CEE598 - Visual Sensing for Civil Infrastructure Eng. & Mgmt.

Session 9- Image Detectors, Part I

Mani Golparvar-Fard
Department of Civil and Environmental Engineering
3129D, Newmark Civil Engineering Lab
e-mail: mgolpar@illinois.edu

Outline

Image Detectors, Part I

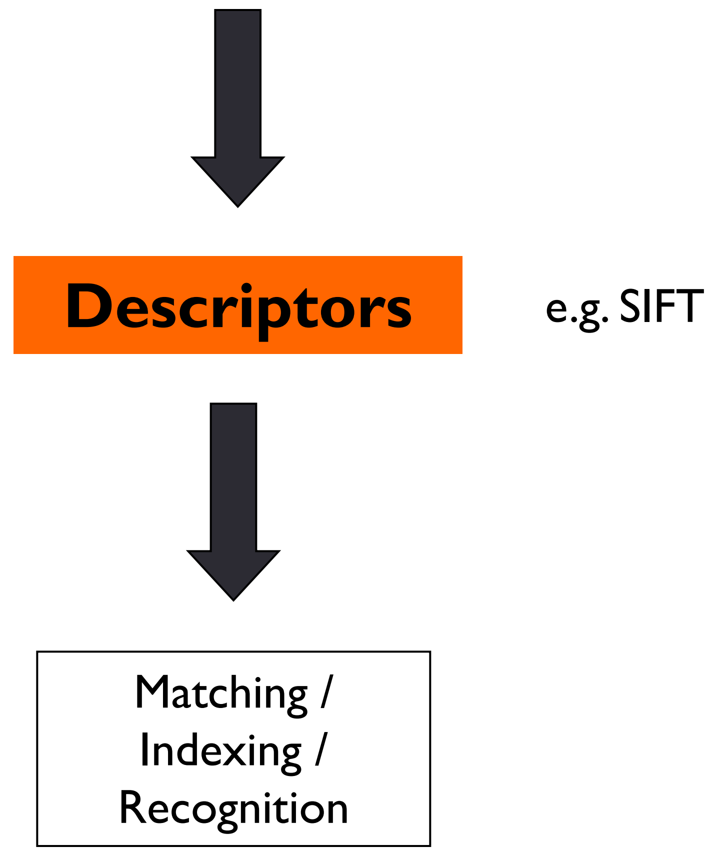
- Edge feature detectors
- Corner feature detectors

Reading: [FP] Chapters 8,9

Some slides in this lecture are courtesy to Prof. S. Savarese, prof F. Li, prof S. Lazebnik, and various other lecturers

Goal

- Identify interesting regions from the images (edges, corners, blobs…)



Linear filtering

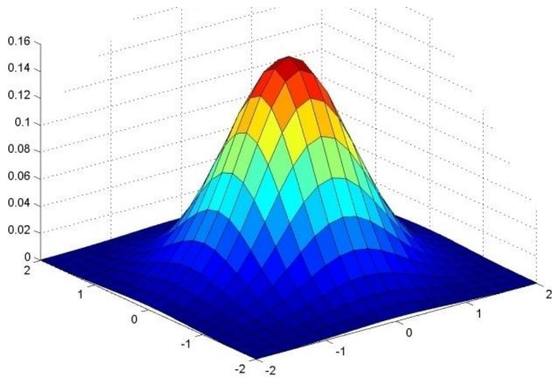
- Convolution:

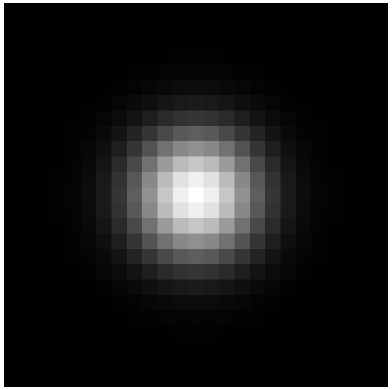
$$(f * g)[m,n] = \sum_{k,l} f[k,l]\, g[m-k, n-l]$$

- Smoothing
- Differentiation

Smoothing with a Gaussian

- Weight contributions of neighboring pixels by nearness





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

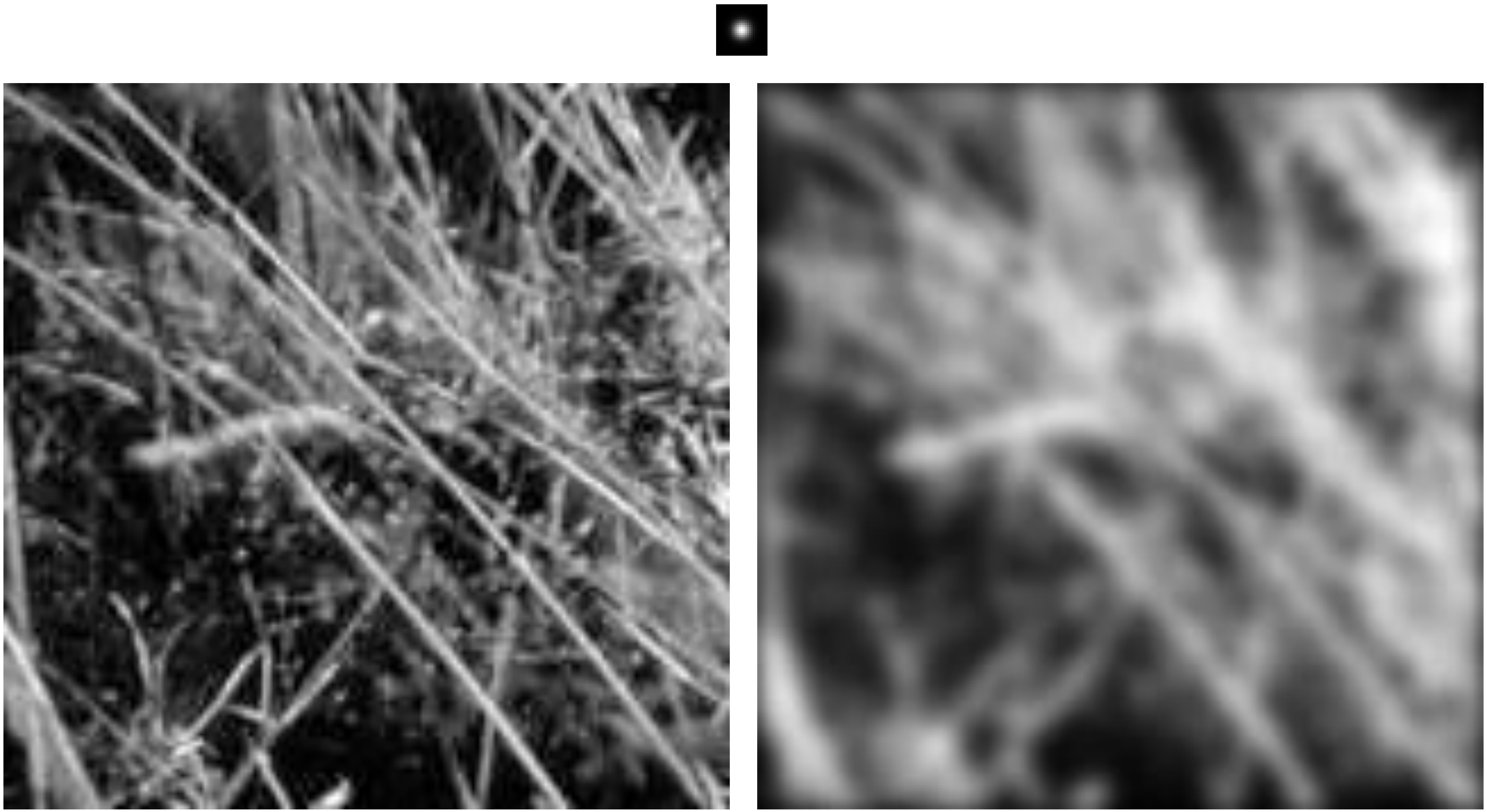
5 x 5, σ = 1

$$G_\sigma = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

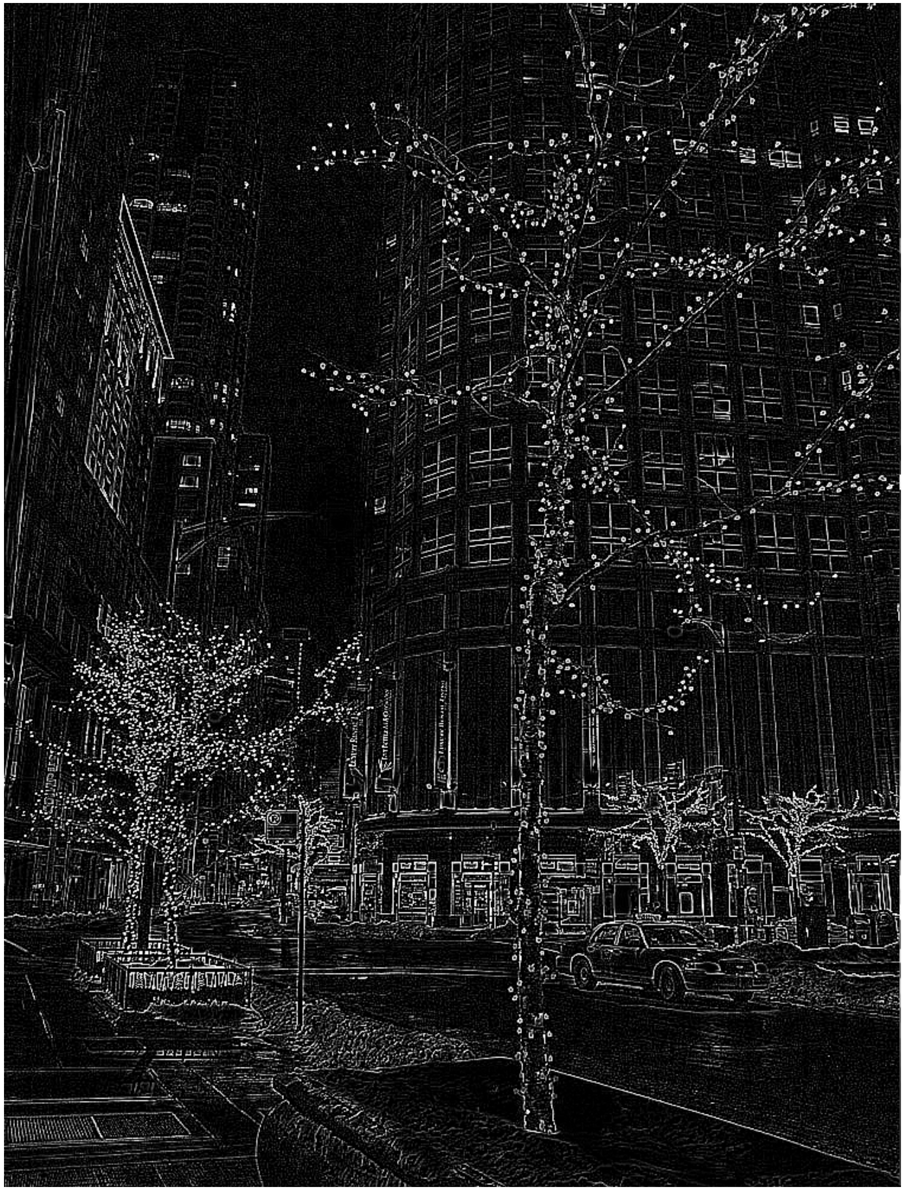
- Constant factor at front makes volume sum to 1 (can be ignored, as we should normalize weights to sum to 1 in any case).

Slide credit: Christopher Rasmussen

Smoothing with a Gaussian



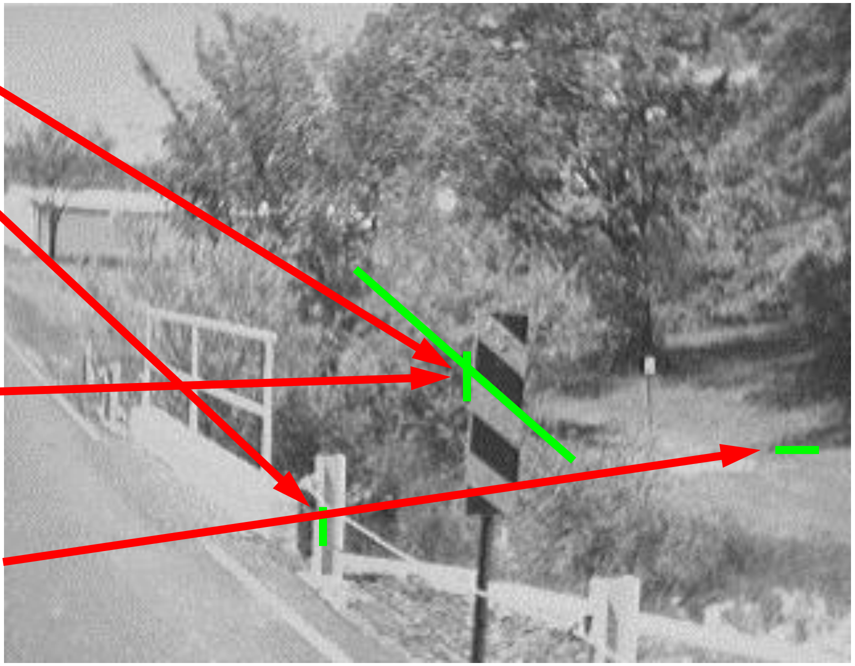
Edge Detection



What causes an edge?

Identifies sudden changes in an image

- Depth discontinuity
- Surface orientation discontinuity
- Reflectance discontinuity (i.e., change in surface material properties)
- Illumination discontinuity (e.g., shadow)



Edge Detection

- Criteria for optimal edge detection (Canny 86):

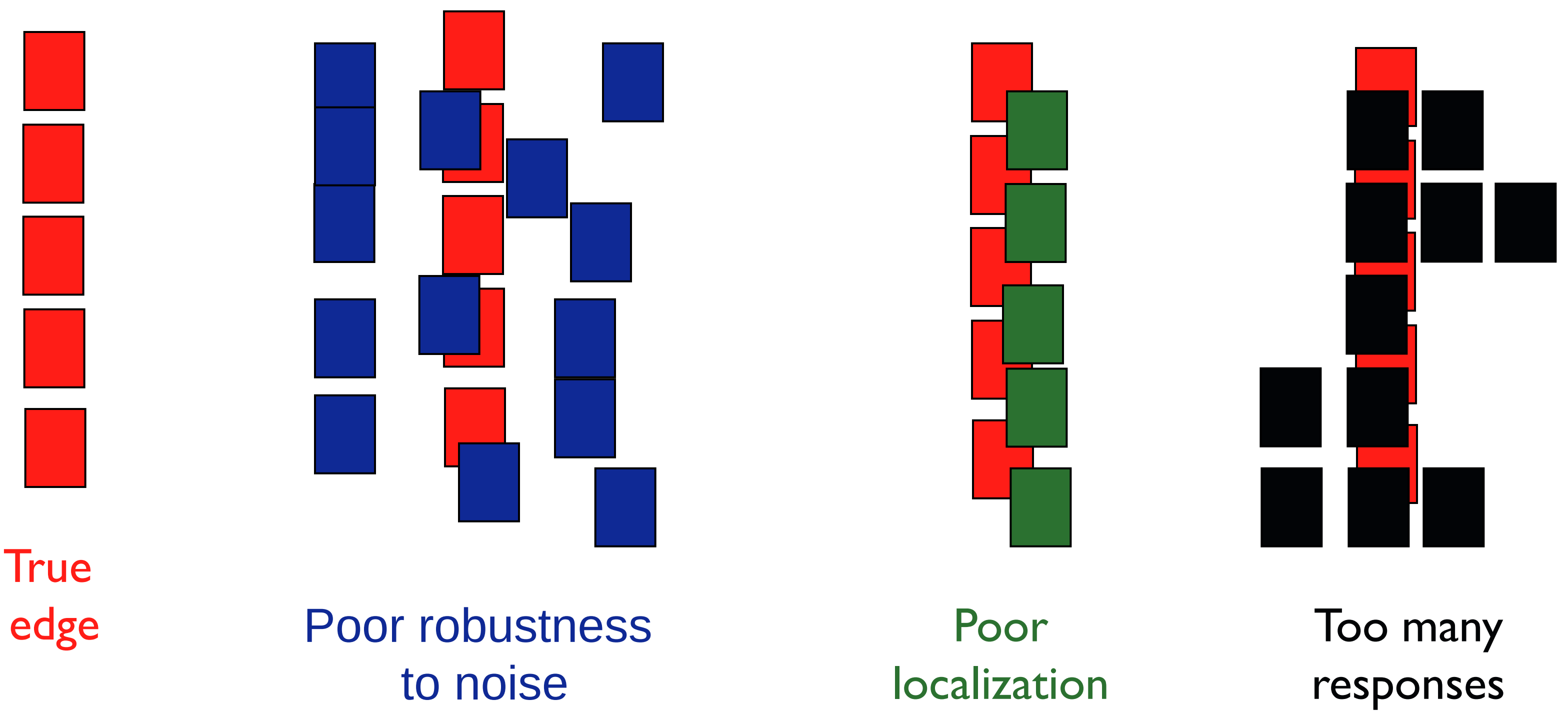
- Good detection accuracy:
  - minimize the probability of false positives (detecting spurious edges caused by noise),
  - false negatives (missing real edges)

- Good localization:
  - edges must be detected as close as possible to the true edges.

- Single response constraint:
  - minimize the number of local maxima around the true edge
  - (i.e. detector must return single point for each true edge point)

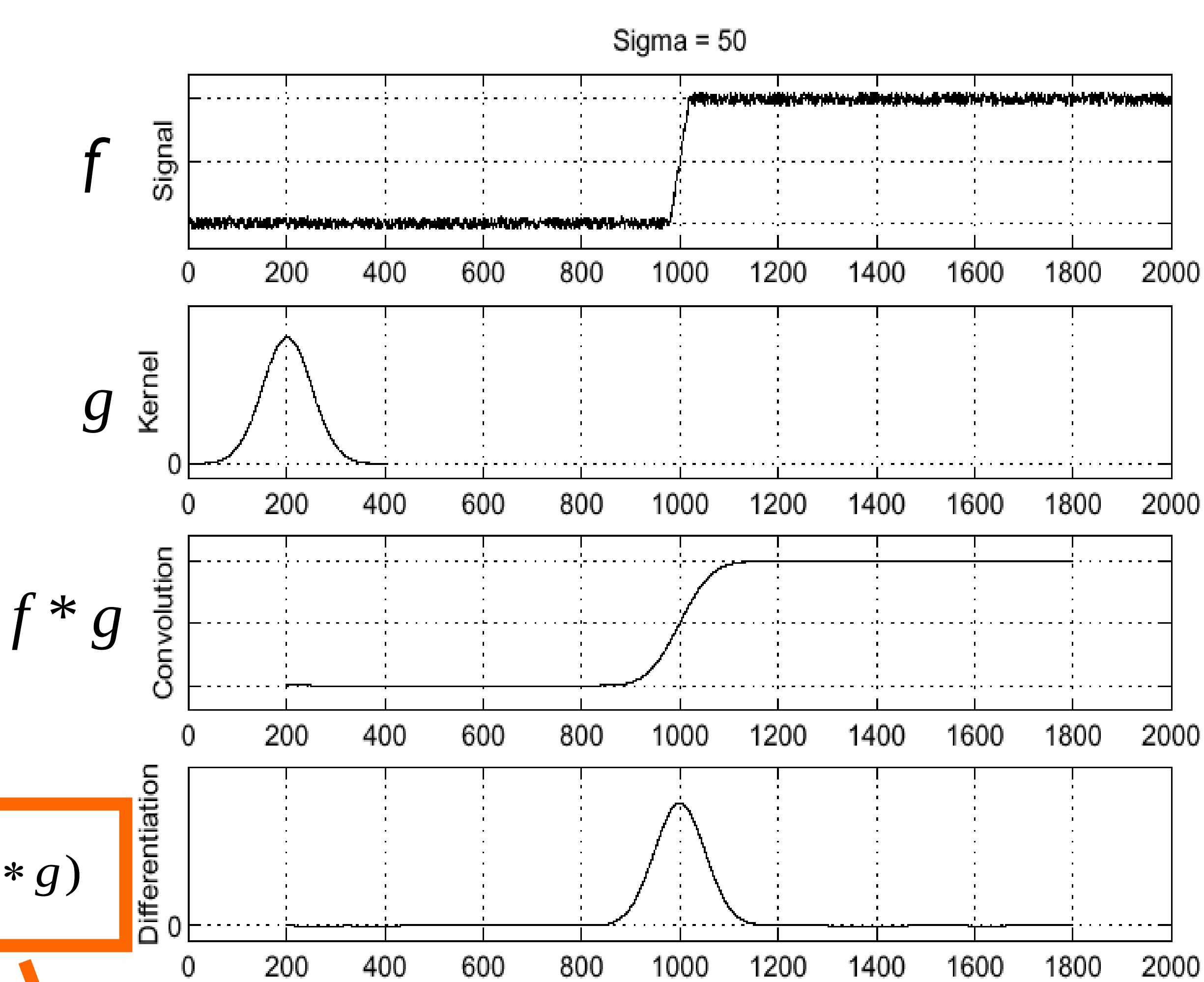
Edge Detection

- Examples:



Designing an edge detector

- **Edge:** a location with high gradient
  (thus, use derivatives)
- Need two derivatives, in x and y direction.
- Need smoothing to reduce noise prior to taking derivative



$$\frac{d}{dx}(f * g)$$

"derivative of Gaussian" filter

Source: S. Seitz

Canny Edge Detection

- Most widely used edge detector in computer vision.
- First derivative of the Gaussian closely approximates the operator that optimizes the product of *signal-to-noise* ratio and localization.

Canny Edge Detection

Steps:

1. Gaussian smoothing
2. & Derivative = Derivative of Gaussian
3. Find magnitude and orientation of gradient
4. Extract edge points: 'Non-maximum suppression'
5. Linking and thresholding 'Hysteresis':

- Matlab: `edge(I, 'canny')`

Canny Edge Detector- First 2 Steps

- Smoothing

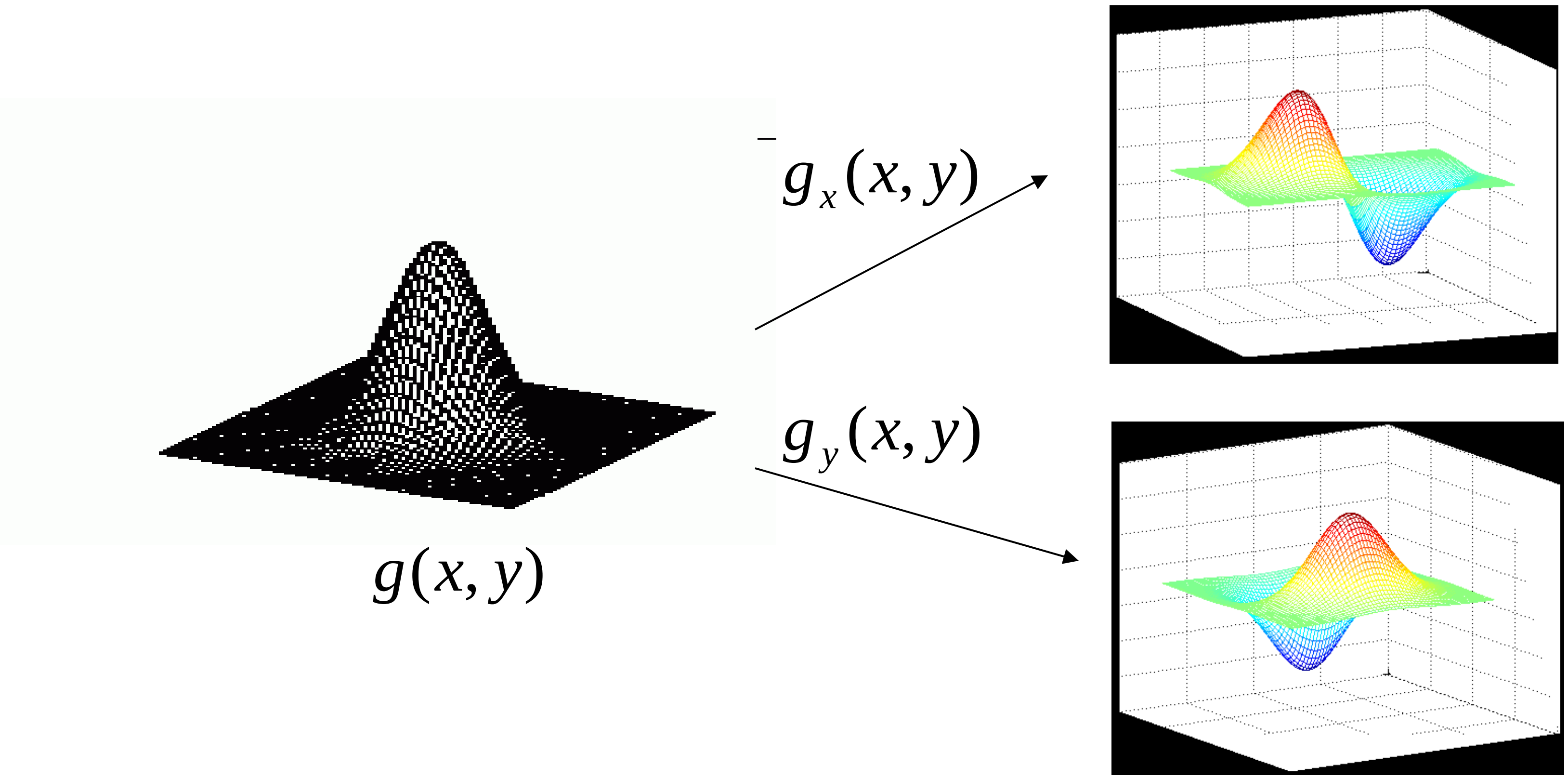
$$I' = g(x, y) * I \qquad g(x,y) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

- Derivative

$$S = \nabla(g * I) = (\nabla g) * I =$$

$$= \begin{bmatrix} g_x \\ g_y \end{bmatrix} * I = \begin{bmatrix} g_x * I \\ g_y * I \end{bmatrix} \qquad \nabla g = \begin{bmatrix} \frac{\partial g}{\partial x} \\ \frac{\partial g}{\partial y} \end{bmatrix} = \begin{bmatrix} g_x \\ g_y \end{bmatrix}$$

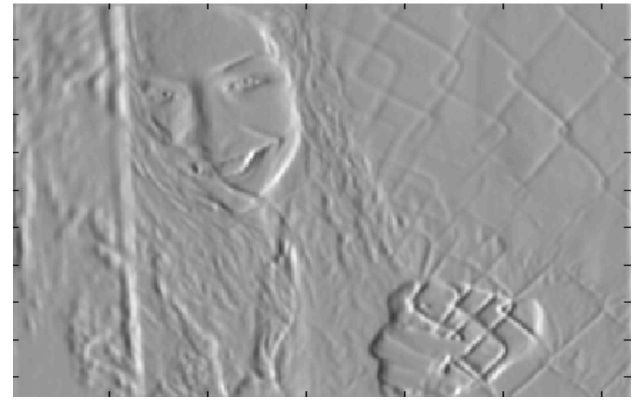
Canny Edge Detector
Derivative of Gaussian



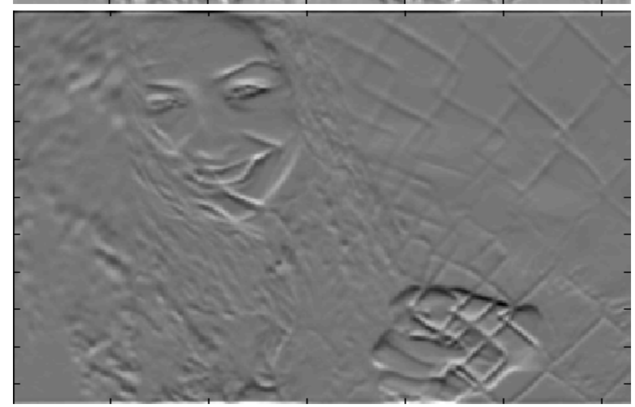
Canny Edge Detector- First 2 Steps

I

S_x

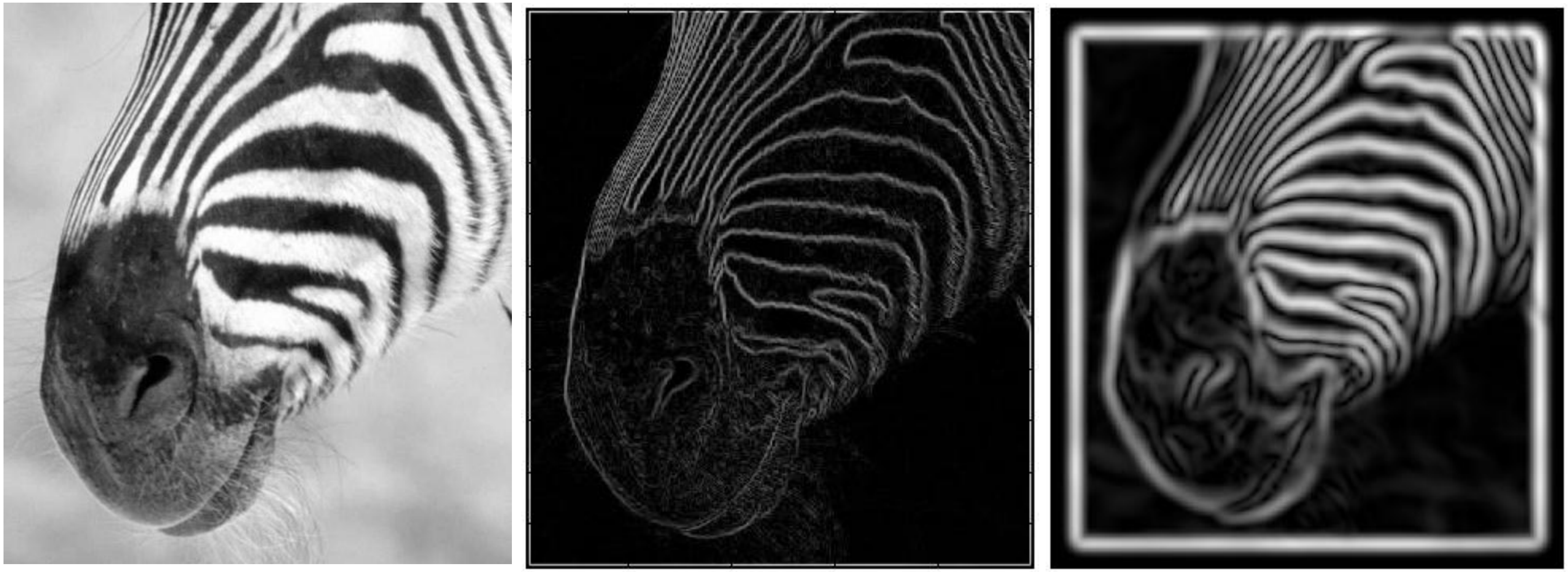


S_y



$$S = \nabla(g * I) = (\nabla g) * I$$

$$S = \begin{bmatrix} S_x & S_y \end{bmatrix}$$ = gradient vector



Increased smoothing:

- Eliminates noise edges.
- Makes edges smoother and thicker.
- Removes fine detail.

Canny Edge Detector- Third Step

- magnitude and direction of

$$S = [S_x \quad S_y]$$

$$\text{magnitude} = \sqrt{(S_x^2 + S_y^2)}$$

$$\text{direction} = \theta = \tan^{-1} \frac{S_y}{S_x}$$

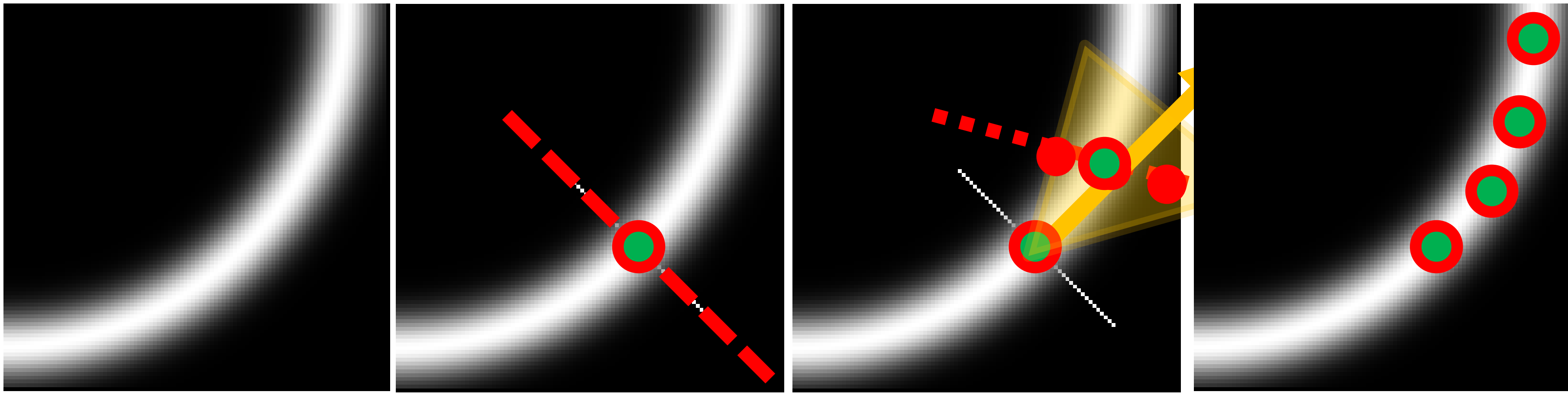
image

gradient magnitude

Canny Edge Detector - Fourth Step

Non maximum suppression

Canny Edge Detector - Fourth Step



1. Initialize:
   - Slice gradient magnitude along the gradient direction
   - Mark the point along the slide where the magnitude is max
2. Propagate chain from current point:
   - Predict next points using the normal to the gradient at that point
   - Find which point is a local max magnitude in gradient direction
   - Retain in magnitude > T

Example: Non-maximum depression

courtesy of G. Loy

Original image

Gradient magnitude

Non-maxima suppressed

Slide credit: Christopher Rasmussen

Canny Edge Detector - Step 5: Thresholding

- Set a threshold T to suppress gradients with magnitude < T

high threshold
(strong edges)

low threshold
(weak edges)

Canny Edge Detector
Step 5: Hysteresis Thresholding

- **Hysteresis**: A lag or momentum factor
- Idea: Maintain two thresholds k_{high} and k_{low}
  - Use k_{high} to find strong edges to start edge chain
  - Use k_{low} to find weak edges along the edge chain
- Typical ratio of thresholds is roughly

$$k_{high} / k_{low} = 2$$

hysteresis threshold

Effect of σ (Gaussian kernel spread/size)

original

Canny with $\sigma = 1$

Canny with $\sigma = 2$

- The choice of σ depends on desired behavior
  - large σ detects large scale edges
  - small σ detects fine features

Source: S. Seitz

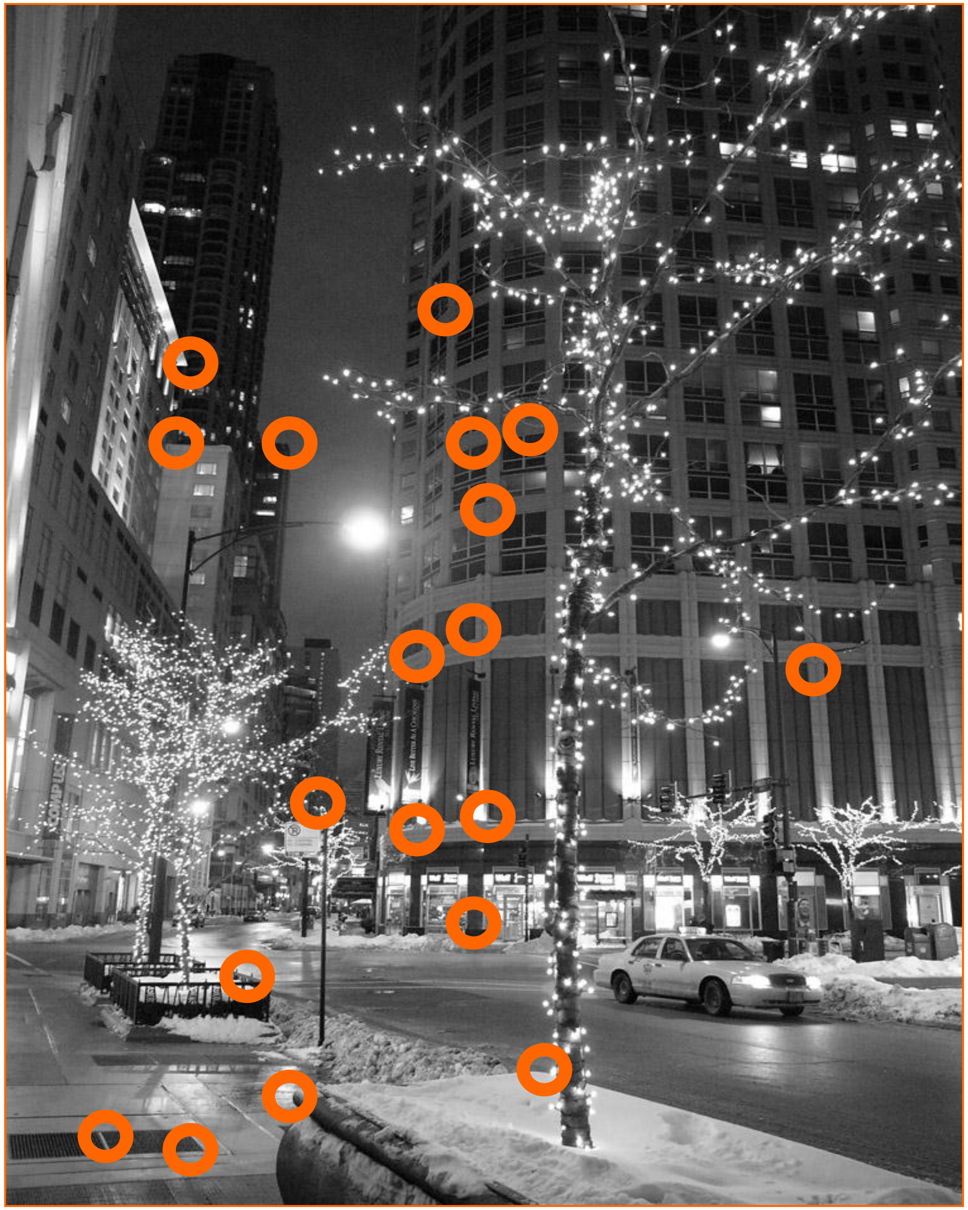
Demo

http://www.cs.washington.edu/research/imagedatabase/demo/edge/

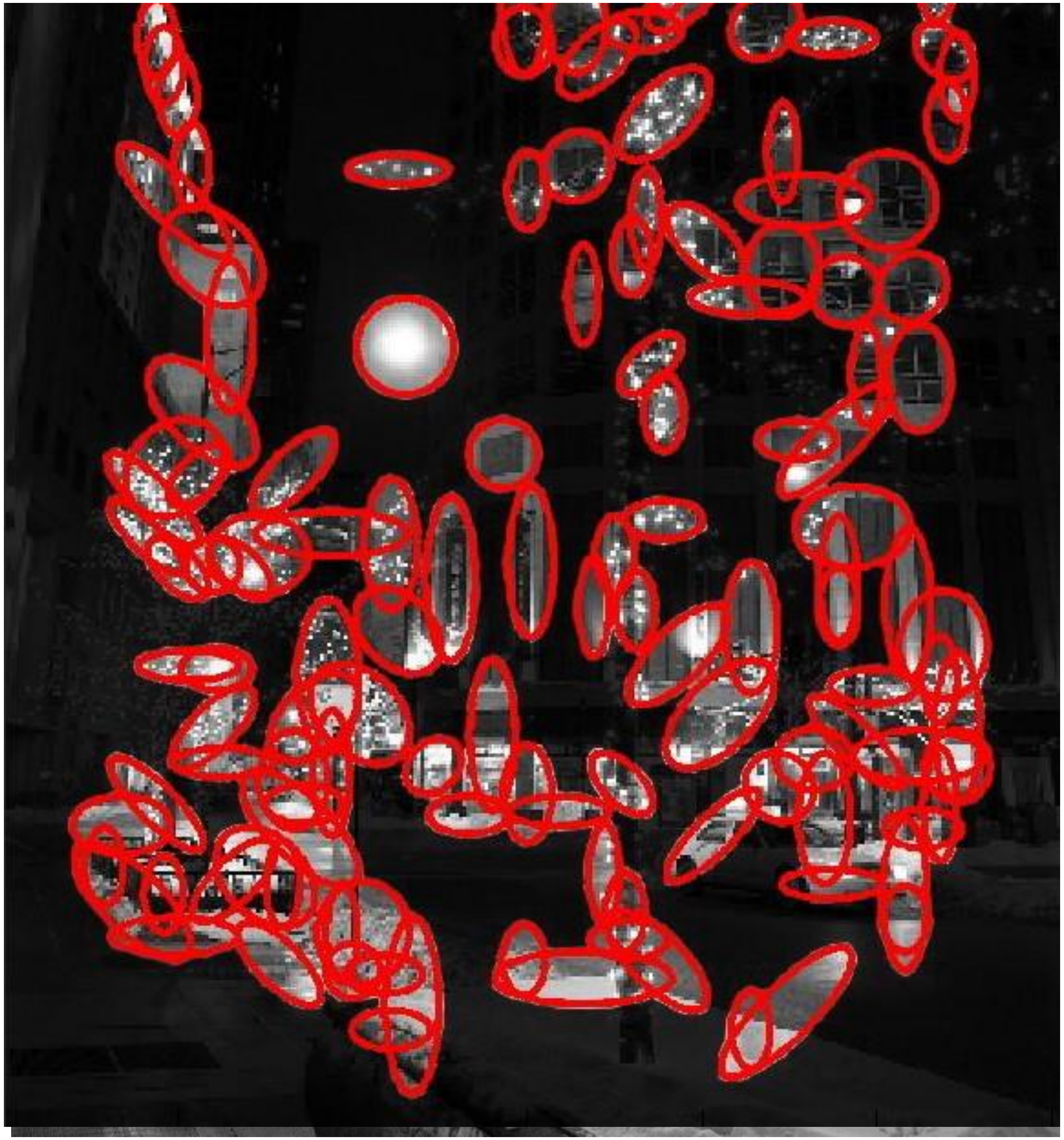
Other edge detectors:

- Sobel
- Canny-Deriche
- Differential

Extract useful building blocks: Corners



Extract useful building blocks: blobs



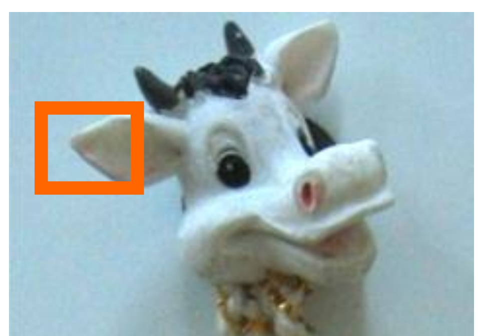
Characteristics

- Repeatability
  - The same feature can be found in several images despite geometric and photometric transformations
- Saliency
  - Each feature is found at an "interesting" region of the image
- Locality
  - A feature occupies a "relatively small" area of the image;

Repeatability

Illumination invariance

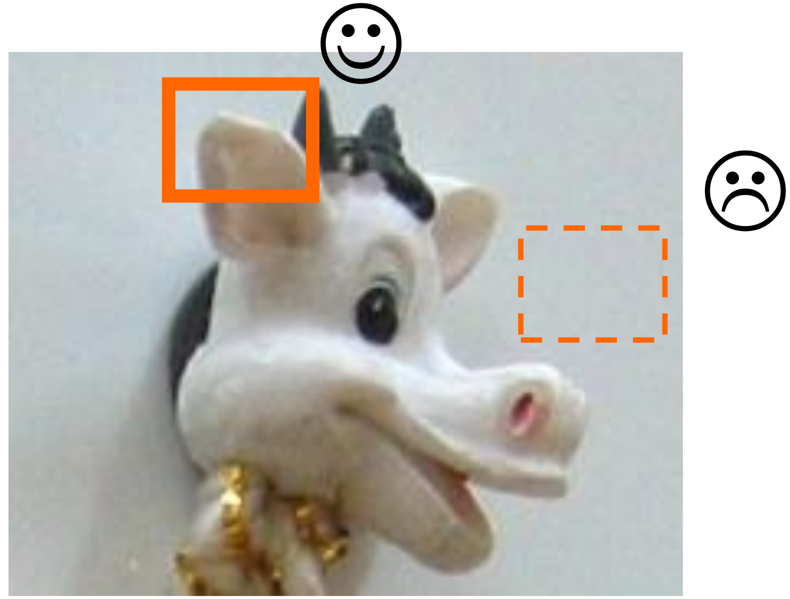
Scale invariance



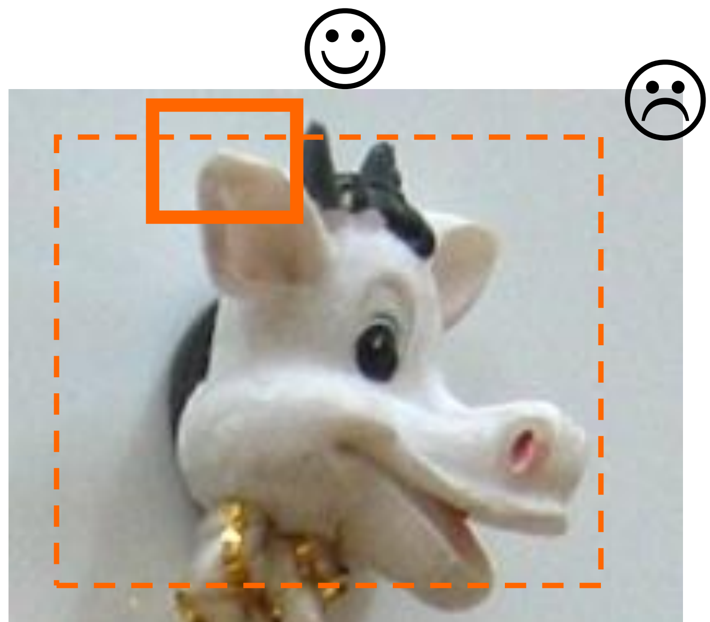
Pose invariance

- Rotation
- Affine

- Saliency

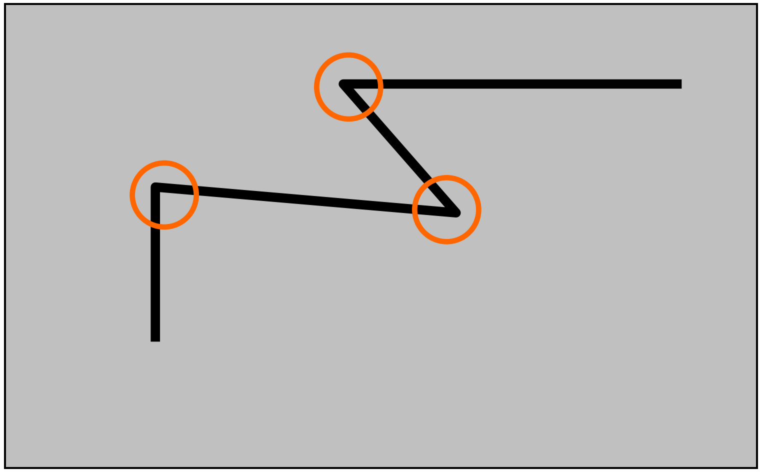


- Locality



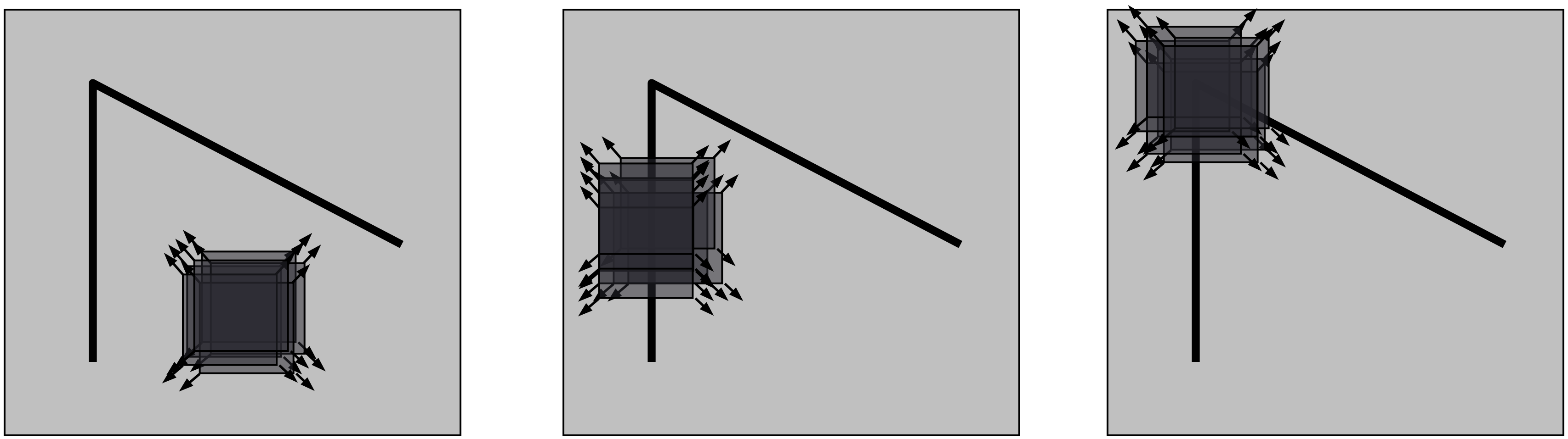
Harris corner detector

- C.Harris and M.Stephens. "A Combined Corner and Edge Detector.“ *Proceedings of the 4th Alvey Vision Conference*: pages 147--151.



Harris Detector: Basic Idea

Explore intensity changes within a window
as the window changes location



"flat" region:
no change in
all directions

"edge":
no change along
the edge direction

"corner":
significant change
in all directions

Harris Detector: Mathematics

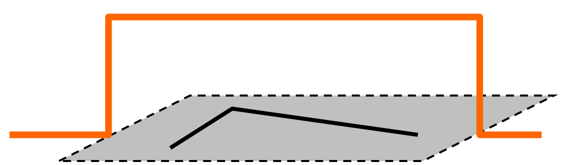
Change of intensity for the shift [*u*,*v*]:

Proportional to the gradient

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

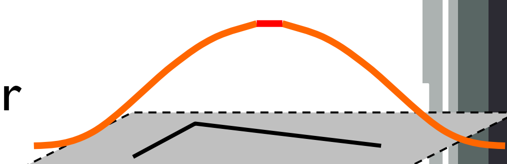
Window function — Shifted intensity — Intensity

Window function w(x,y) =



1 in window, 0 outside

or



Gaussian

Harris Detector: Mathematics

For small shifts $[u,v]$ we have a *bilinear* approximation:

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

where M is a 2×2 matrix computed from image derivatives:

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

Second-moment matrix

Gradient with respect to x, times gradient with respect to y

$$\left(g_x * I\right)\left(g_y * I\right)$$

Sum over a small region around the hypothetical corner (we can omit "w")

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

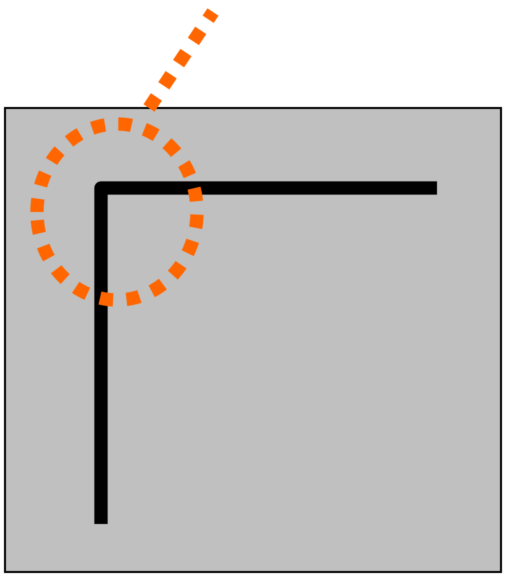
Matrix is symmetric

Slide credit: David Jacobs

Second-moment matrix

First, consider case where dominant gradient directions aligned with x or y

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

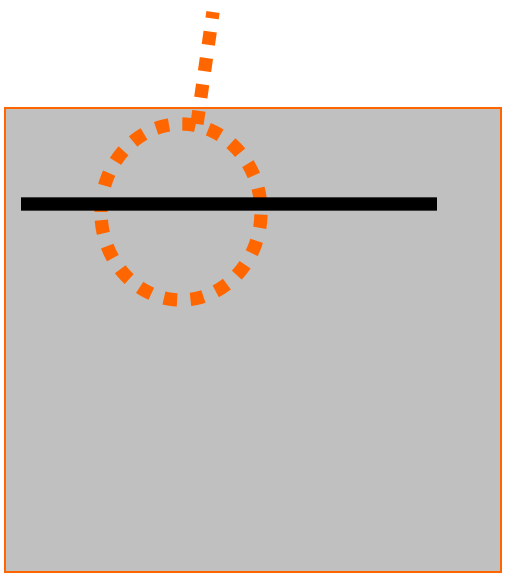


Second-moment matrix

First, consider case where dominant gradient directions aligned with x or y

Structure tensor analyzing the eigenvalues of *A*

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

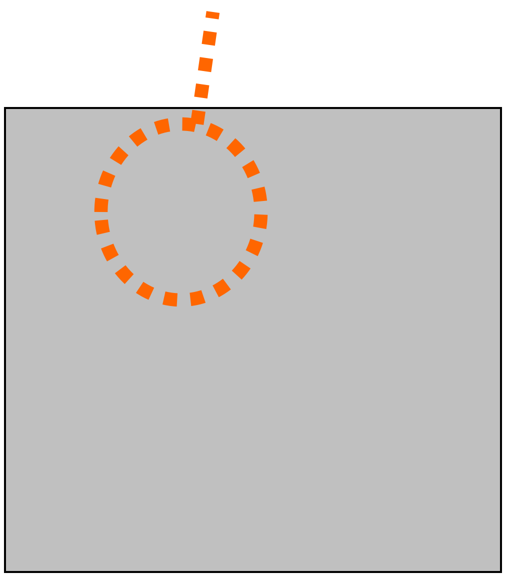


If either λ is close to 0, then this is **an edge**

Second-moment matrix

First, consider case where dominant gradient directions aligned with x or y

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

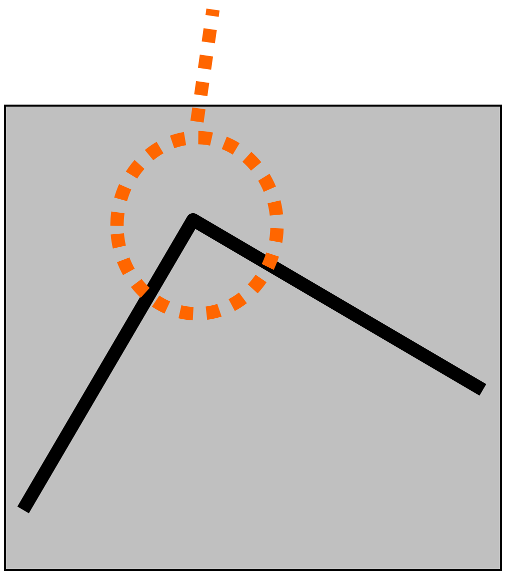


If both λs are close to 0, then this is **a flat region**

Second-moment matrix

For generic window alignments, the eigenvalue decomposition of M returns similar information:

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

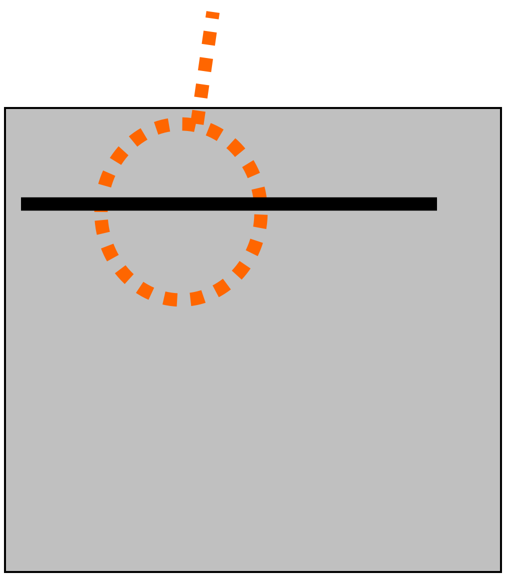


Lambda 1, 2 are the eigenvalues of M

Second-moment matrix

For generic window alignments, the eigenvalue decomposition of M returns similar information:

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

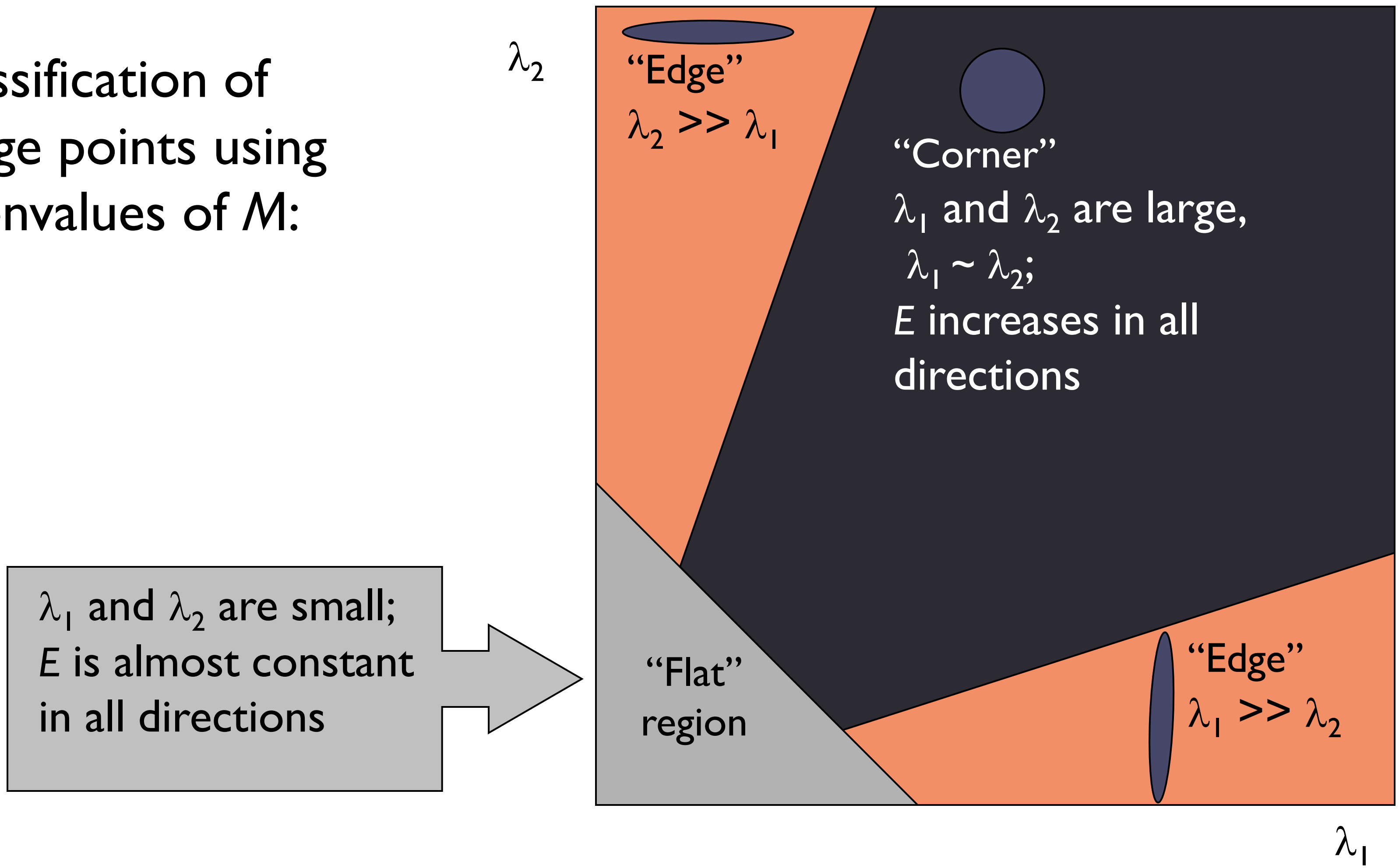


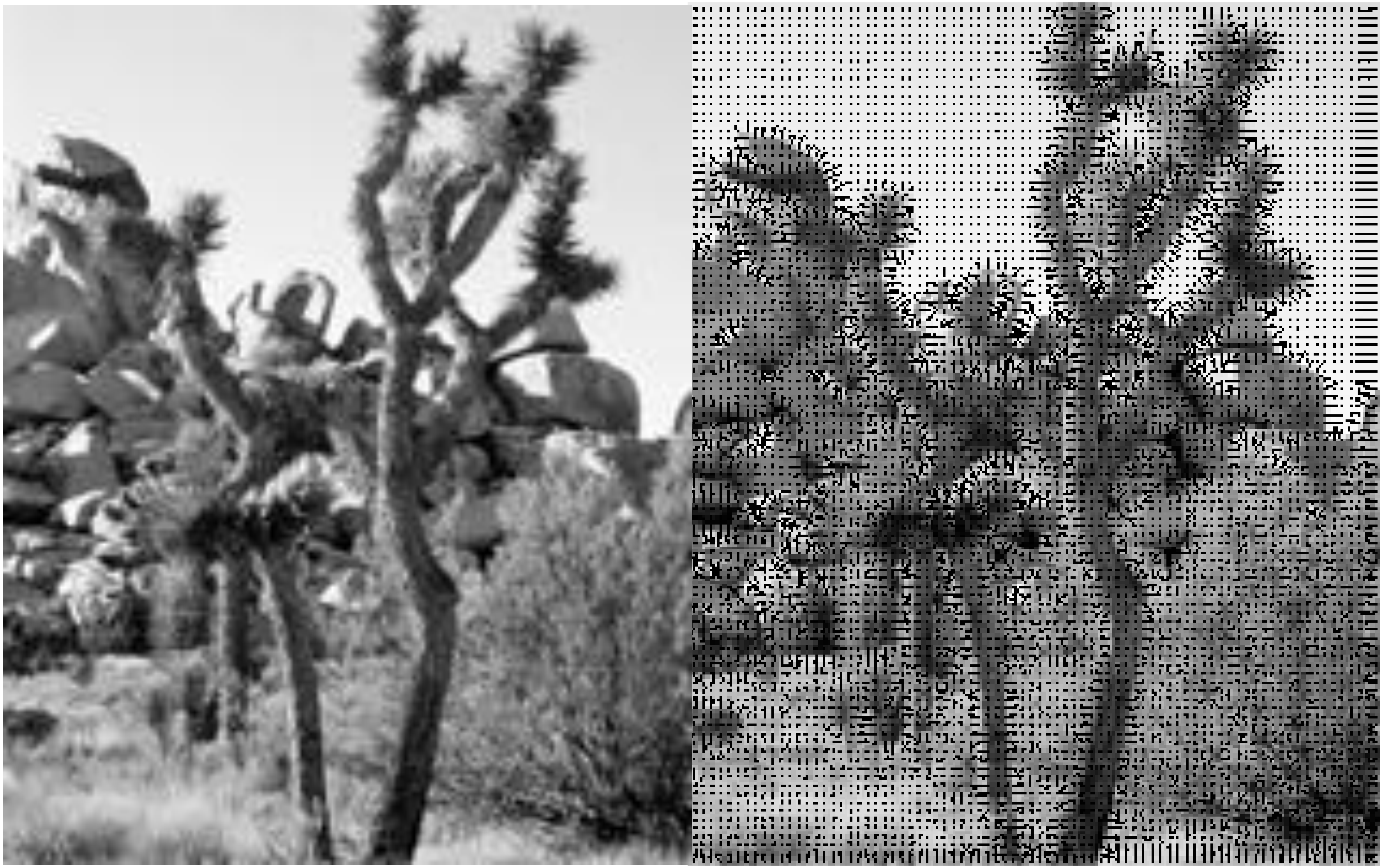
If either λ is close to 0, then this is **an edge**

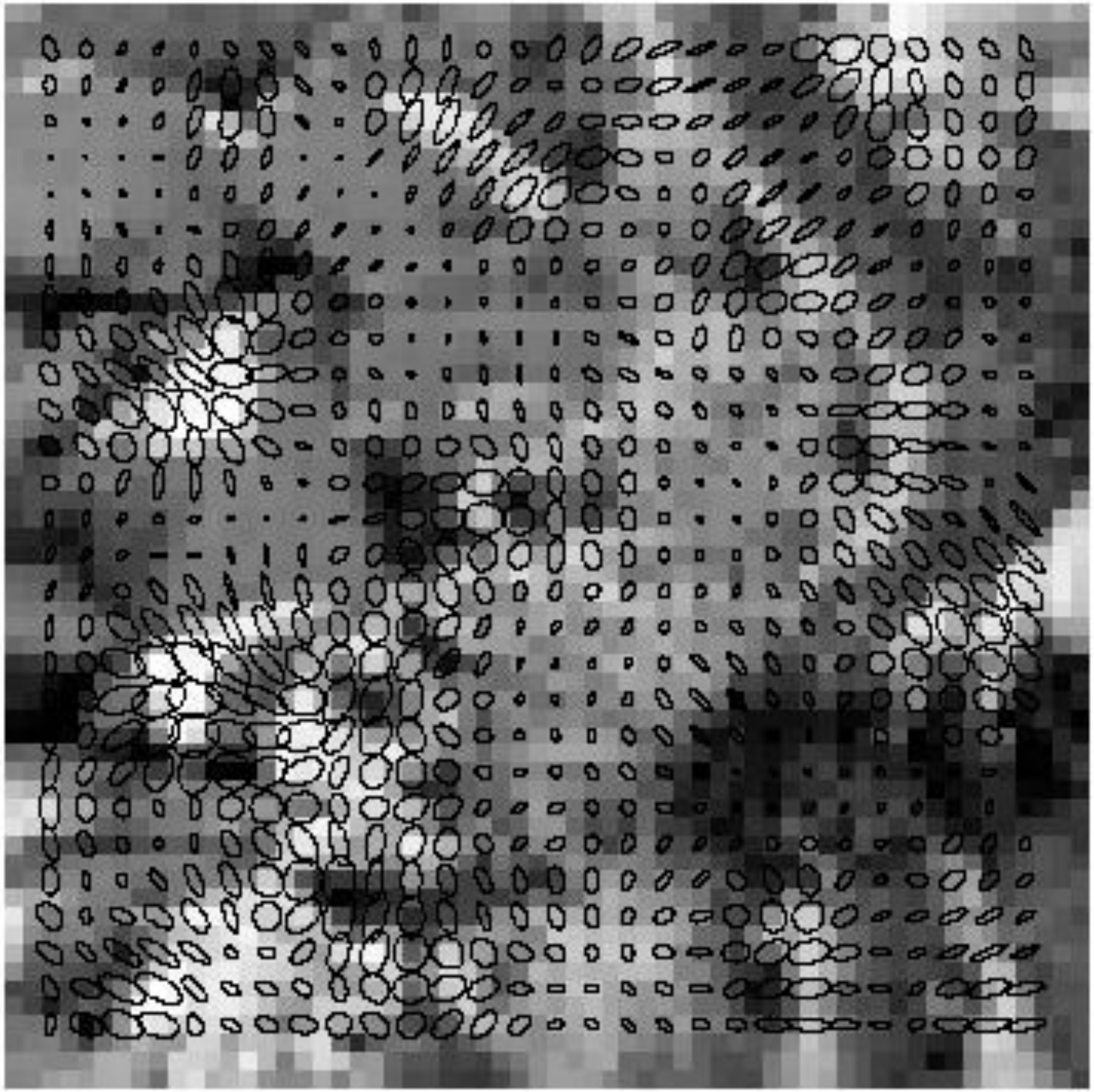
Non-zero eigenvector of M gives direction of the edge

Harris Detector: Mathematics

Classification of image points using eigenvalues of *M*:







Harris Detector: Mathematics

Measure of corner response:

$$R = \det M - k\left(\operatorname{trace} M\right)^2$$

$$\det M = \lambda_1 \lambda_2$$
$$\operatorname{trace} M = \lambda_1 + \lambda_2$$

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

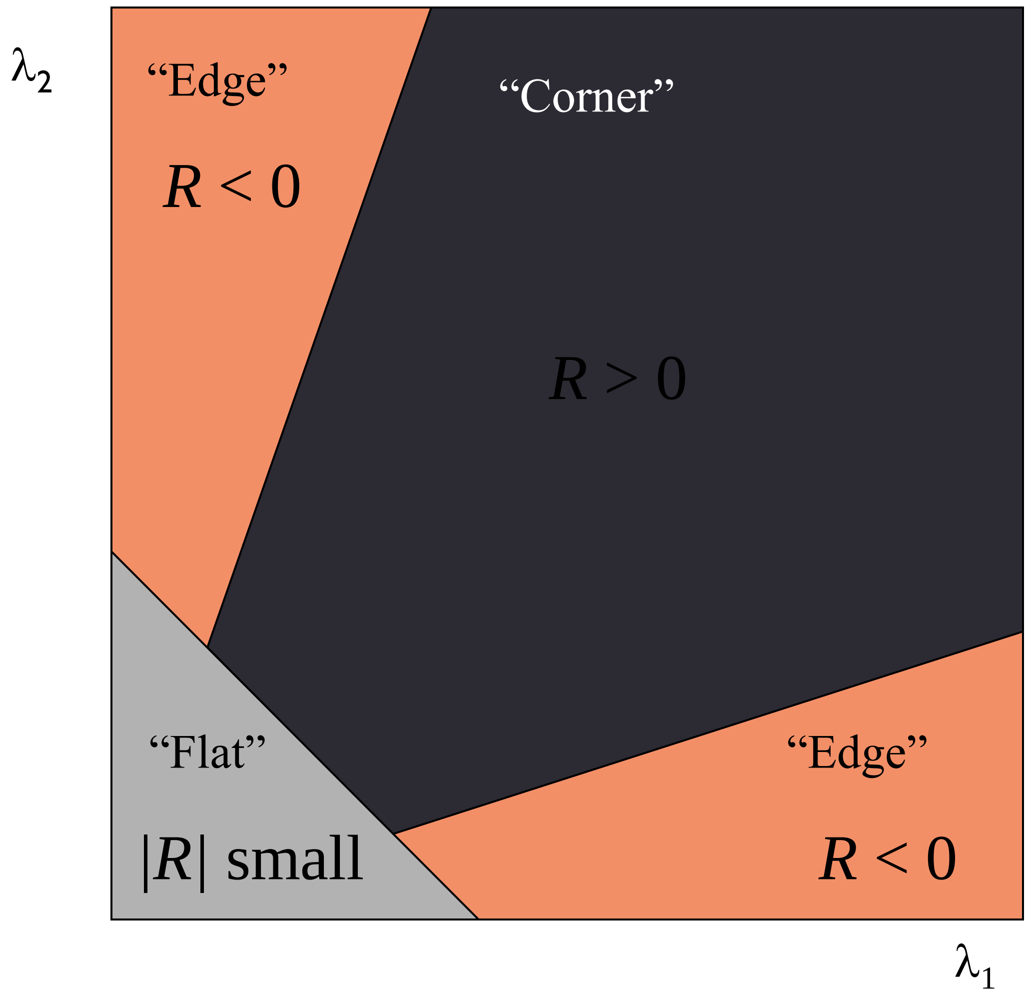
Harris Detector: Algorithm

- 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
- If R>T a corner is detected; retain point of local maxima

$$R = \det M - k\left(\operatorname{trace} M\right)^2$$

Harris Detector: Mathematics

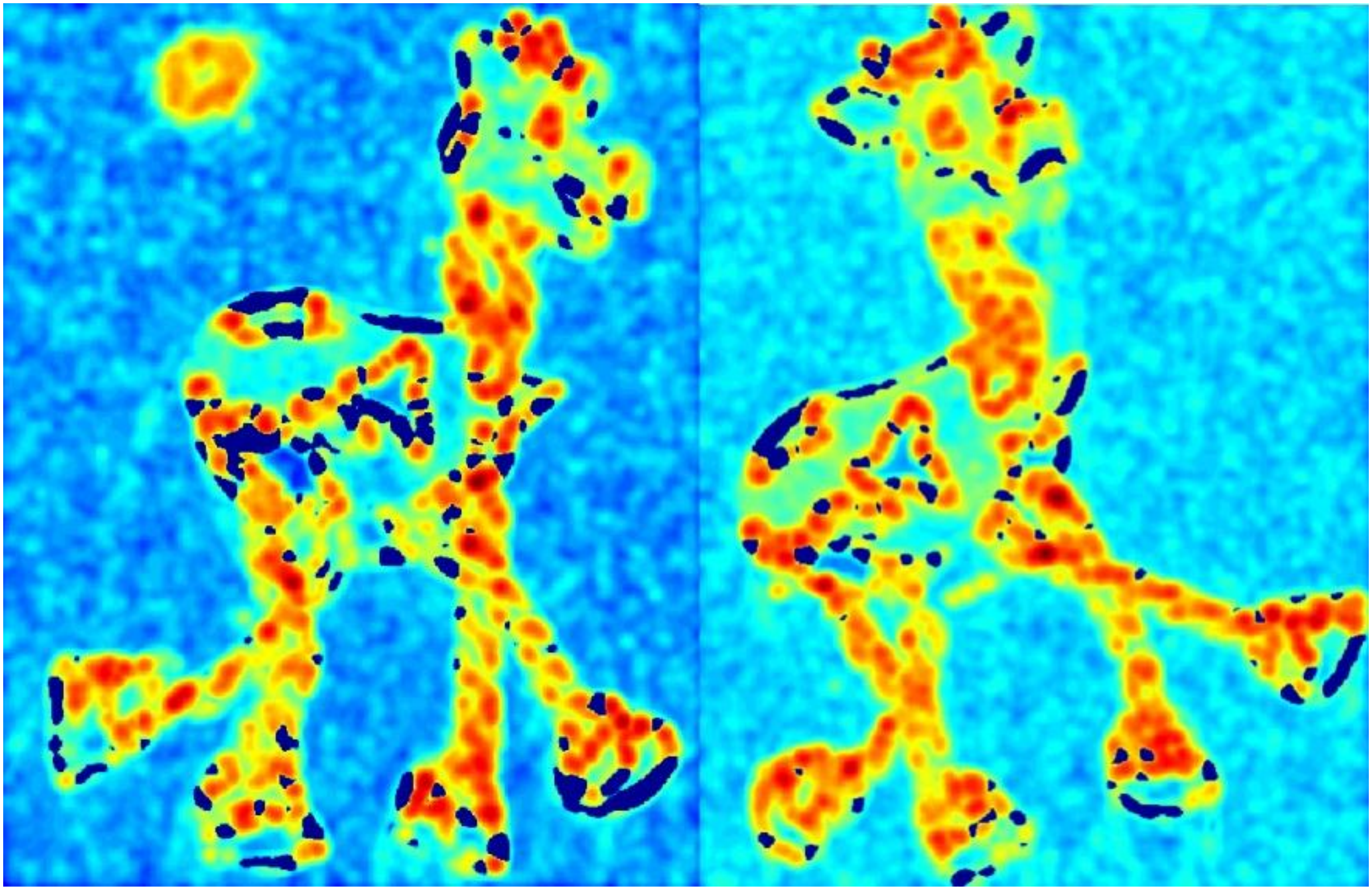
- *R* depends only on eigenvalues of M
- *R* is large for a corner
- *R* is negative with large magnitude for an edge
- |*R*| is small for a flat region



Harris Detector: Workflow

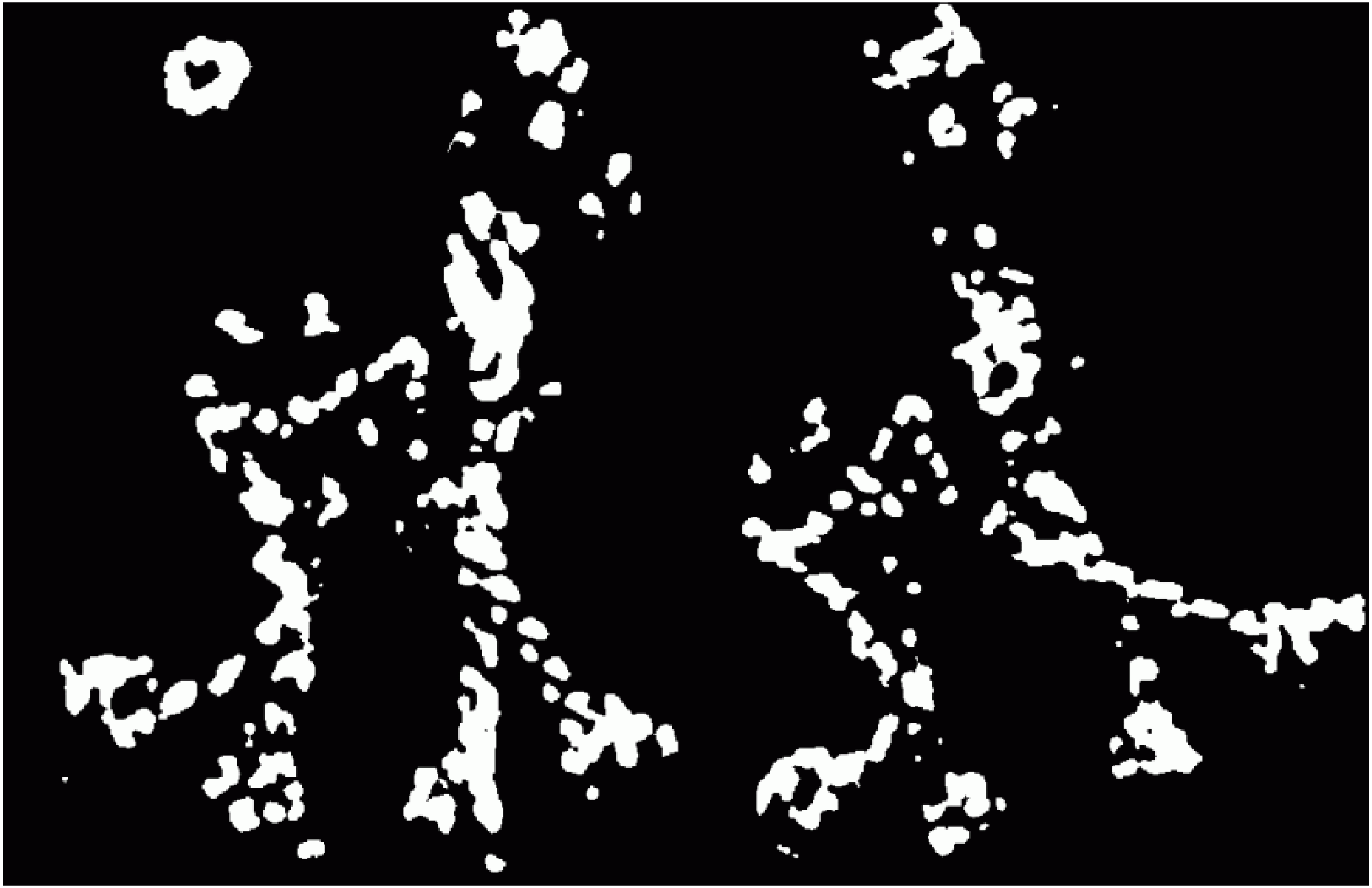
Harris Detector: Workflow

Compute corner response R



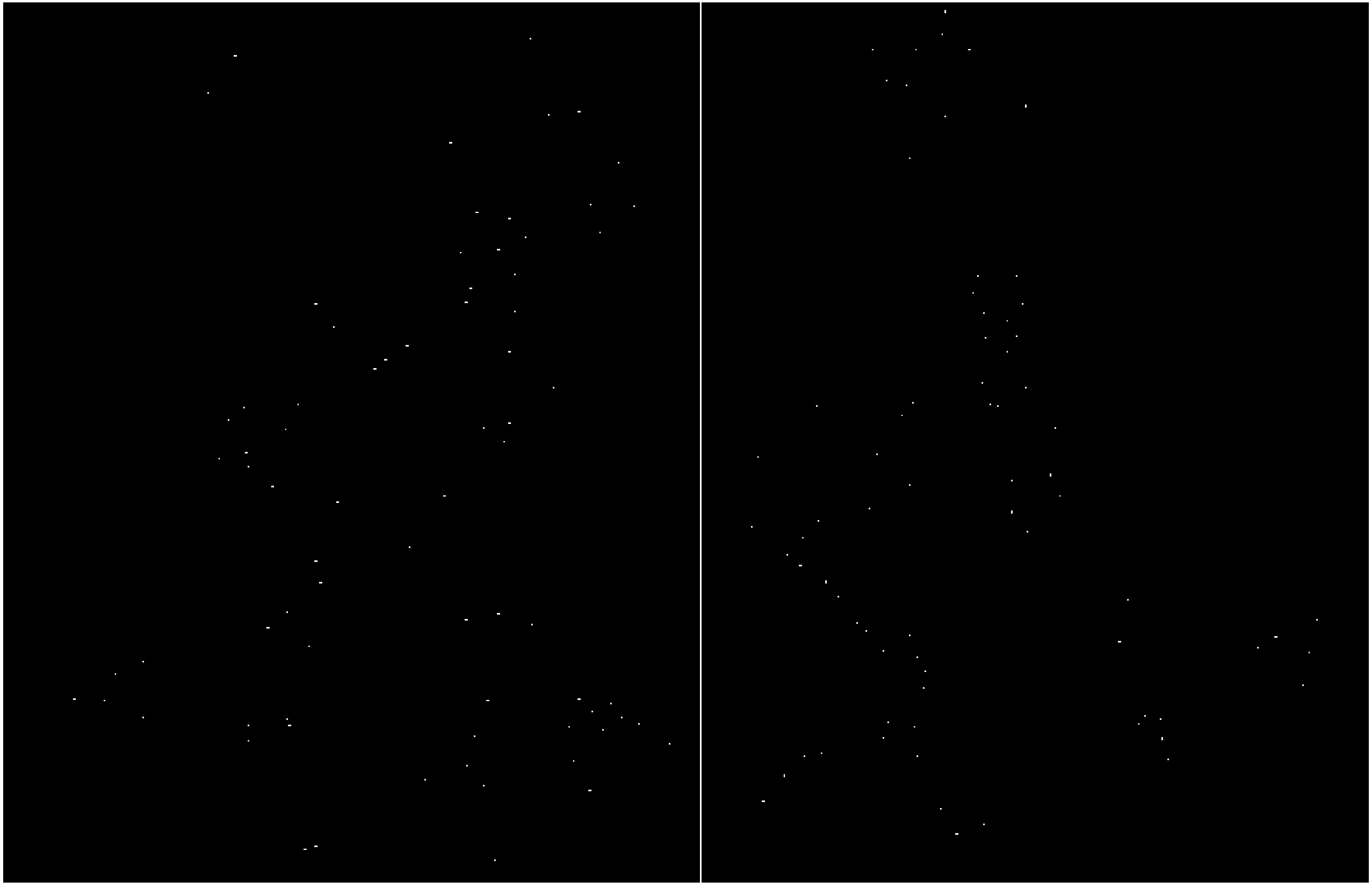
Harris Detector: Workflow

Find points with large corner response: R>threshold



Harris Detector: Workflow

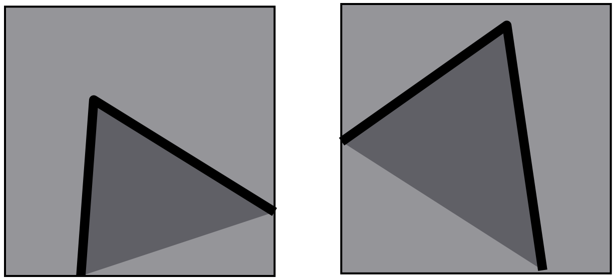
Take only the points of local maxima of R



Harris Detector: Workflow

Harris Detector: Some Properties

- Rotation invariance



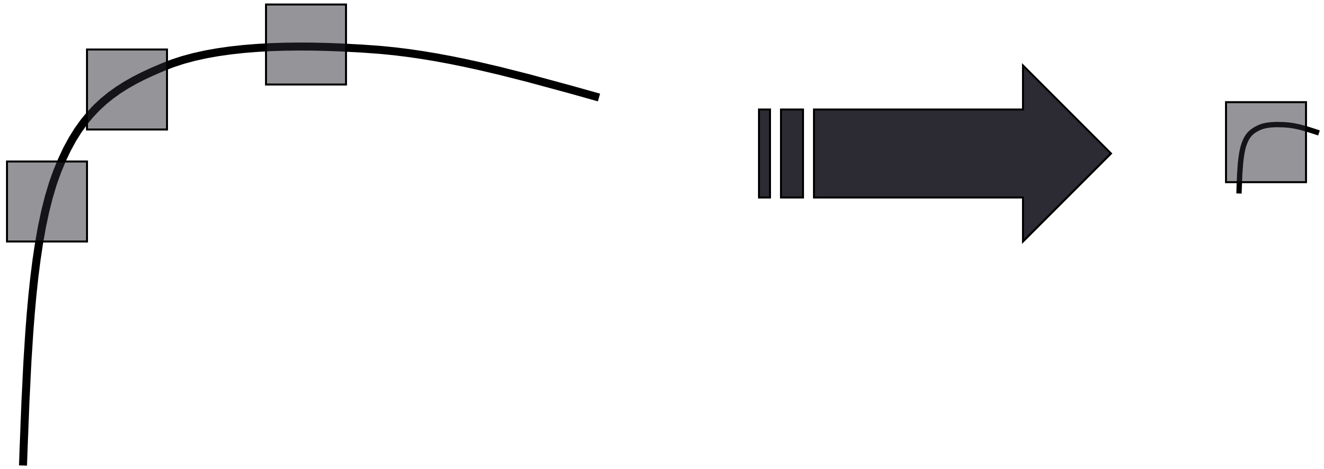
Corner response R is invariant to image rotation

$$C = U^{-1}\begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} U$$

→ R = R(λ_1 , λ_1) doesn't change!

Harris Detector: Some Properties

- But: non-invariant to *image scale*!



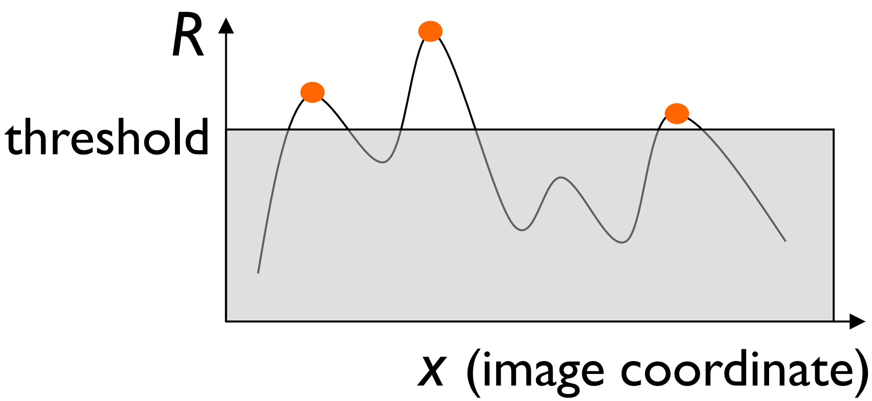
All points will be classified as edges

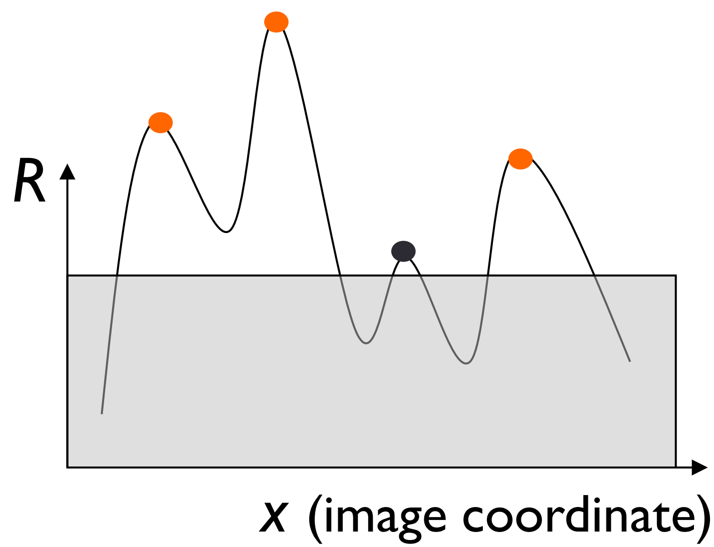
Corner !

Harris Detector: Some Properties

- Partial invariance to *affine intensity* change
  - invariance to intensity shift $I \rightarrow I + b$ *(why?)*

    (only derivatives are used)
  - Intensity scale: $I \rightarrow a\, I$





Next lecture:

- Descriptors
- Detectors part 2