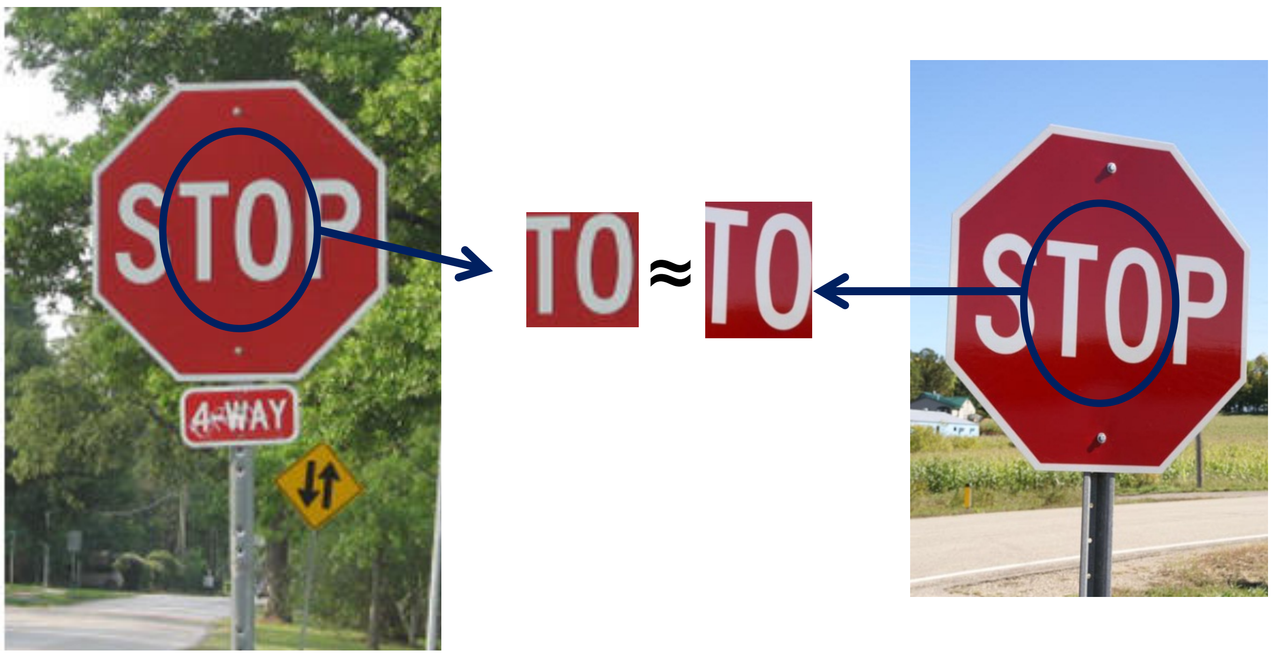
Interest Points and Features

Finding Correspondence

Matching points, patches, edges, or regions



Interest points

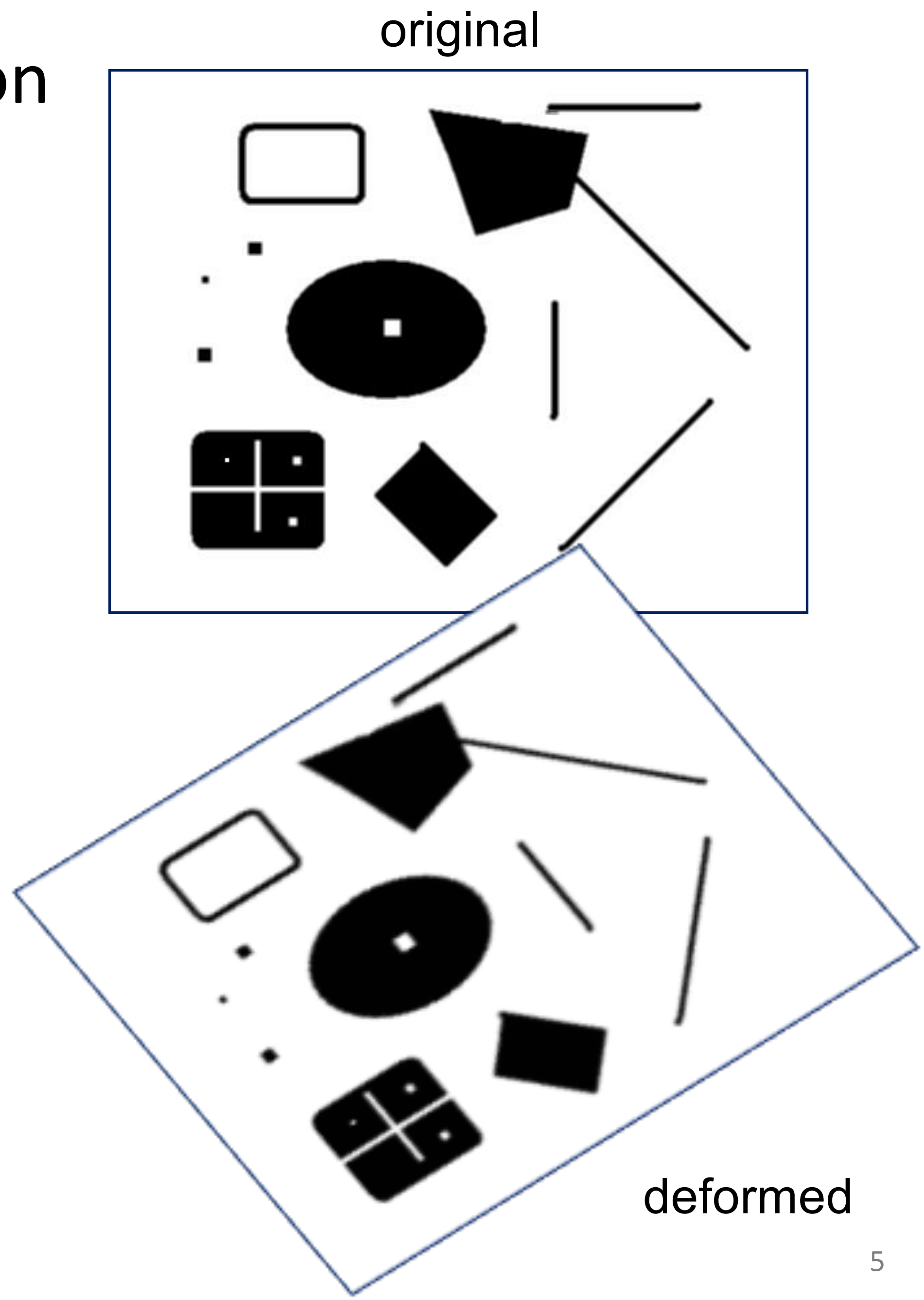
- Note: “interest points” = “keypoints”, also sometimes called “features”

Applications

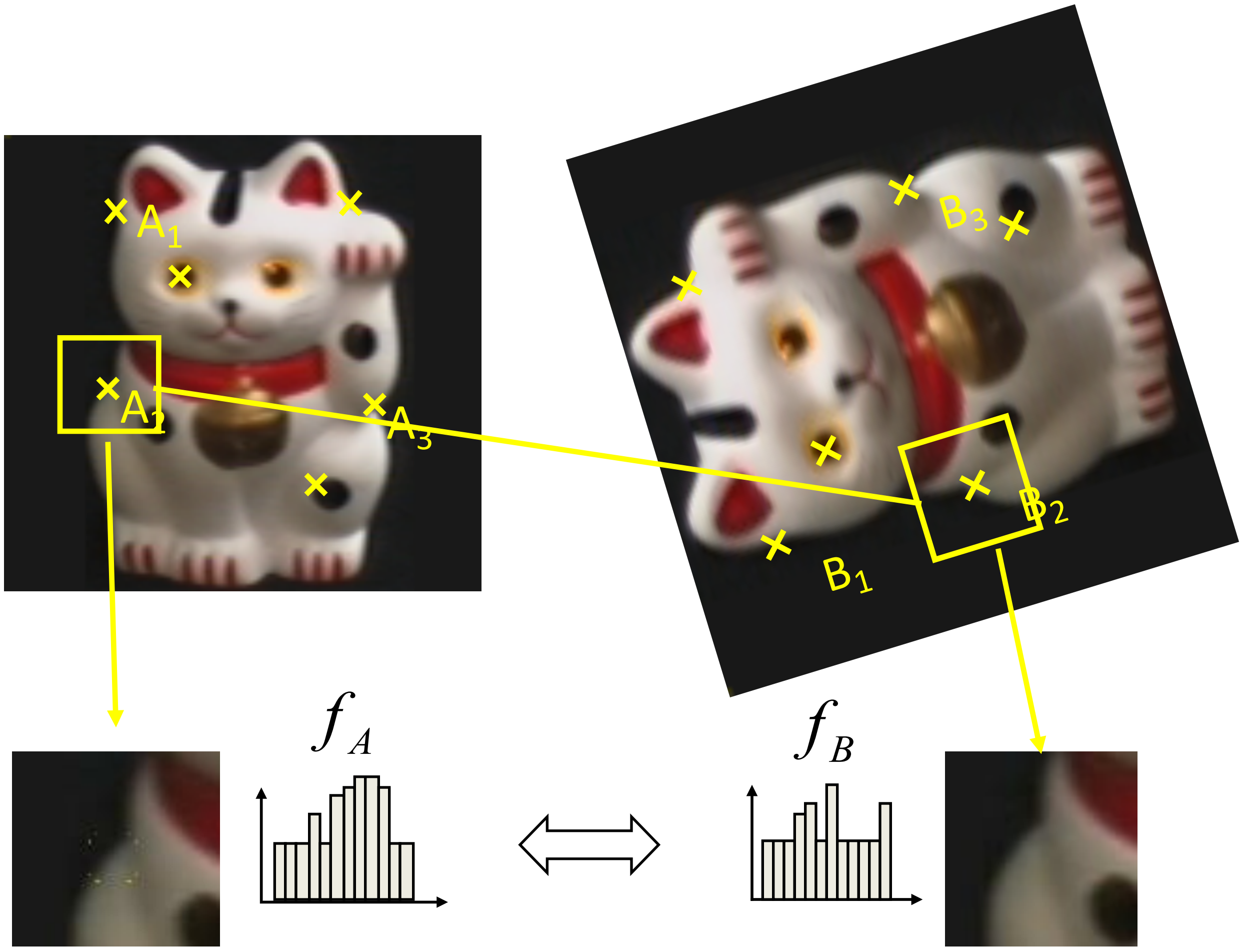
- Keypoints are used for:
  - Image alignment
  - 3D reconstruction
  - Motion tracking
  - Robot navigation
  - Indexing and database retrieval
  - Object recognition

Interest points

- Suppose you have to click on some point, go away and come back after I deform the image, and click on the same points again.
  - Which points would you choose?



Overview of Keypoint Matching



$$f_A \Longleftrightarrow f_B$$

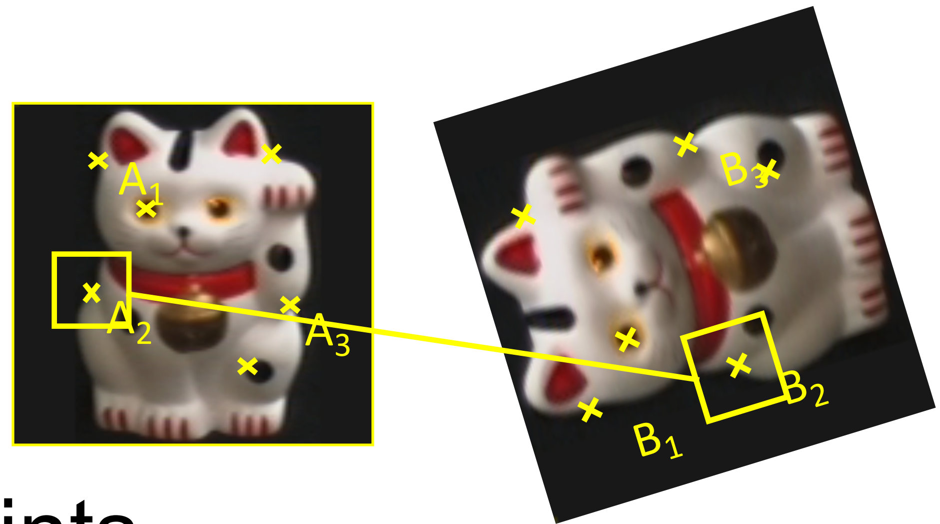
$$d(f_A, f_B) < T$$

1. Find a set of distinctive key-points
2. Define a region around each keypoint
3. Extract and normalize the region content
4. Compute a local descriptor from the normalized region
5. Match local descriptors

Goals for Keypoints

Detect points that are *repeatable* and *distinctive*

Key trade-offs



Detection of interest points

More Repeatable

Robust detection
Precise localization

More Points

Robust to occlusion
Works with less texture

Description of patches

More Distinctive

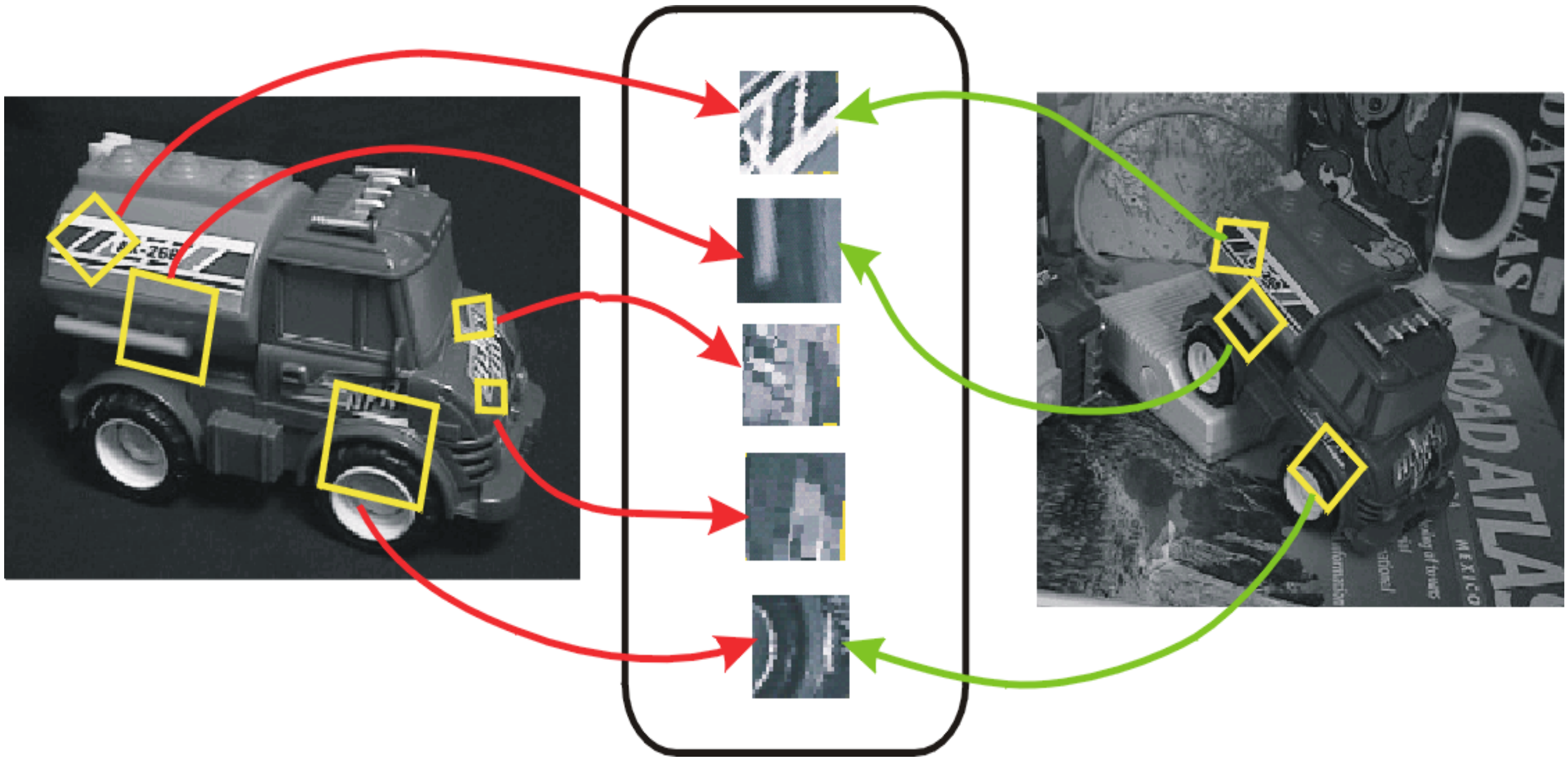
Minimize wrong matches

More Flexible

Robust to expected variations
Maximize correct matches

Invariant Local Features

- Image content is transformed into local feature coordinates that are invariant to translation, rotation, scale, and other imaging parameters



Features Descriptors

Choosing interest points

Where would you tell your friend to meet you?

Choosing interest points

Where would you tell your friend to meet you?

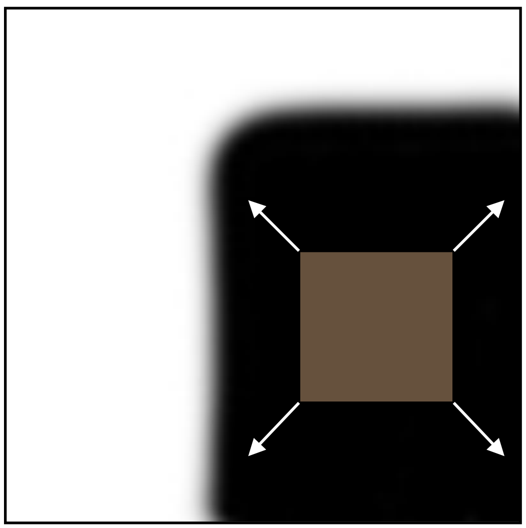
Many Existing Detectors Available

Hessian & Harris	[Beaudet '78], [Harris '88]
Laplacian, DoG	[Lindeberg '98], [Lowe 1999]
Harris-/Hessian-Laplace	[Mikolajczyk & Schmid '01]
Harris-/Hessian-Affine	[Mikolajczyk & Schmid '04]
EBR and IBR	[Tuytelaars & Van Gool '04]
MSER	[Matas '02]
Salient Regions	[Kadir & Brady '01]
Others…	

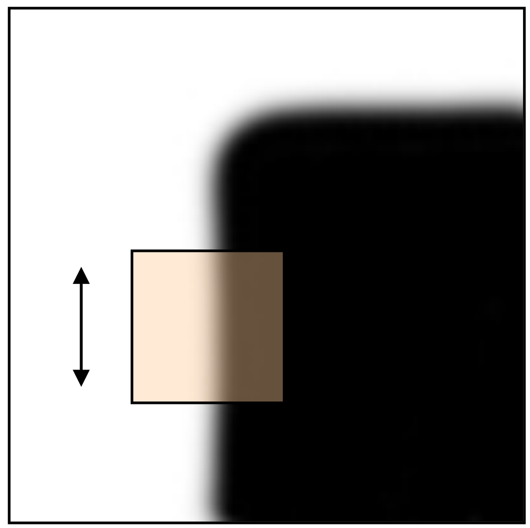
- What points would you choose?

Corner Detection: Basic Idea

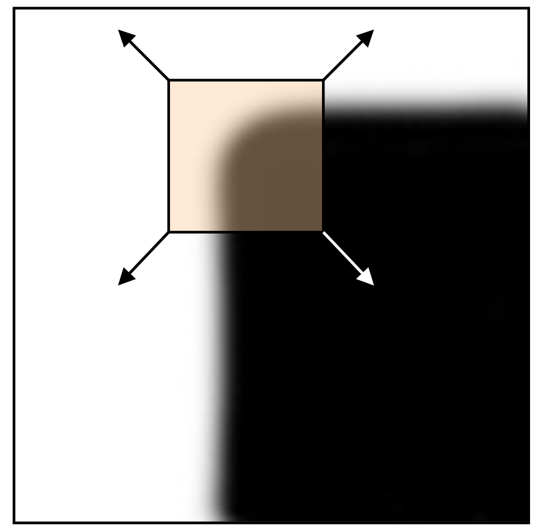
- We should easily recognize the point by looking through a small window
- Shifting a window in *any direction* should give *a large change* in intensity



"flat" region:
no change in
all directions



"edge":
no change along
the edge direction



"corner":
significant change
in all directions

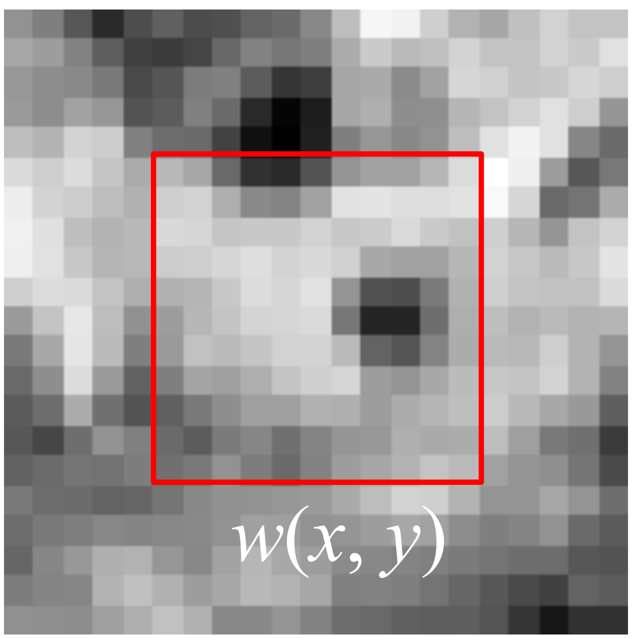
Source: A. Efros

Corner Detection: Mathematics

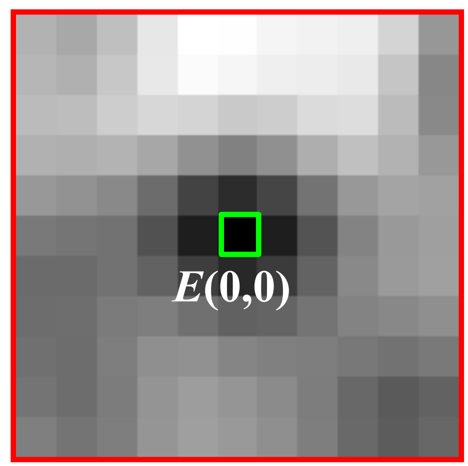
Change in appearance of window *w*(*x*,*y*)
for the shift [*u*,*v*]:

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

$I(x, y)$



$E(u, v)$



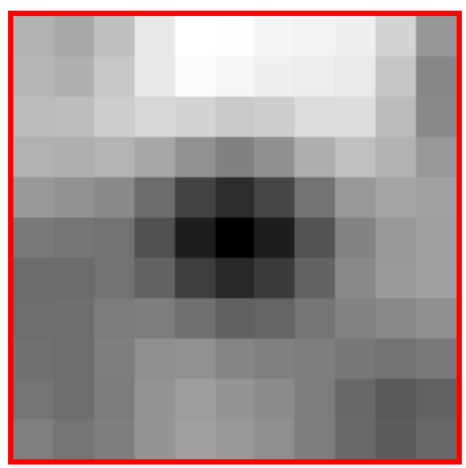
Corner Detection: Mathematics

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

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

We want to find out how this function behaves for small shifts

$E(u, v)$



Taylor series approx to shifted image

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

$$= \sum_{x,y} w(x,y)[uI_x + vI_y]^2$$

$$= \sum_{x,y} w(x,y)(u \quad v)\begin{bmatrix} I_x I_x & I_x I_y \\ I_x I_y & I_y I_y \end{bmatrix}\begin{pmatrix} u \\ v \end{pmatrix}$$

Corner Detection: Mathematics

The quadratic approximation simplifies to

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

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

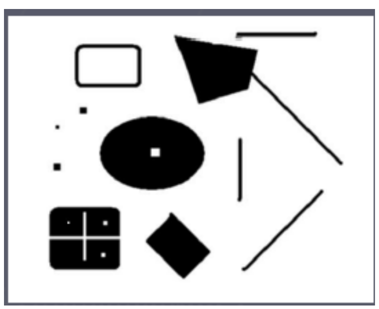
$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

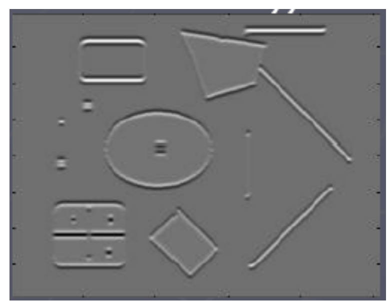
$$M = \begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} = \sum \begin{bmatrix} I_x \\ I_y \end{bmatrix} [I_x\ I_y] = \sum \nabla I (\nabla I)^T$$

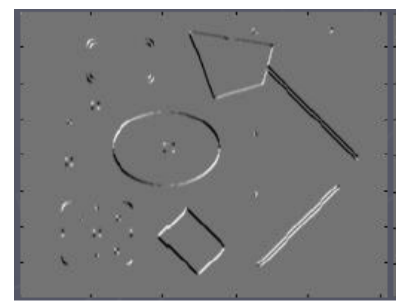
Corners as Distinctive Interest Points

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

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







Notation:

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

Harris Detector: Mathematics

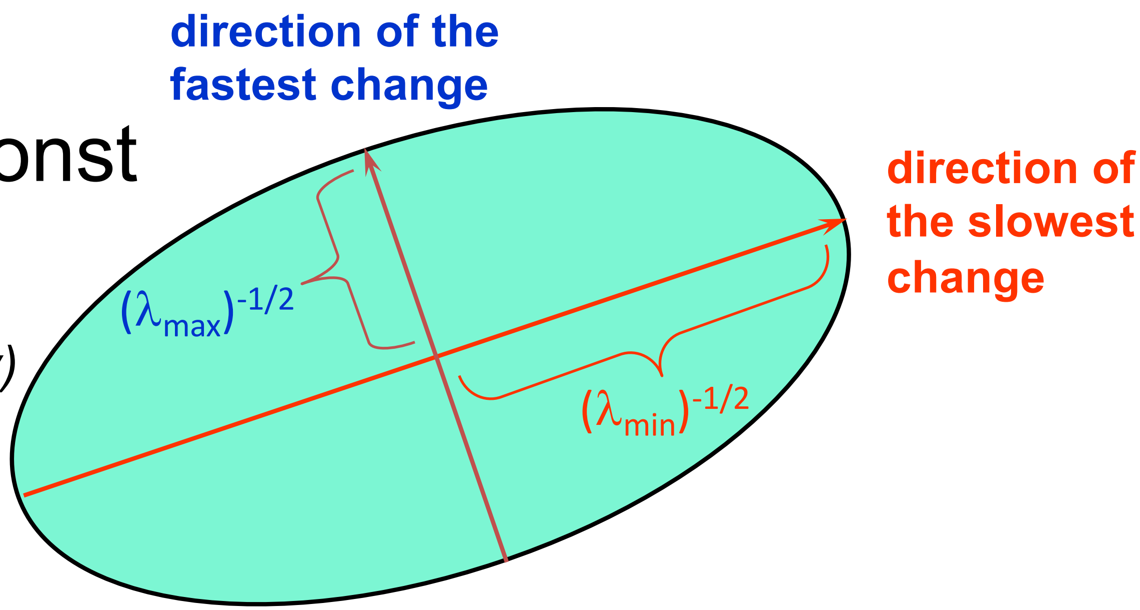
Intensity change in shifting window: eigenvalue analysis

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

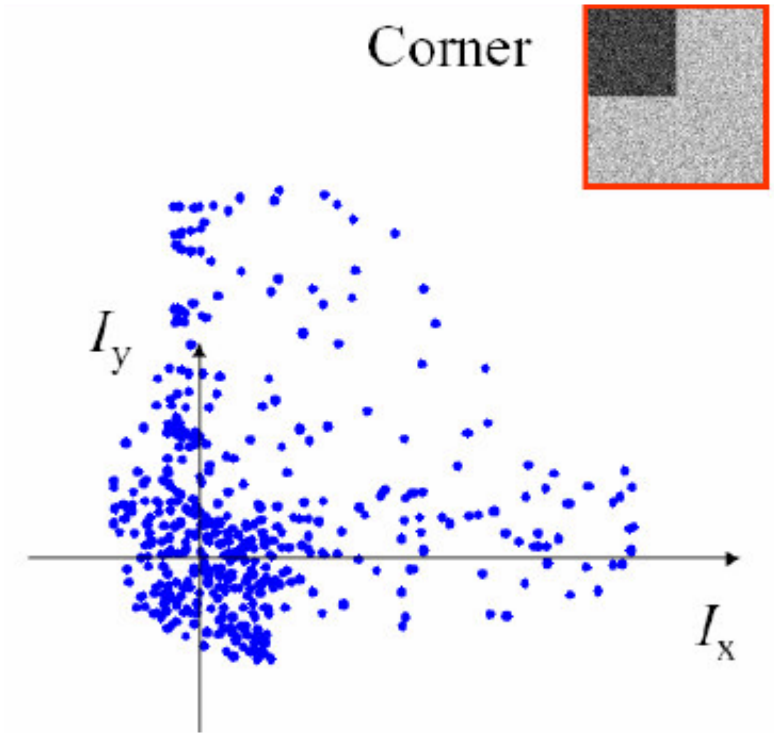
λ_1, λ_2 – eigenvalues of M

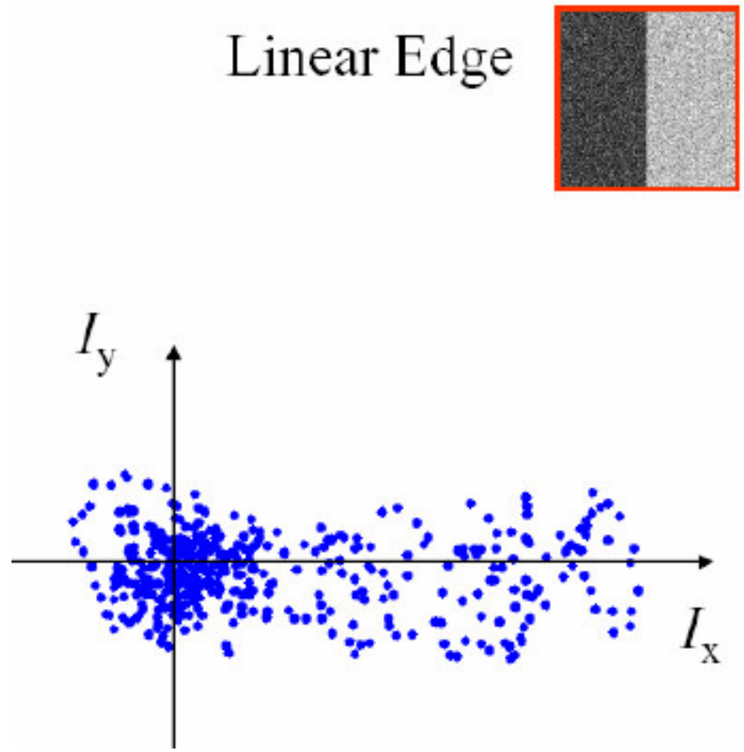
Ellipse $E(u,v)$ = const

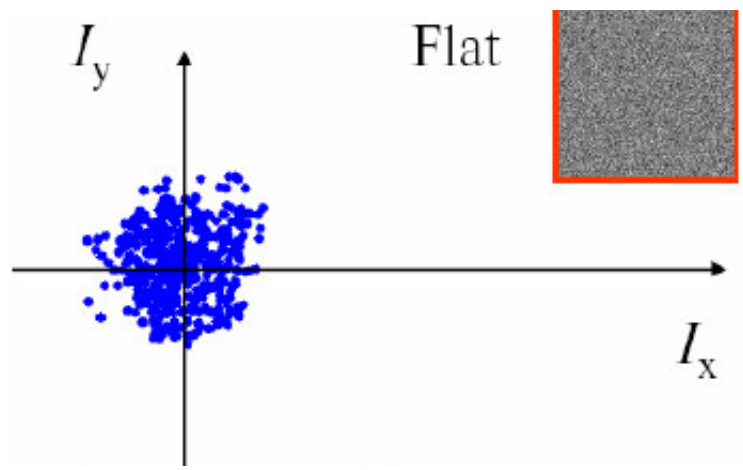
Iso-intensity contour of $E(u,v)$



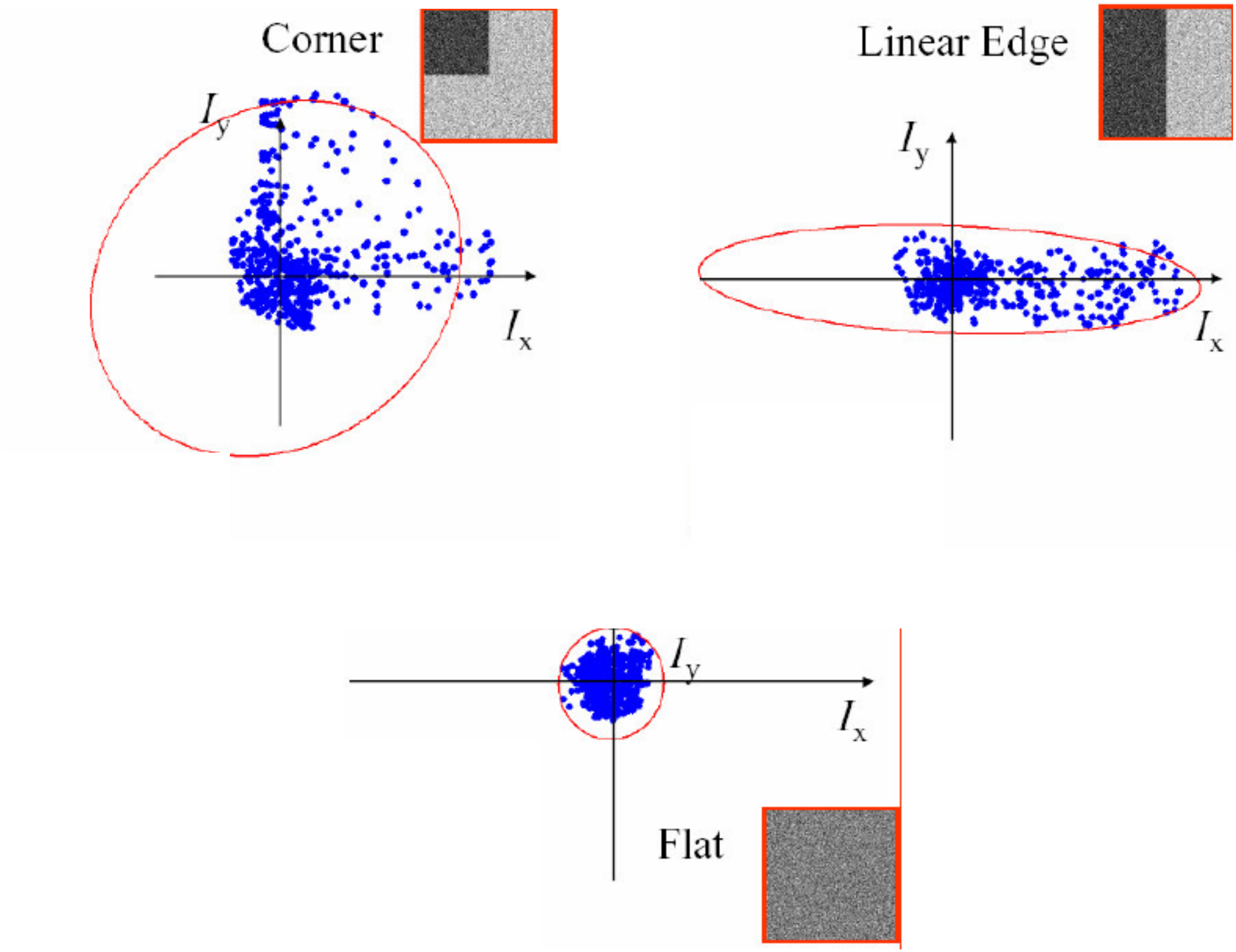
Statistics of x and y derivatives



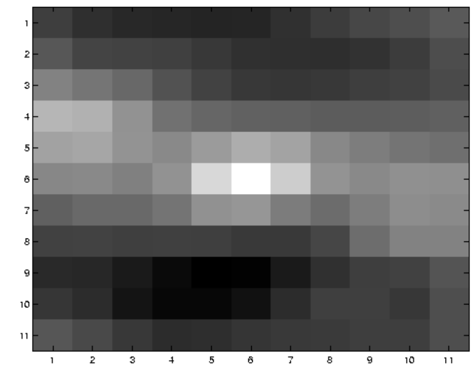


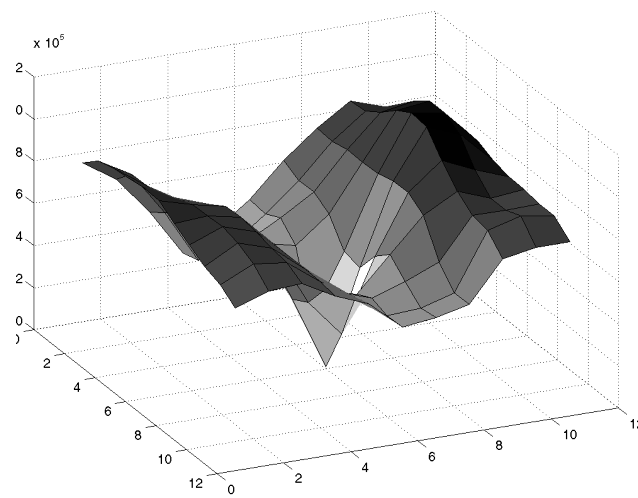


Covariance matrix: ellipse containing data



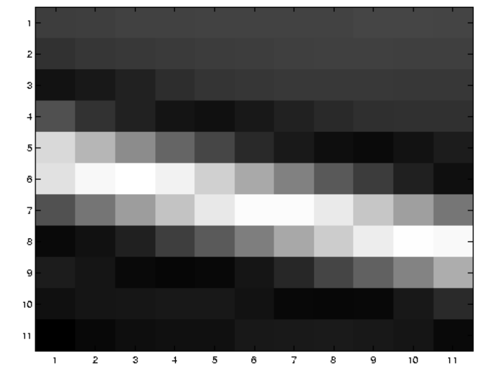
Selecting Good Features

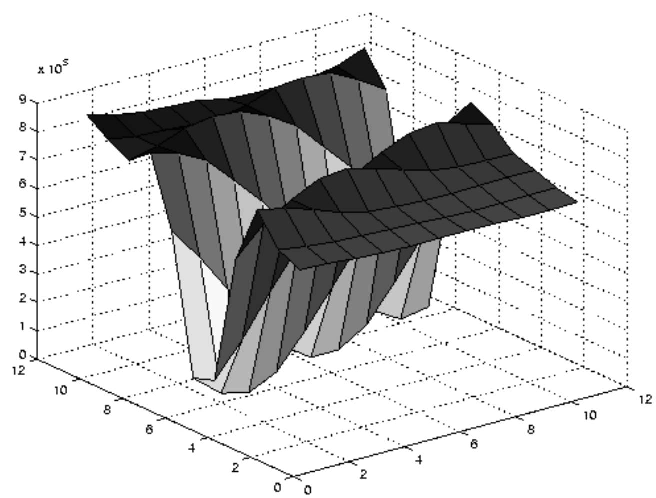




λ_1 and λ_2 are large

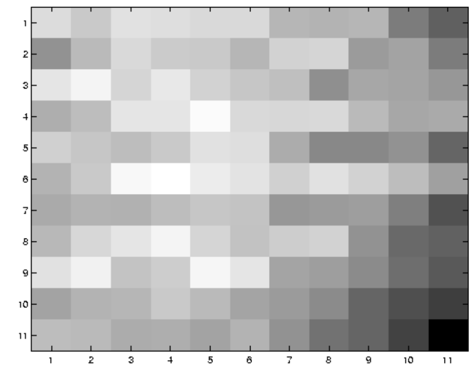
Selecting Good Features

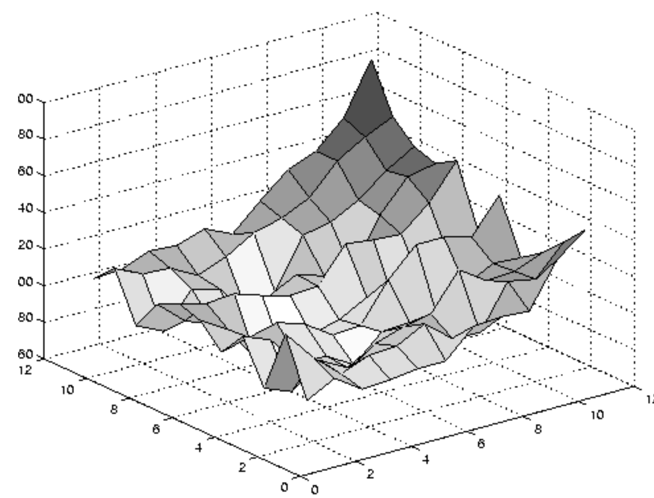




large λ_1, small λ_2

Selecting Good Features

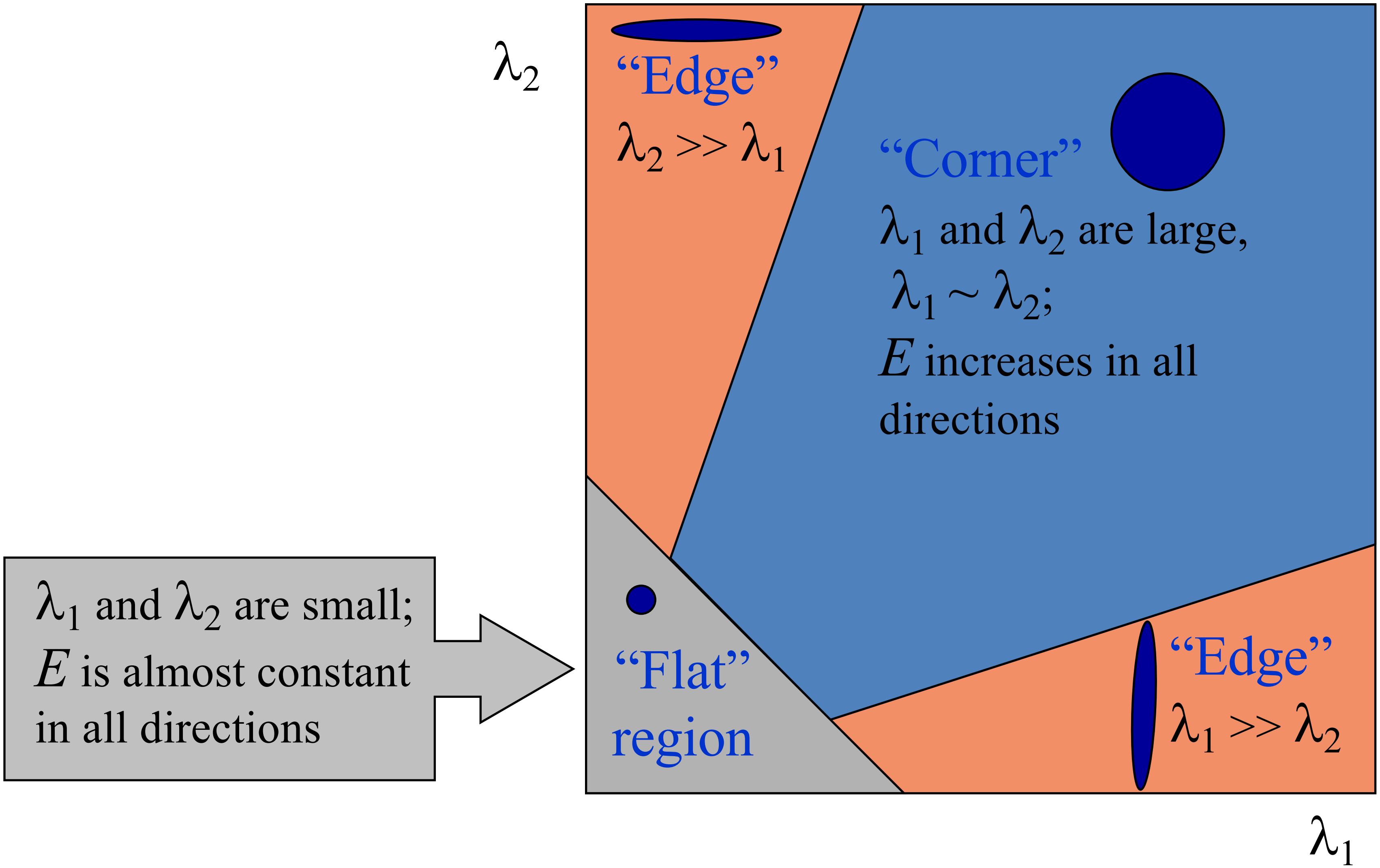




small λ_1, small λ_2

Interpreting the eigenvalues

Classification of image points using eigenvalues of *M*:



Harris Detector: Mathematics

Measure of corner response:

$$R = det(M) - k(trace(M))^2$$

This expression does not requires computing the eigenvalues.

$$det(M) = \lambda_1 \lambda_2$$

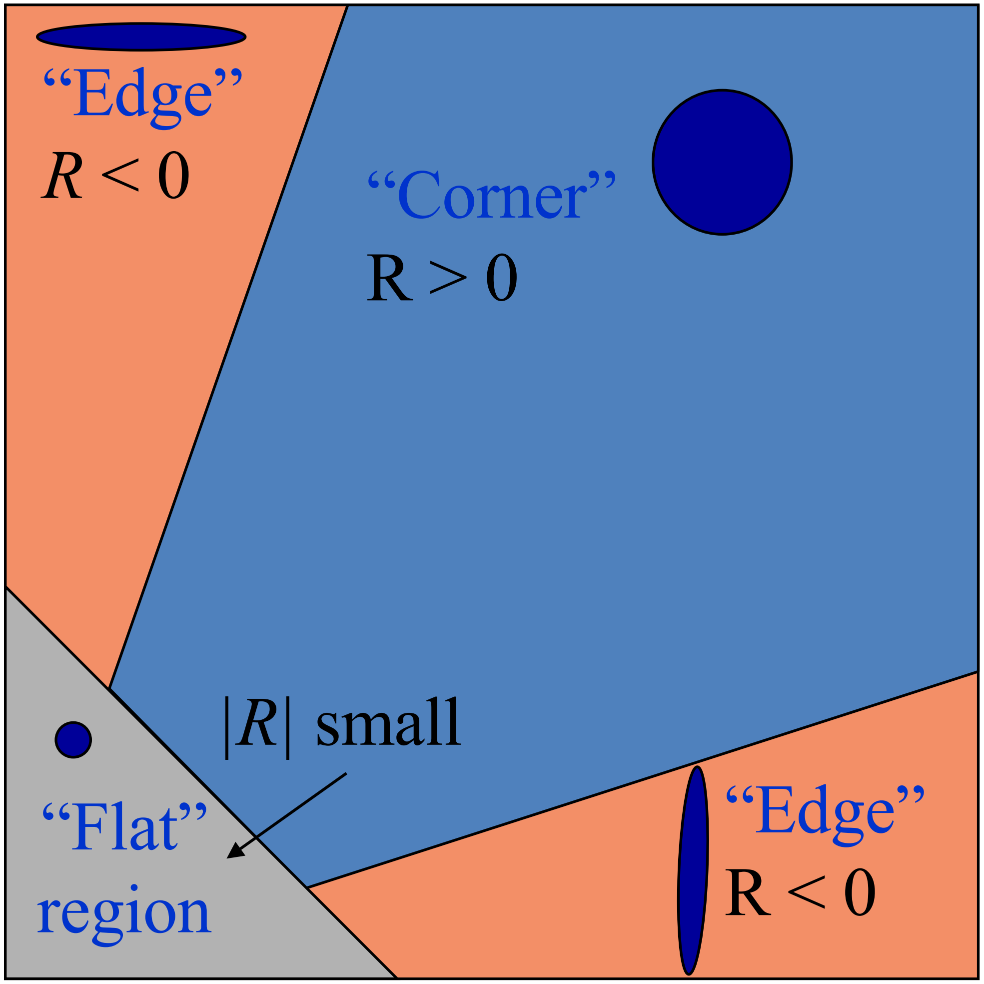
$$trace(M) = \lambda_1 + \lambda_2$$

(*k* – empirical constant, *k* = 0.04-0.06)

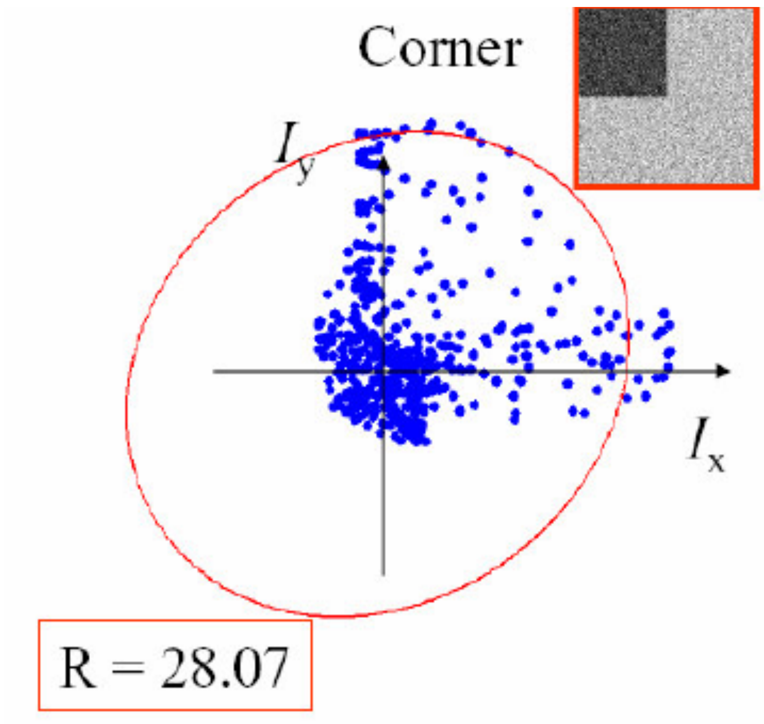
Corner response function

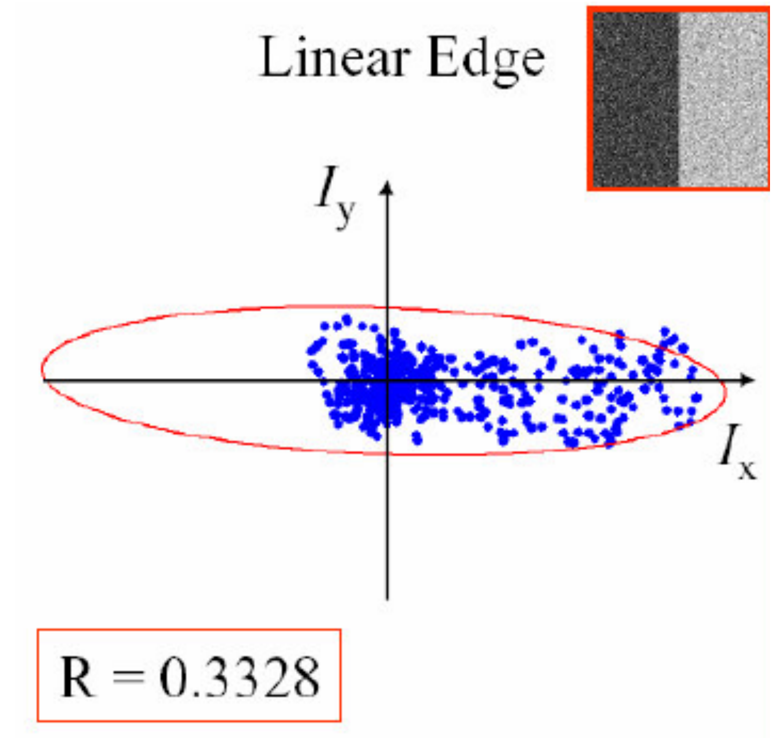
$$R = det(M) - k(trace(M))^2$$

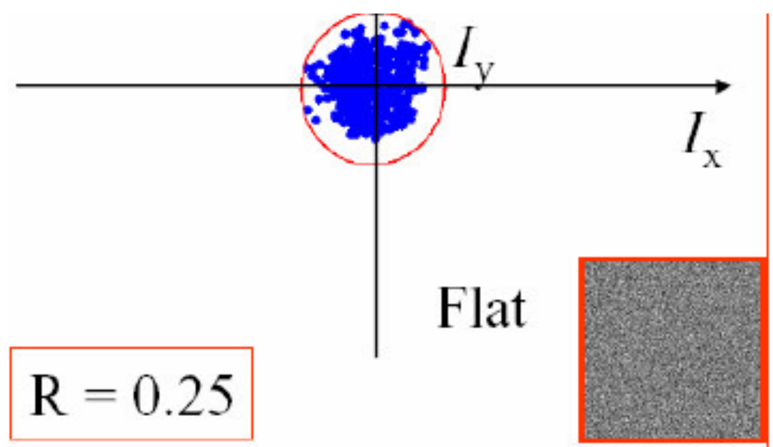
α: constant (0.04 to 0.06)



Interest operator values







Harris corner detector

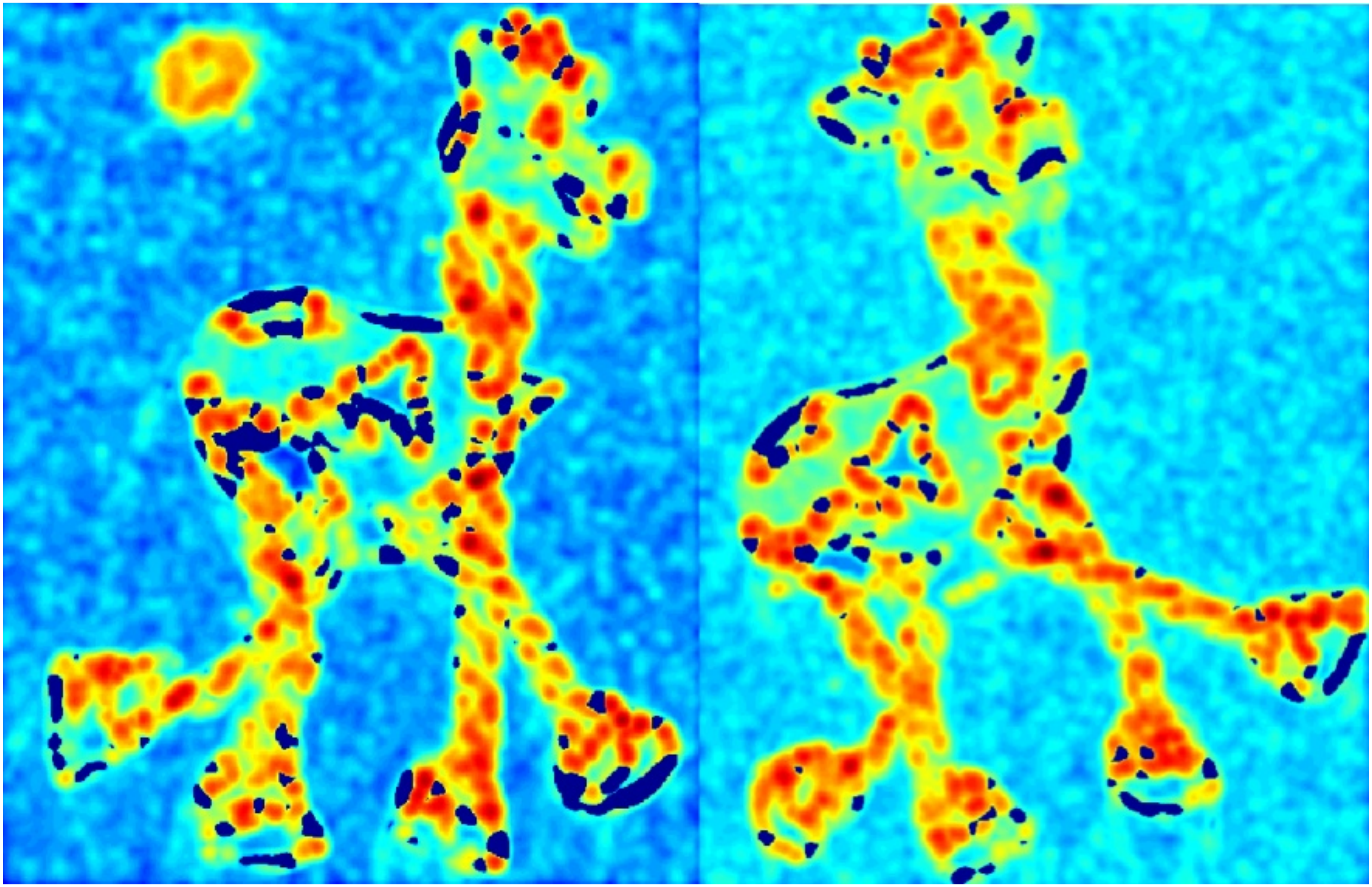
1) Compute *M* matrix for each image window to get their *cornerness* scores.
2) Find points whose surrounding window gave large corner response (f > threshold)
3) Take the points of local maxima, i.e., perform non-maximum suppression

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

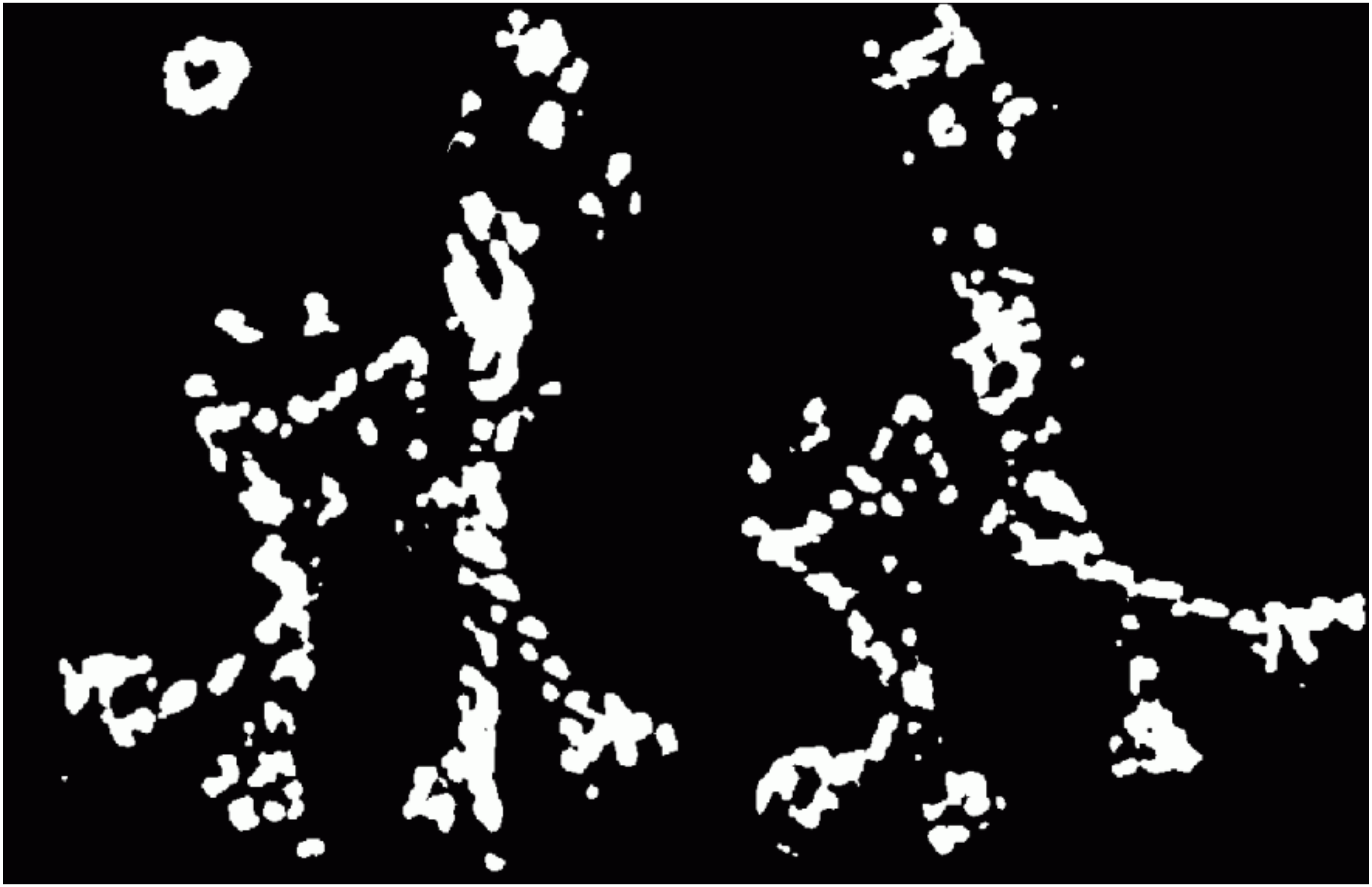
Harris Detector: Steps

Compute corner response R



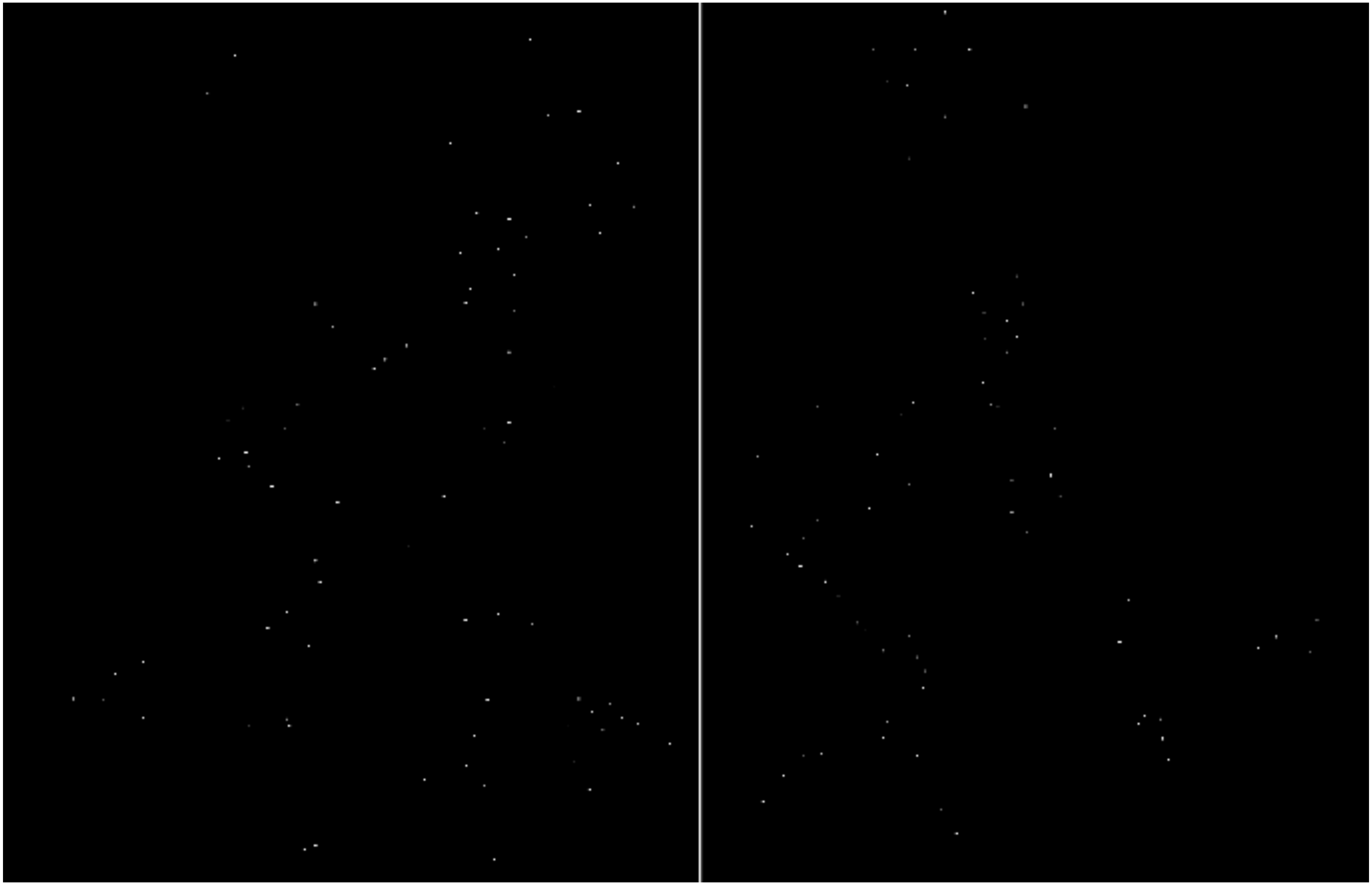
Harris Detector: Steps

Find points with large corner response: R>threshold



Harris Detector: Steps

Take only the points of local maxima of R

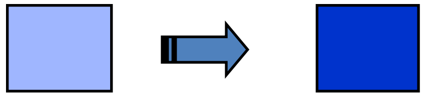


Harris Detector: Steps

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

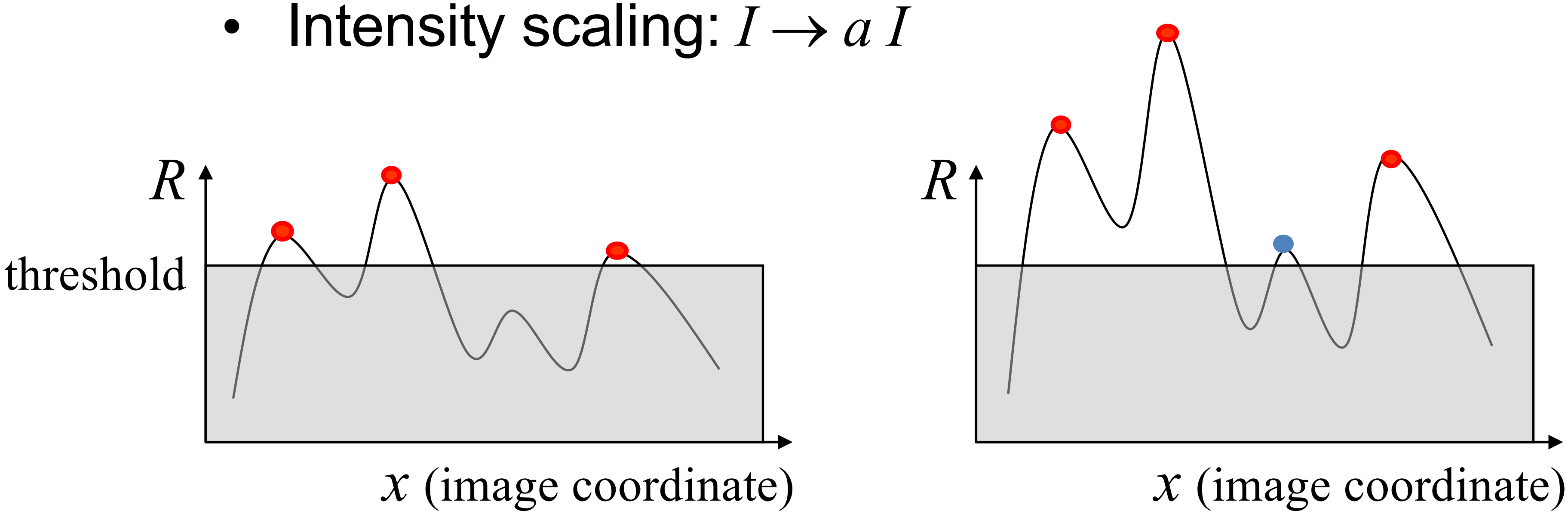
Affine intensity change



$$I \rightarrow a\,I + b$$

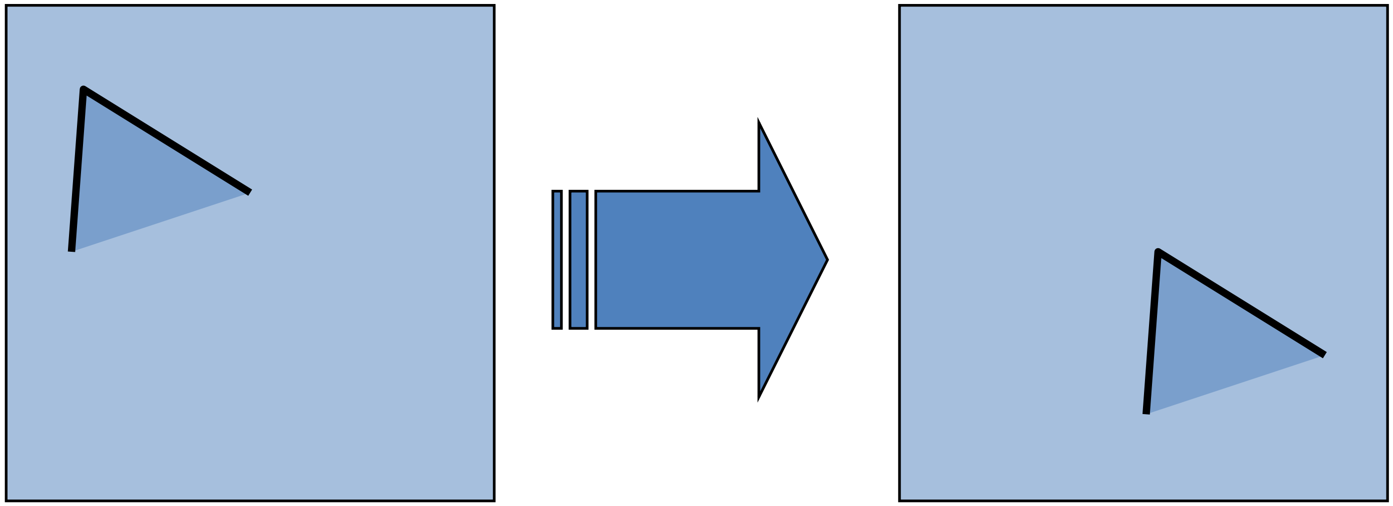
- Only derivatives are used => invariance to intensity shift $I \rightarrow I + b$

- Intensity scaling: $I \rightarrow a\,I$



Partially invariant to affine intensity change

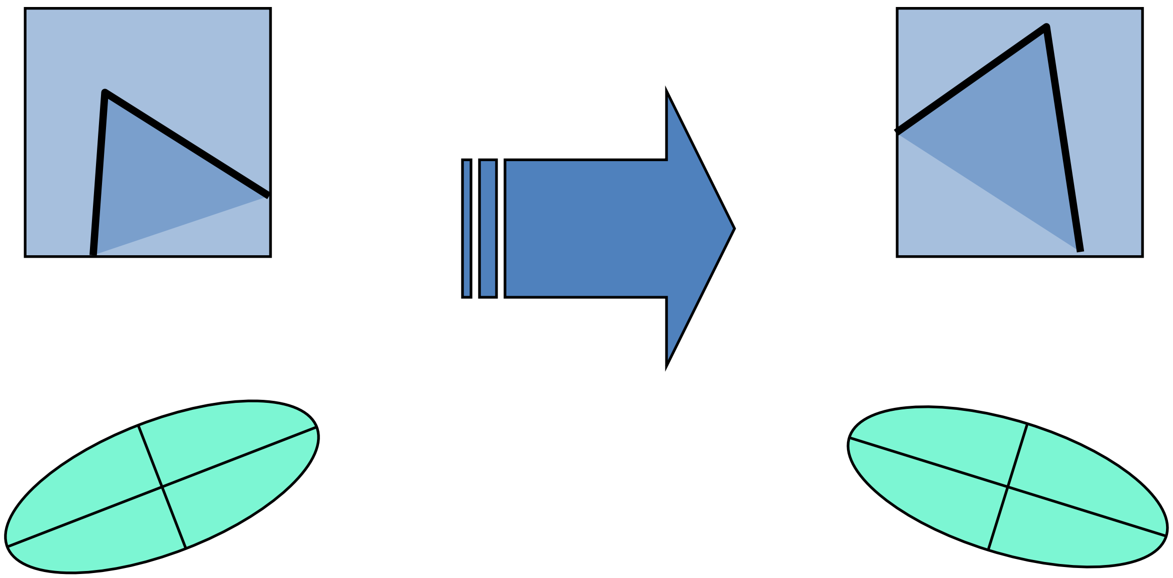
Image translation



- Derivatives and window function are shift-invariant

Corner location is covariant w.r.t. translation

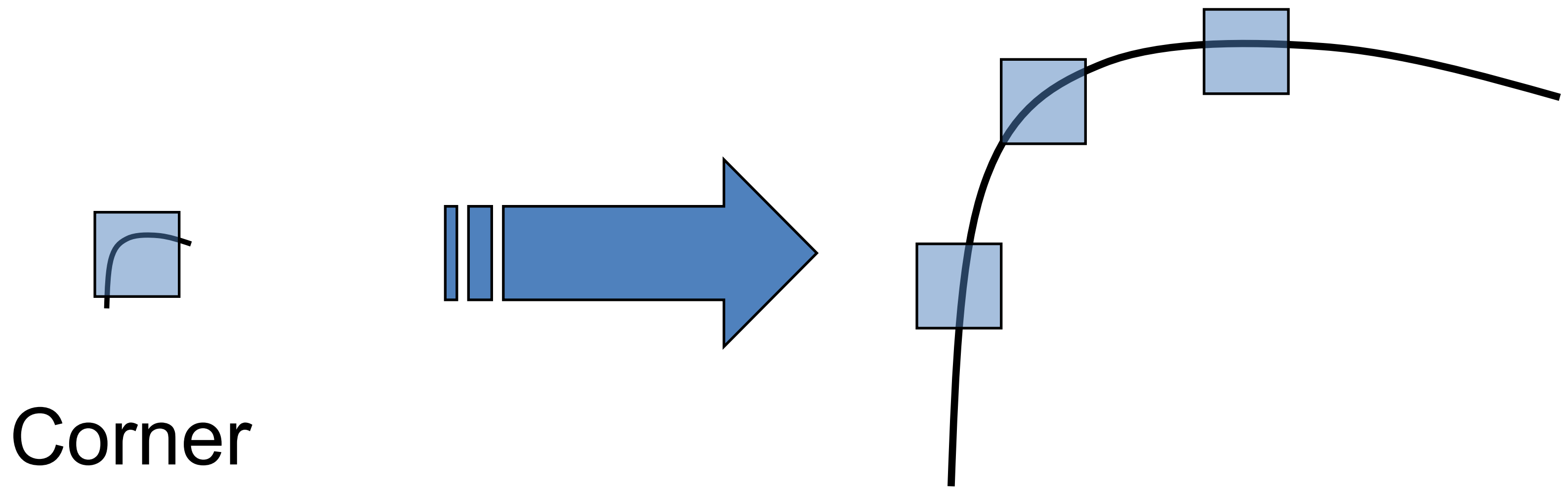
Image rotation



Second moment ellipse rotates but its shape (i.e. eigenvalues) remains the same

Corner location is covariant w.r.t. rotation

Scaling



All points will be classified as edges

Corner location is not covariant to scaling!

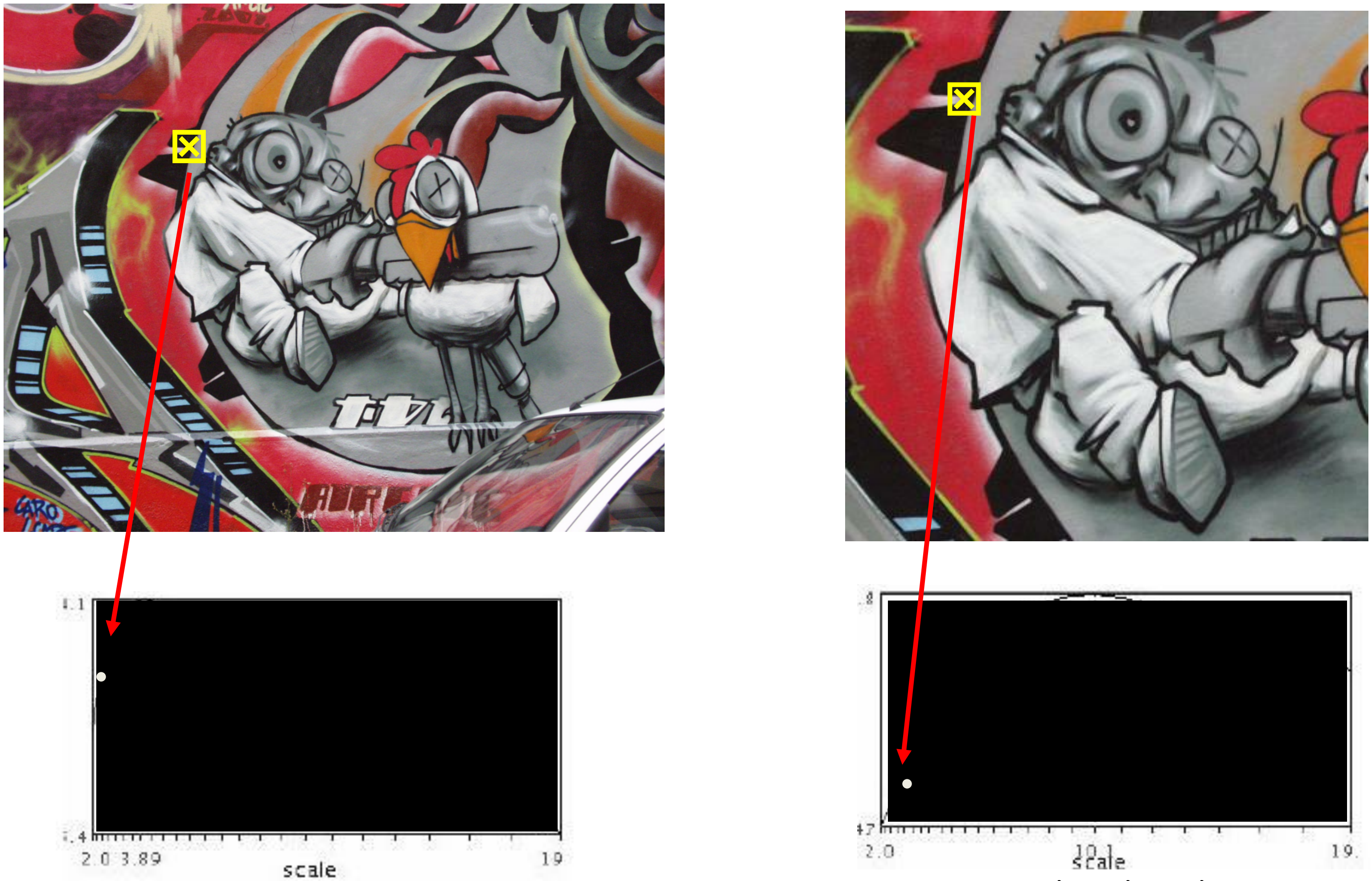
Automatic Scale Selection

$$f(I(x,\sigma)) = f(I'(x',\sigma'))$$

How to find corresponding patch sizes?

Automatic Scale Selection

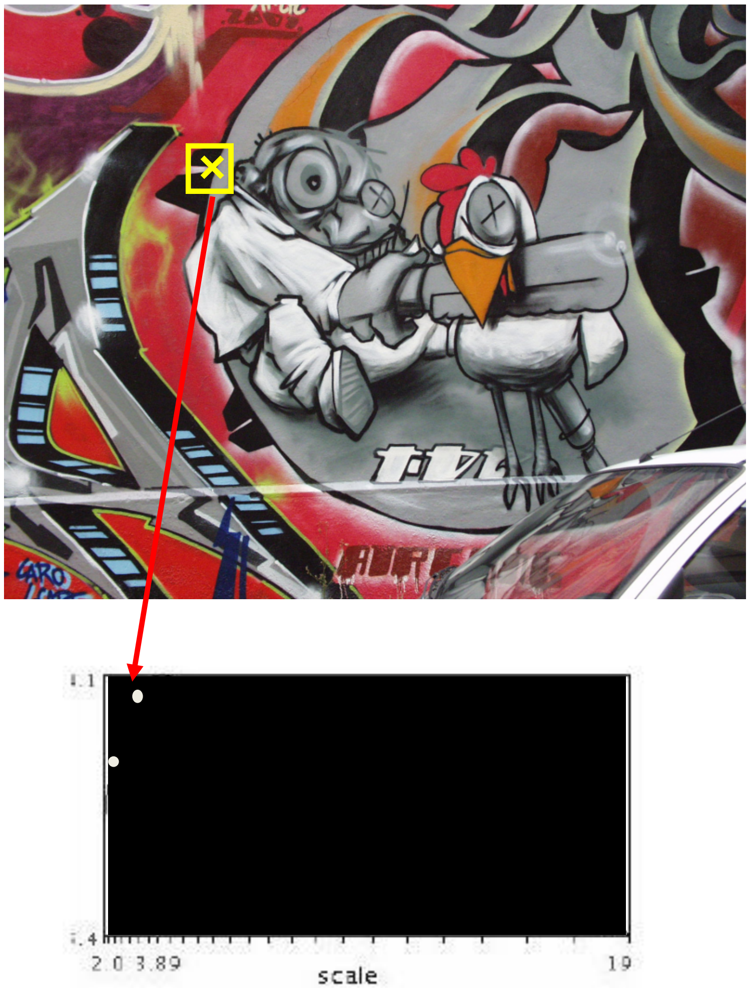
- Function responses for increasing scale (scale signature)



$f(I(x, \sigma))$ $\quad$ $f(I'(x', \sigma'))$

Automatic Scale Selection

- Function responses for increasing scale (scale signature)

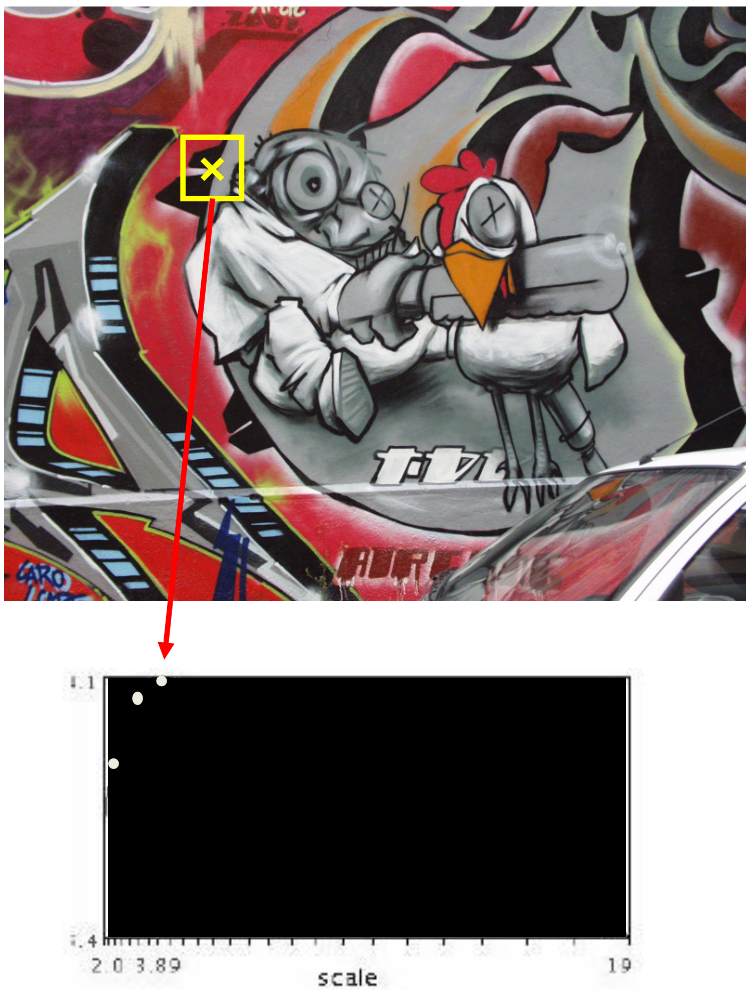


$f(I(x,\sigma))$

$f(I'(x',\sigma'))$

Automatic Scale Selection

- Function responses for increasing scale (scale signature)

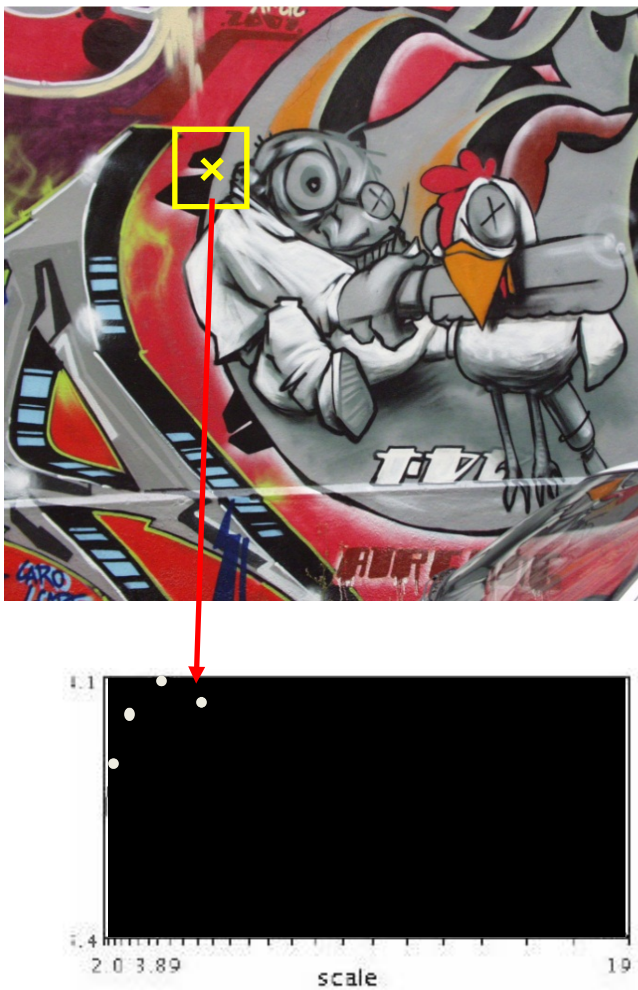


$$f(I(x, \sigma))$$

$$f(I'(x', \sigma'))$$

Automatic Scale Selection

- Function responses for increasing scale (scale signature)

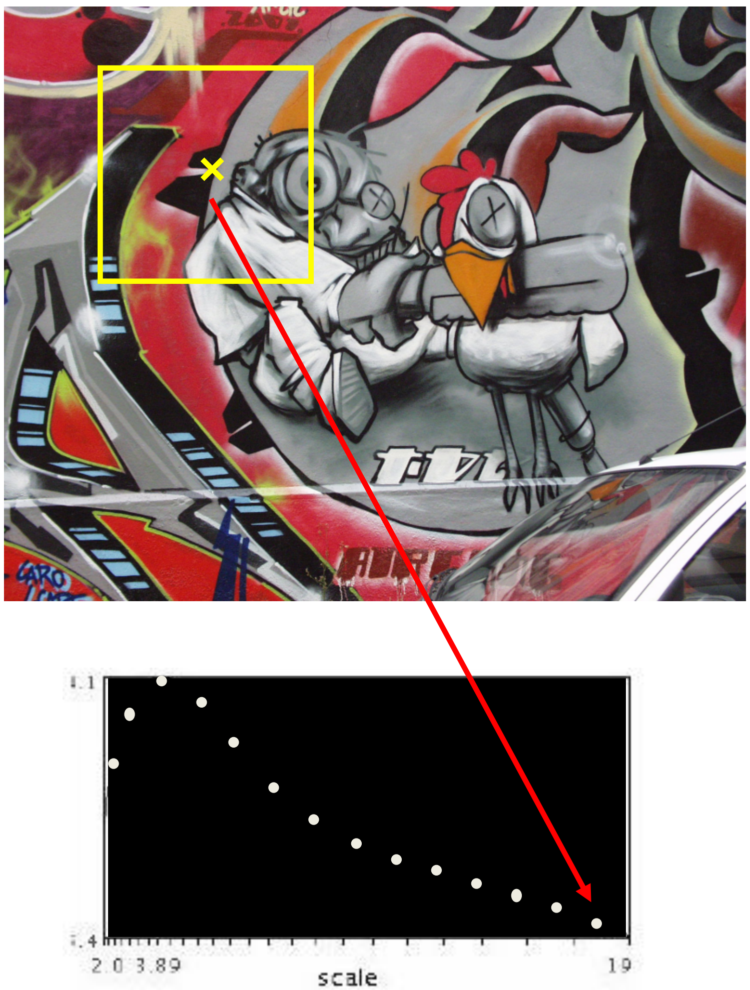


$$f(I(x,\sigma))$$

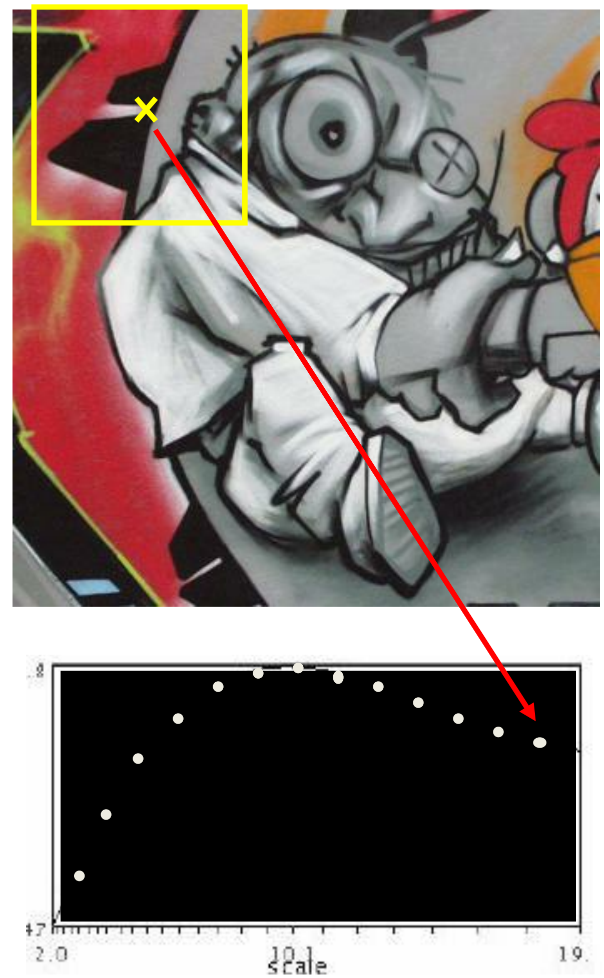
$$f(I'(x',\sigma'))$$

Automatic Scale Selection

- Function responses for increasing scale (scale signature)



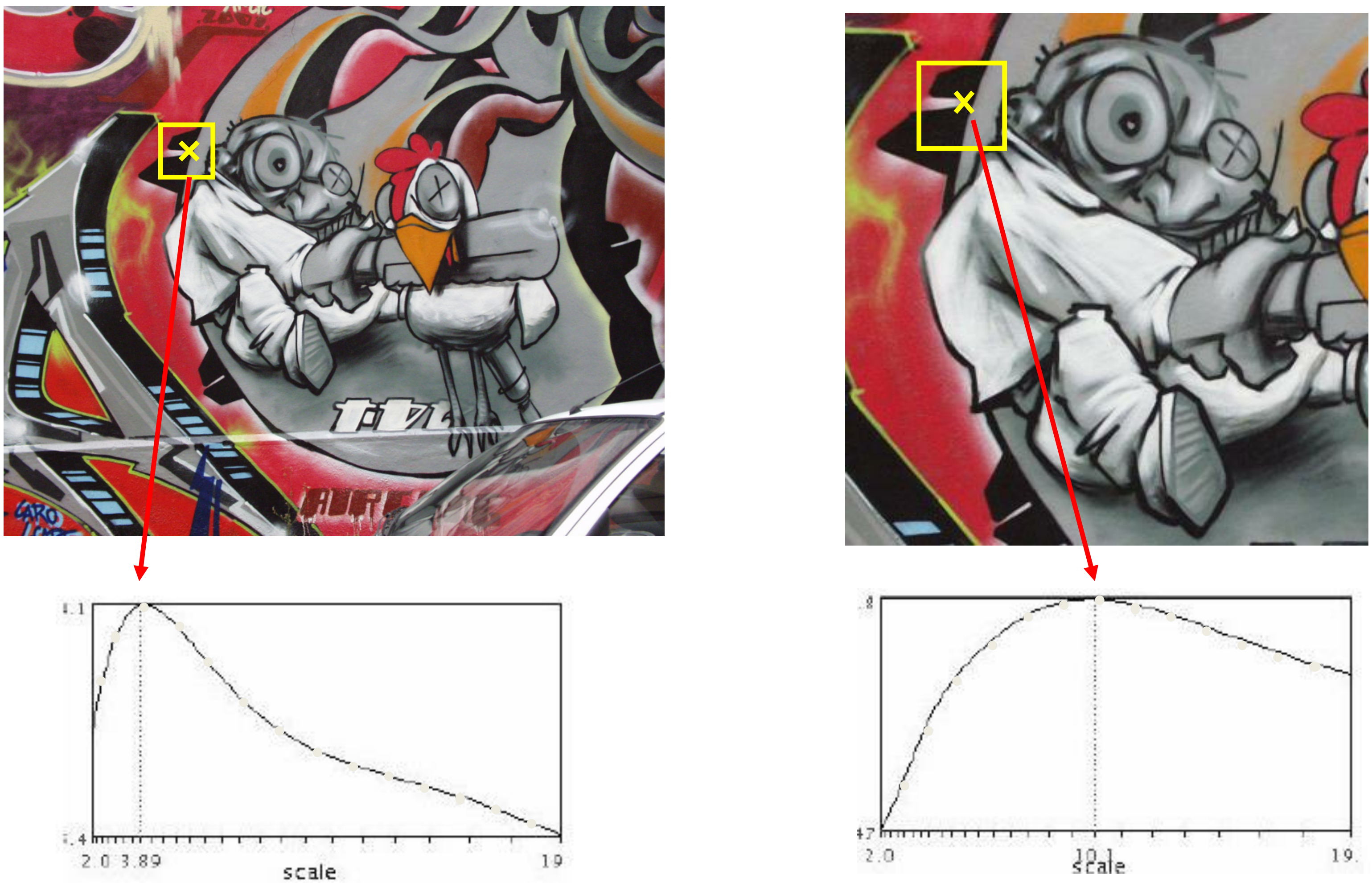
$$f(I(x,\sigma))$$



$$f(I'(x',\sigma'))$$

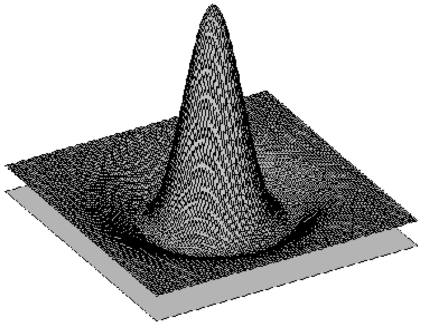
Automatic Scale Selection

- Function responses for increasing scale (scale signature)



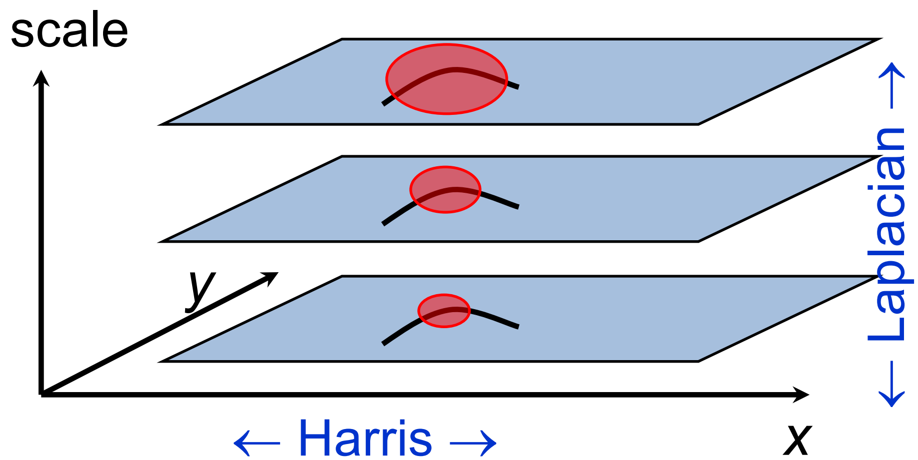
$f(I(x,\sigma))$ $f(I'(x',\sigma'))$

Difference-of-Gaussian (DoG)

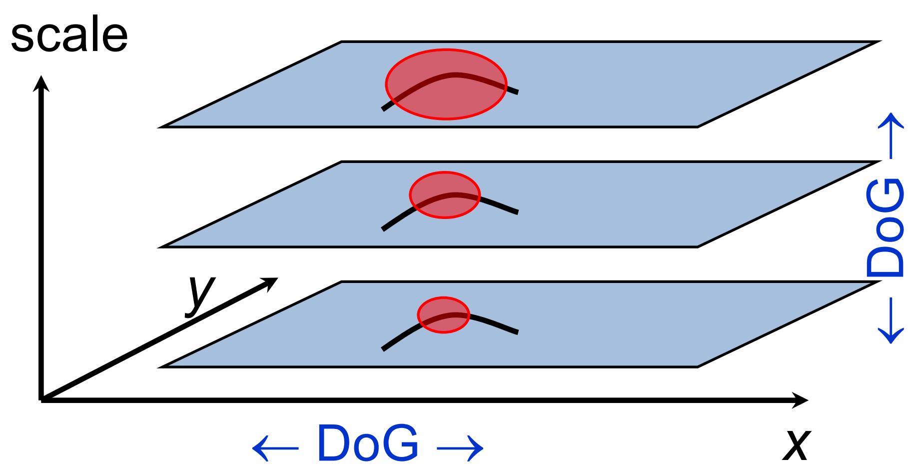


Scale Invariant Detectors

- Harris-Laplacian[1]
  Find local maximum of:
  - Harris corner detector in space (image coordinates)
  - Laplacian in scale



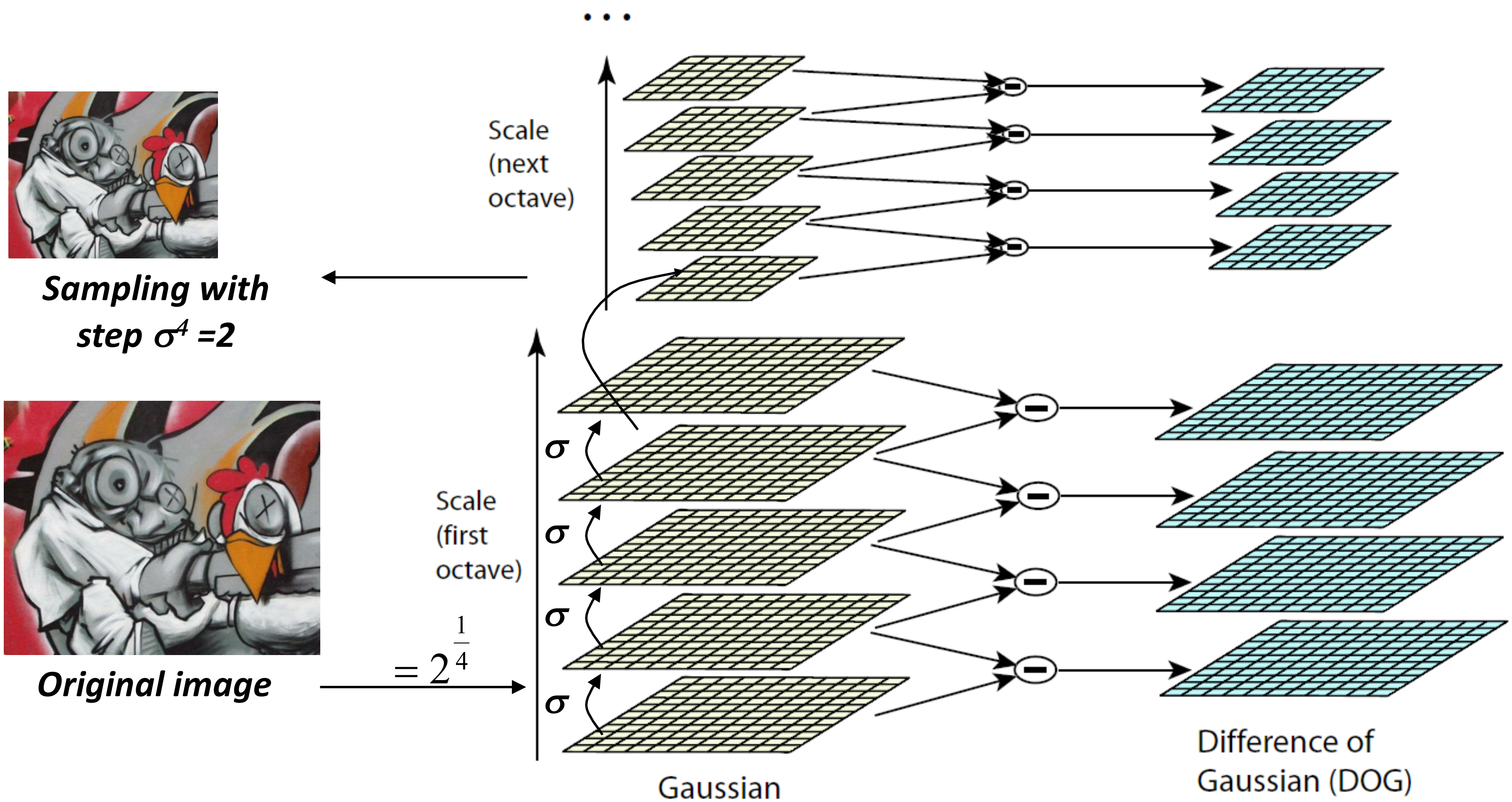
- SIFT (Lowe)[2]
  Find local maximum of:
  - Difference of Gaussians in space and scale



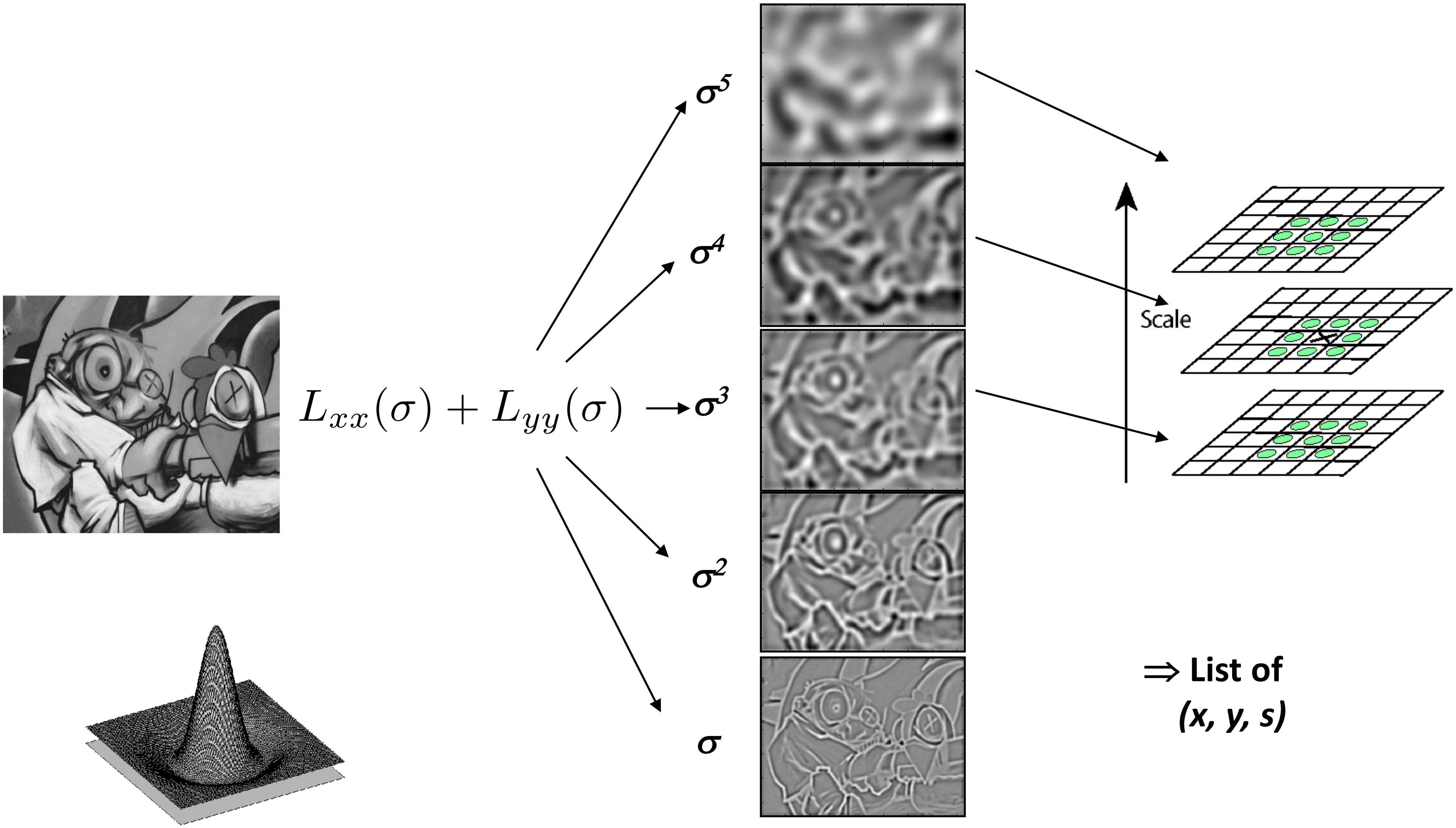
[1] K.Mikolajczyk, C.Schmid. "Indexing Based on Scale Invariant Interest Points". ICCV 2001
[2] D.Lowe. "Distinctive Image Features from Scale-Invariant Keypoints". IJCV 2004

DoG – Efficient Computation

- Computation in Gaussian scale pyramid

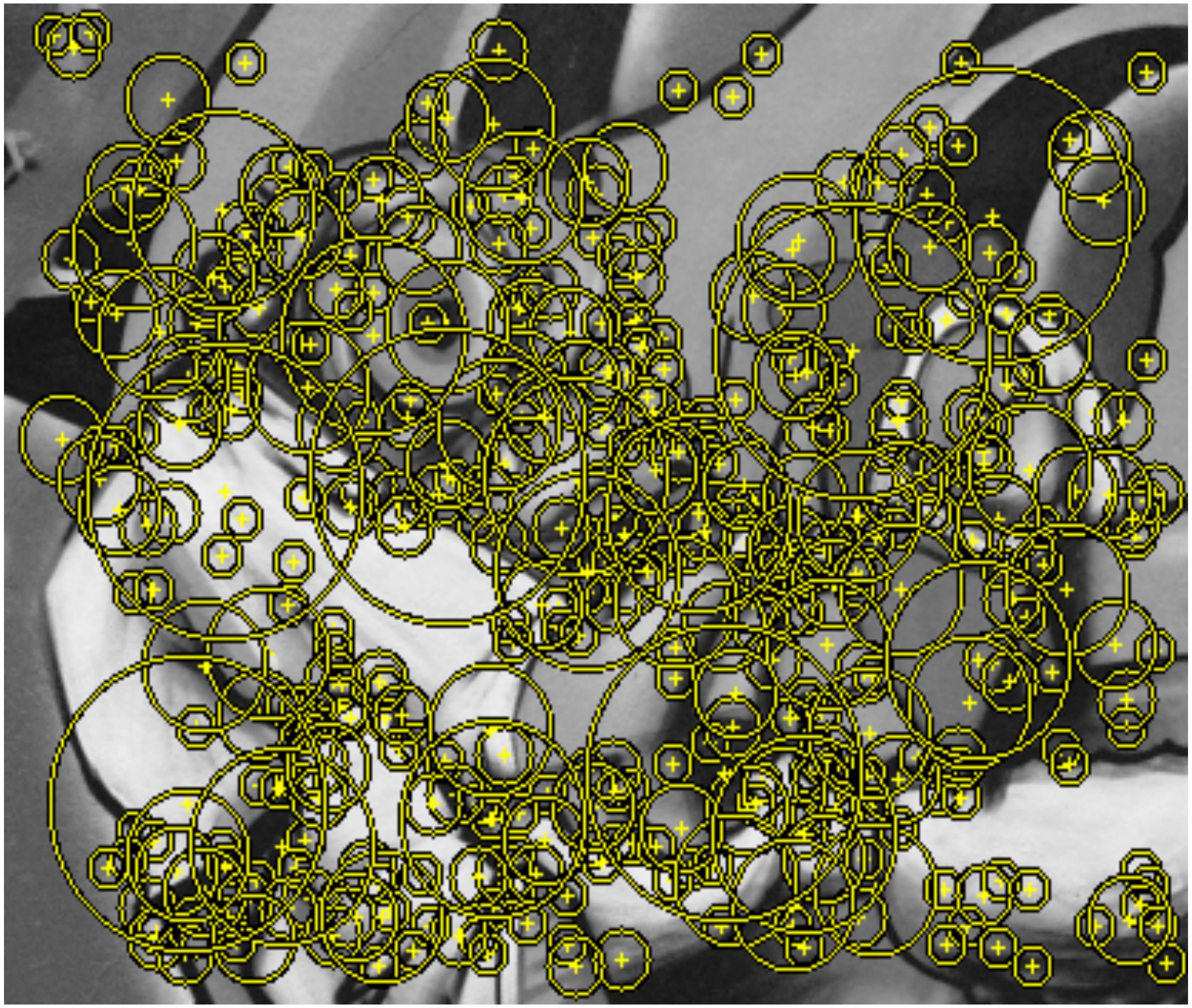


Find local maxima in position-scale space of Difference-of-Gaussian



$\Rightarrow$ **List of *(x, y, s)***

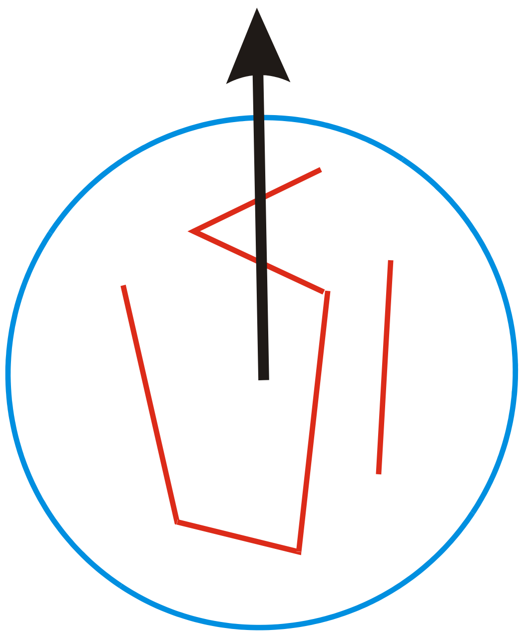
Results: Difference-of-Gaussian



Orientation Normalization

- Compute orientation histogram
- Select dominant orientation
- Normalize: rotate to fixed orientation

[Lowe, SIFT, 1999]



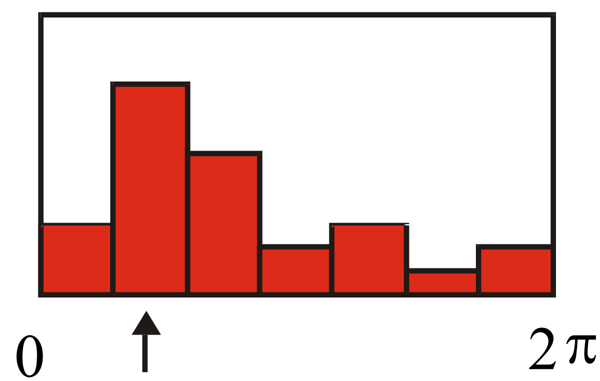
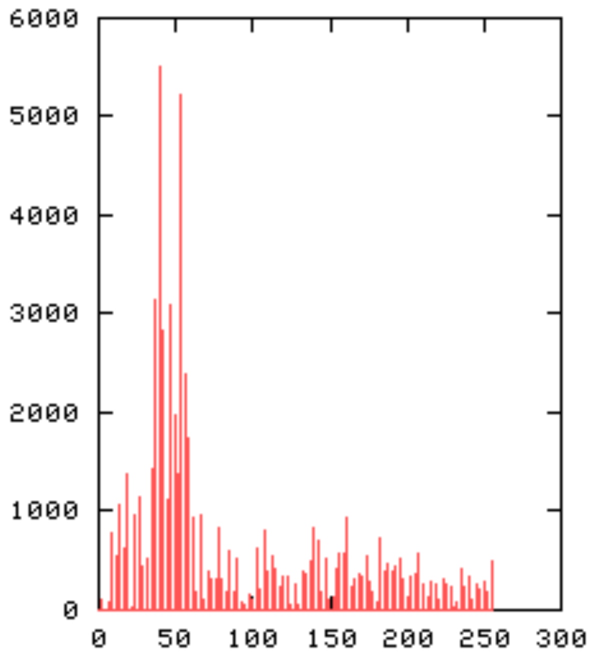


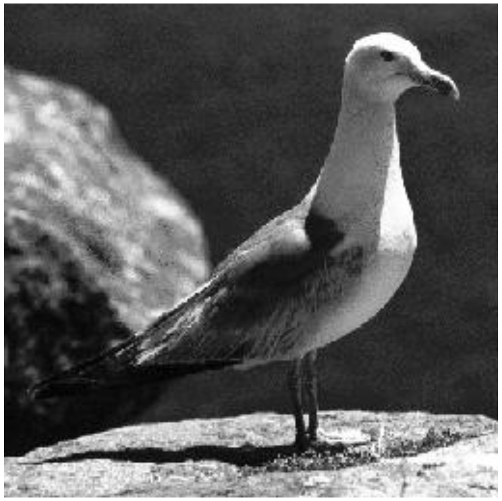
Image representations

- Templates
  - Intensity, gradients, etc.

- Histograms
  - Color, texture, SIFT descriptors, etc.

Image Representations: Histograms





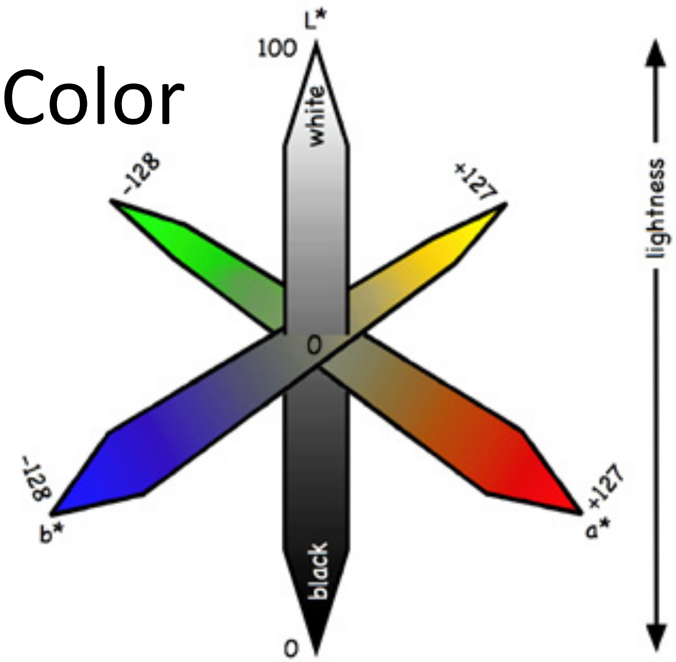
Global histogram

- Represent distribution of features
  - Color, texture, depth, …

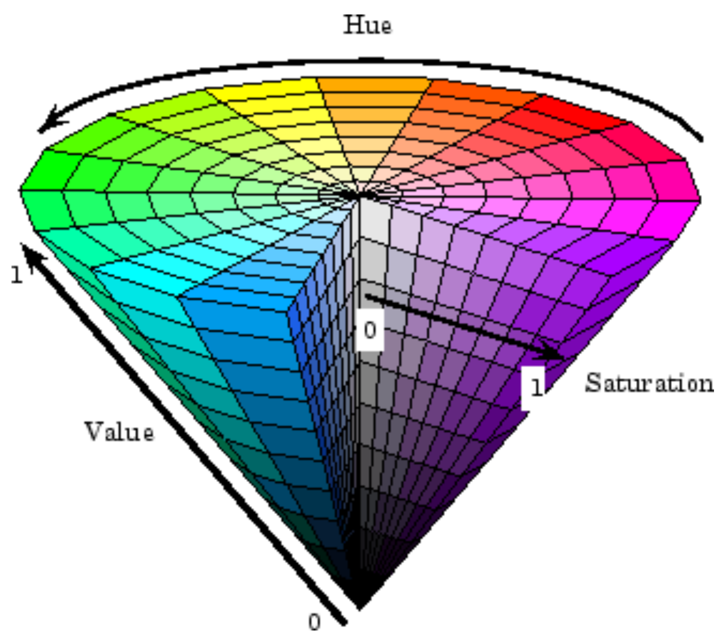
Images from Dave Kauchak

What kind of things do we compute histograms of?

- Color



L*a*b* color space

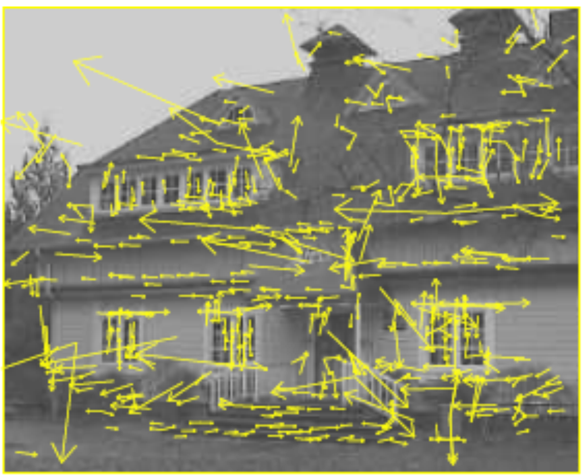


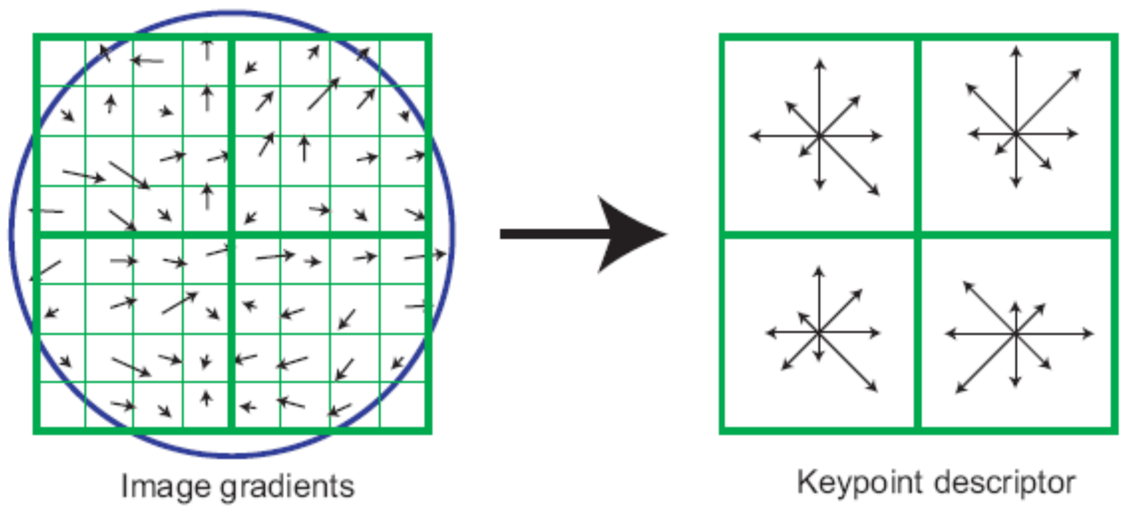
HSV color space

- Texture (filter banks or HOG over regions)

What kind of things do we compute histograms of?

Histograms of oriented gradients





SIFT – Lowe IJCV 2004

SIFT vector formation

- Computed on rotated and scaled version of window according to computed orientation & scale
  - resample the window
- Based on gradients weighted by a Gaussian of variance half the window (for smooth falloff)

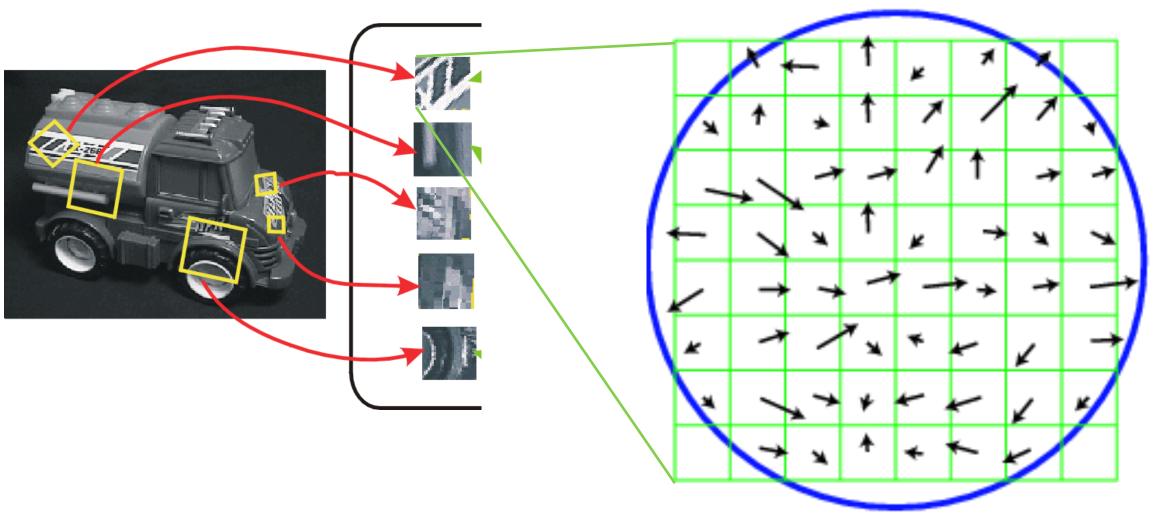
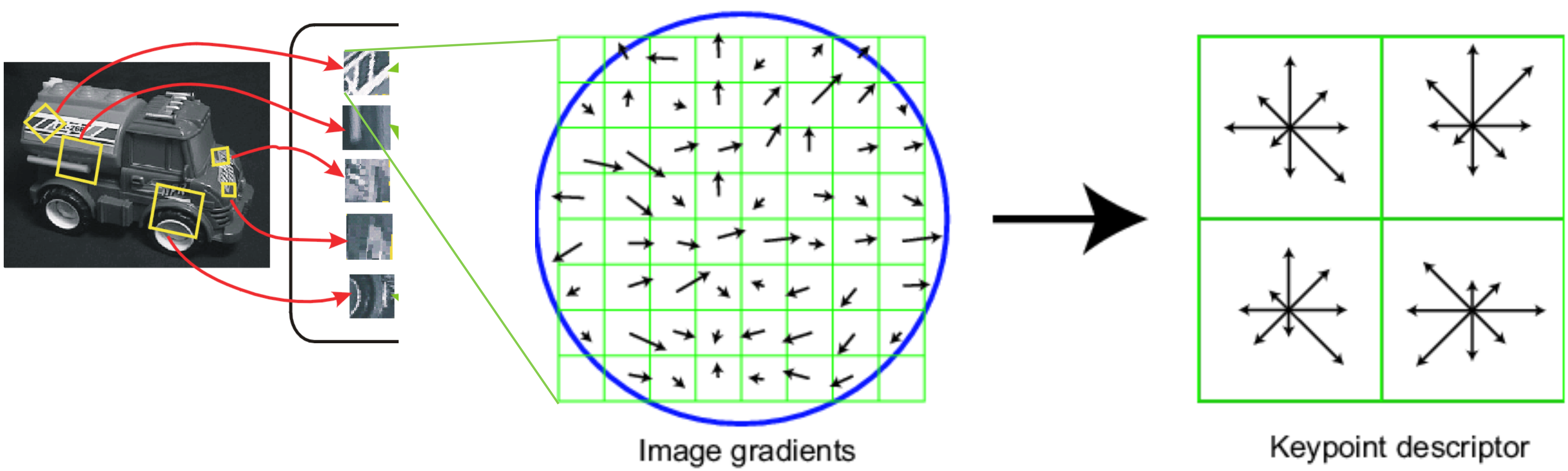


Image gradients

SIFT vector formation

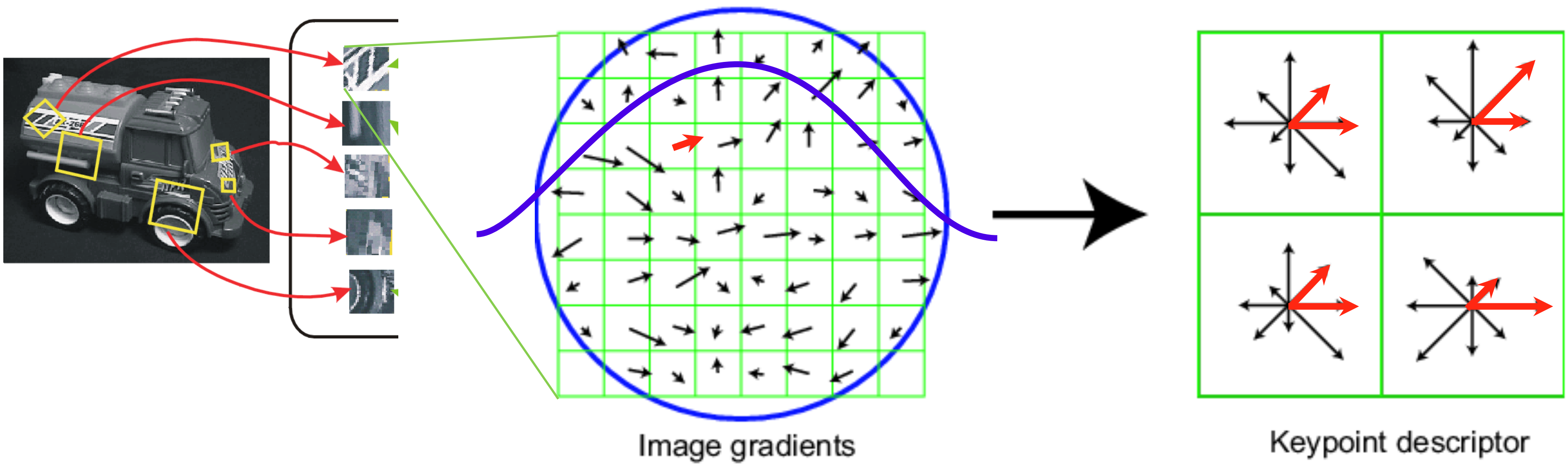
- 4x4 array of gradient orientation histogram weighted by magnitude
- 8 orientations x 4x4 array = 128 dimensions
- Motivation: some sensitivity to spatial layout, but not too much.



showing only 2x2 here but is 4x4

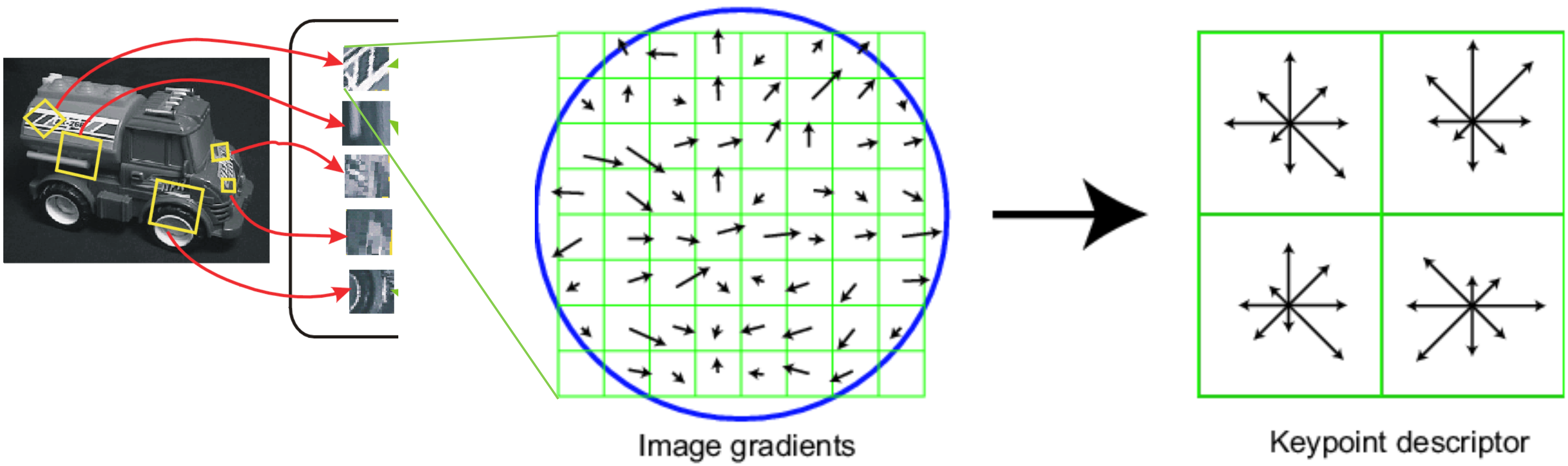
Ensure Smoothness

- Gaussian weight
- Trilinear interpolation
  - a given gradient contributes to 8 bins: 4 in space times 2 in orientation



Reduce Effect of Illumination

- 128-dim vector normalized to 1
- Threshold gradient magnitudes to avoid excessive influence of high gradients
  - after normalization, clamp gradients >0.2
  - renormalize

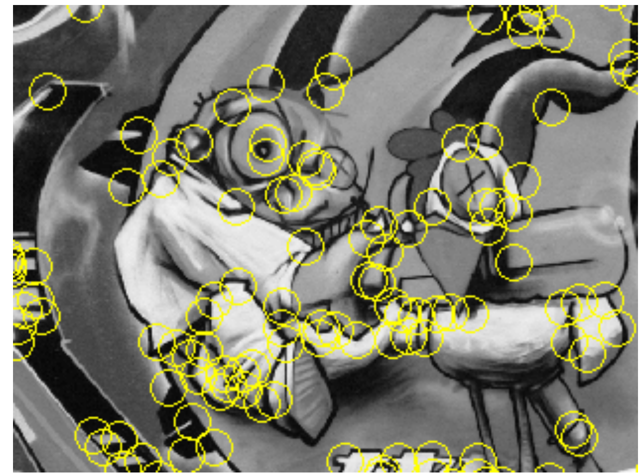


Local Descriptors

- Most features can be thought of as templates, histograms (counts), or combinations
- The ideal descriptor should be
  - Robust
  - Distinctive
  - Compact
  - Efficient
- Most available descriptors focus on edge/gradient information
  - Capture texture information
  - Color rarely used

Things to remember

- Keypoint detection: repeatable and distinctive
  - Corners, blobs, stable regions
  - Harris, DoG



- Descriptors: robust and selective
  - spatial histograms of orientation
  - SIFT