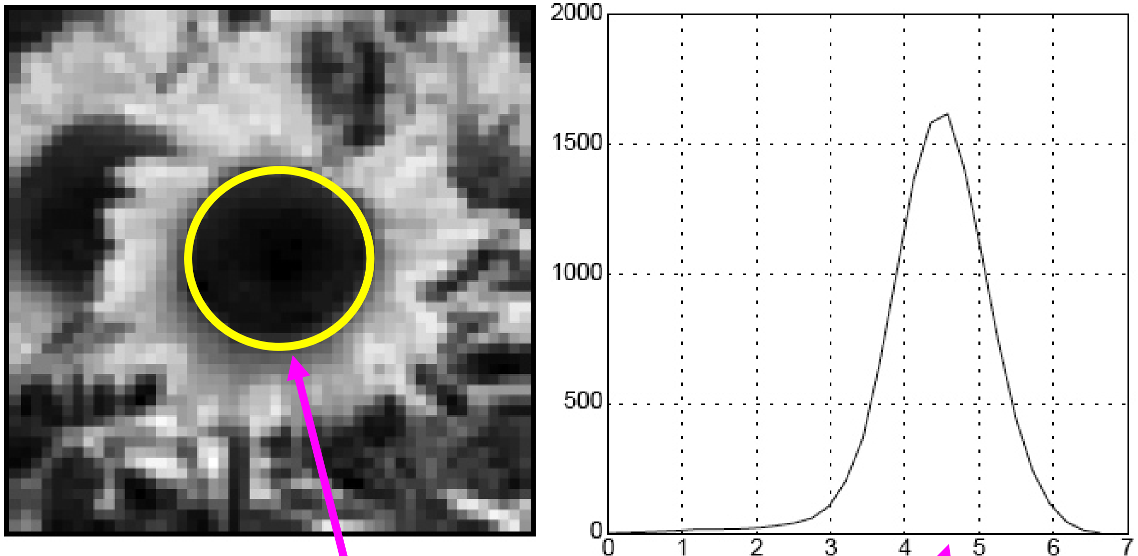
image

$$\sigma = r / \sqrt{2}$$



Characteristic scale

- We define the characteristic scale as the scale that produces peak of Laplacian response

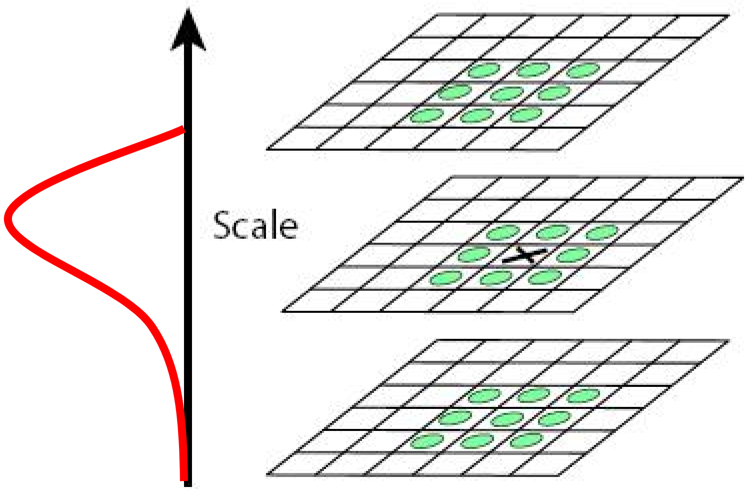


characteristic scale

T. Lindeberg (1998). "Feature detection with automatic scale selection." *International Journal of Computer Vision* **30** (2): pp 77--116.

Scale-space blob detector

1. Convolve image with scale-normalized Laplacian at several scales
2. Find maxima of squared Laplacian response in scale-space
3. This indicate if a blob has been detected
4. And what's its intrinsic scale

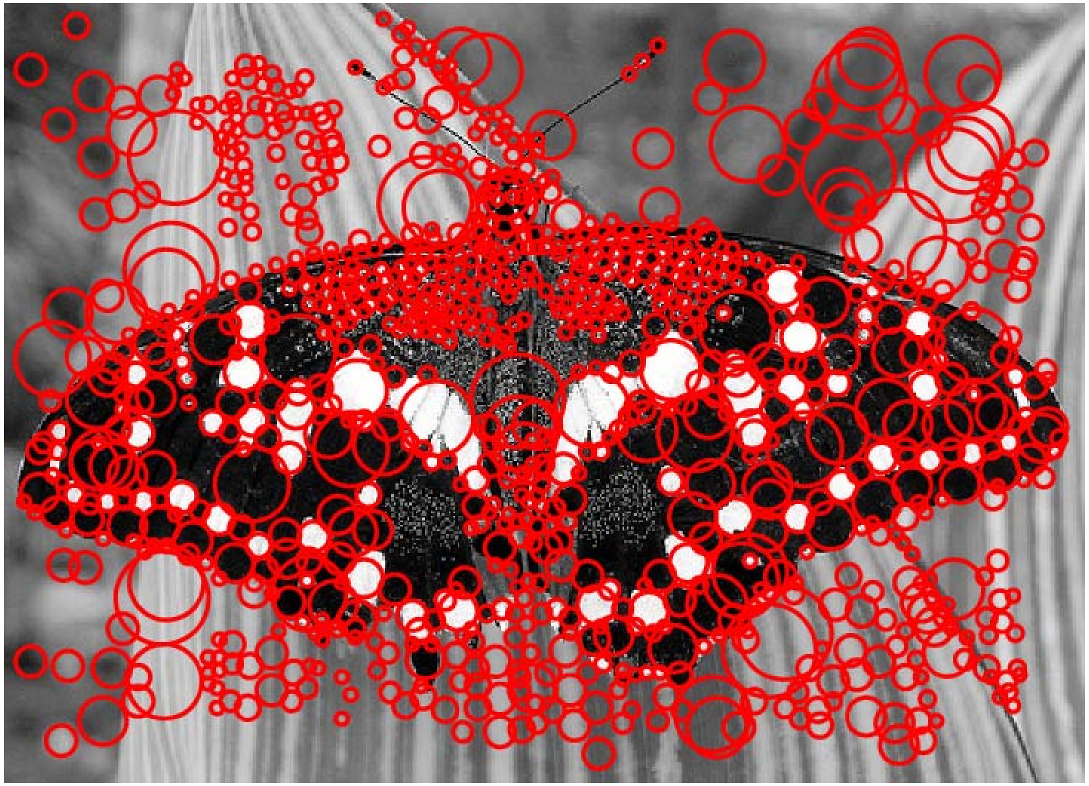


Scale-space blob detector: Example

Scale-space blob detector: Example

sigma = 11.9912

Scale-space blob detector: Example



DOG

David G. Lowe. ["Distinctive image features from scale-invariant keypoints."](#) *IJCV* 60 (2), 04

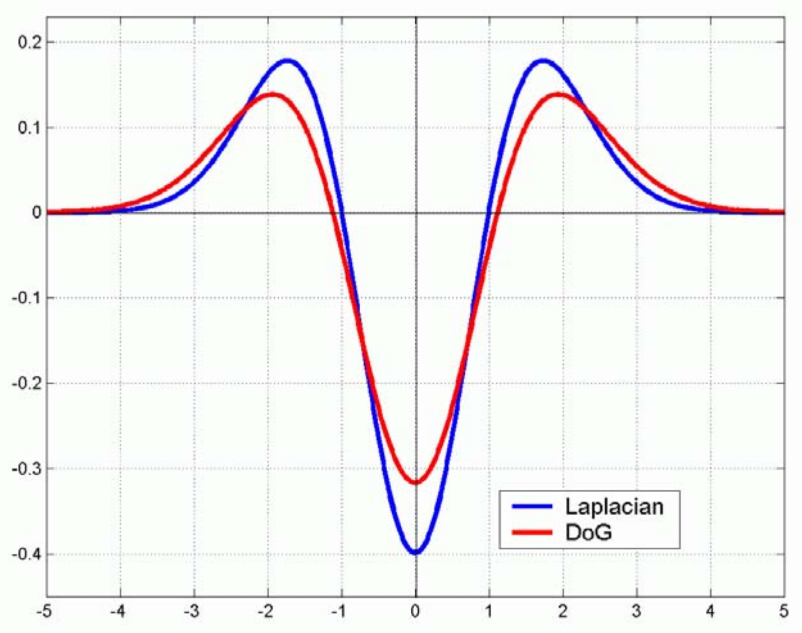
- Approximating the Laplacian with a difference of Gaussians:

$$L = \sigma^2 \left(G_{xx}(x, y, \sigma) + G_{yy}(x, y, \sigma) \right)$$

(Laplacian)

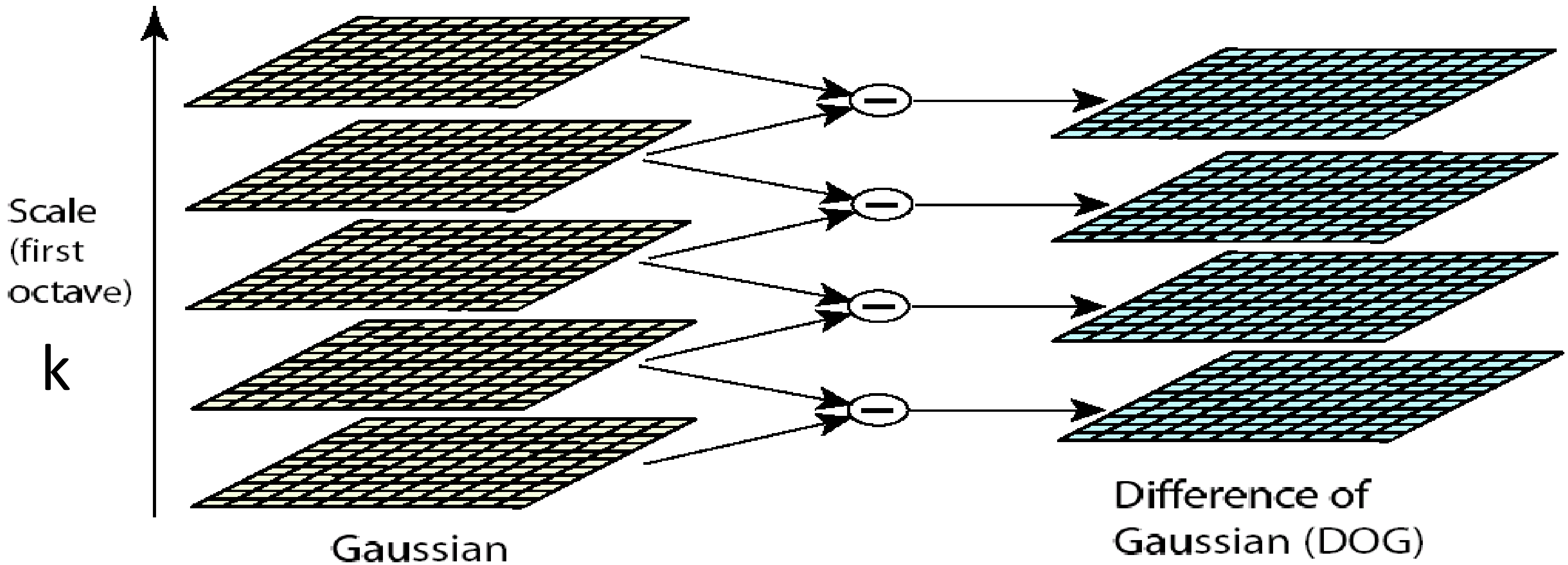
$$DoG = G(x, y, k\sigma) - G(x, y, \sigma)$$

(Difference of Gaussians)
or
Difference of gaussian blurred images at scales k σ and σ



$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k - 1)\sigma^2 \boldsymbol{L}$$

DOG



Output: location, scale, orientation (more later)

Invariance

Detector	Illumination	Rotation	Scale	View point
Harris corner	Yes	Yes	No	No
Lowe '99 (DoG)	Yes	Yes	Yes	No

References

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

[2]T. Lindeberg , "Feature detection with automatic scale selection.", *International Journal of Computer Vision (IJCV)* 30(2): pp 77—116, 1998.

[3]David G. Lowe, "Distinctive image features from scale-invariant keypoints.”, *IJCV* 60(2): pp. 91-110, 2004.

[4] Silvio Savarese, Course slide, Image Enhancement (III) http://www.eecs.umich.edu/~silvio/teaching/EECS556_2009/class_schedule.html