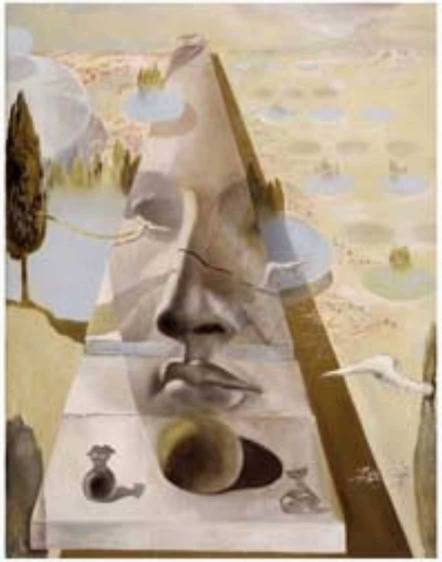
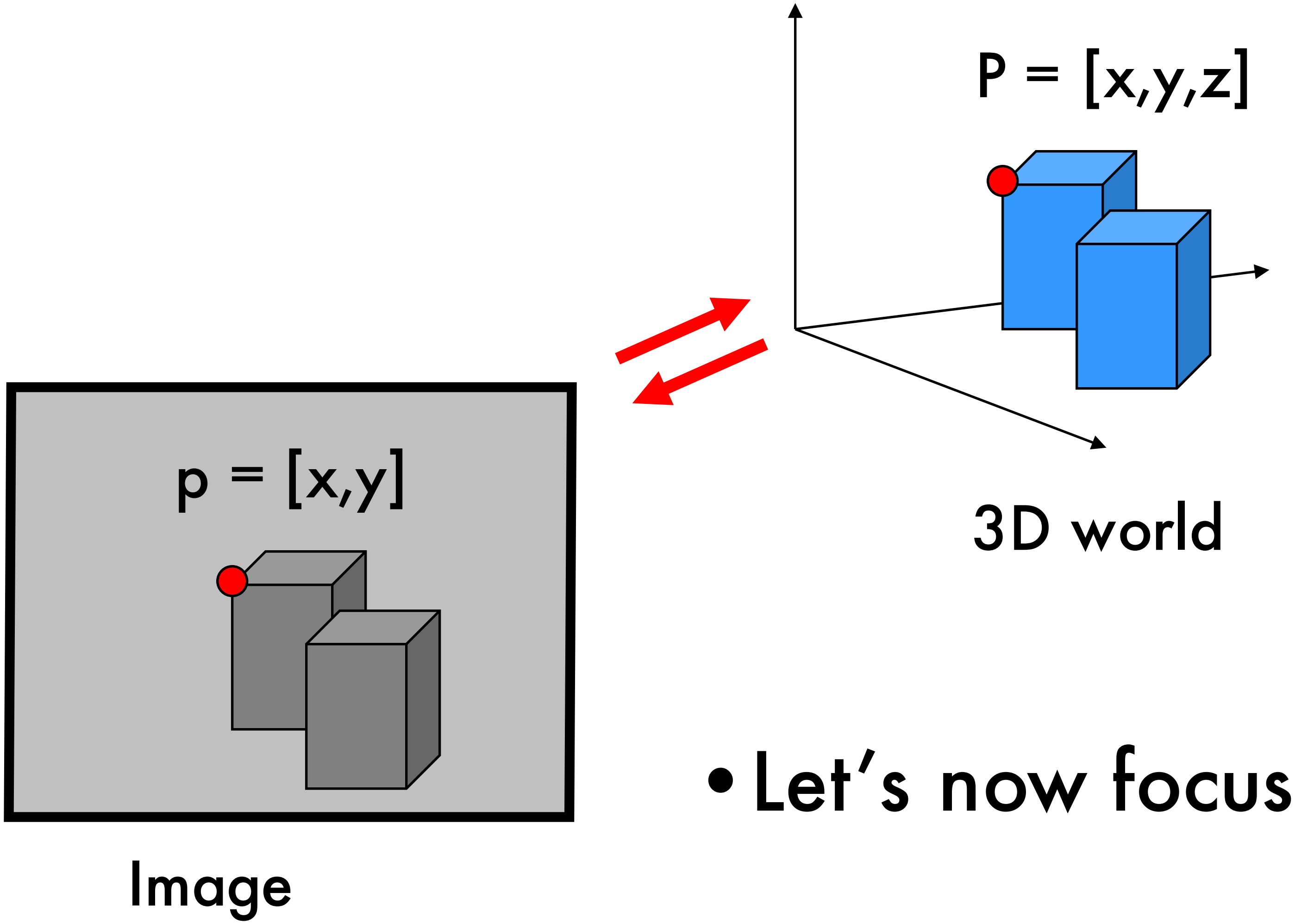
Lecture 9
Detectors and descriptors



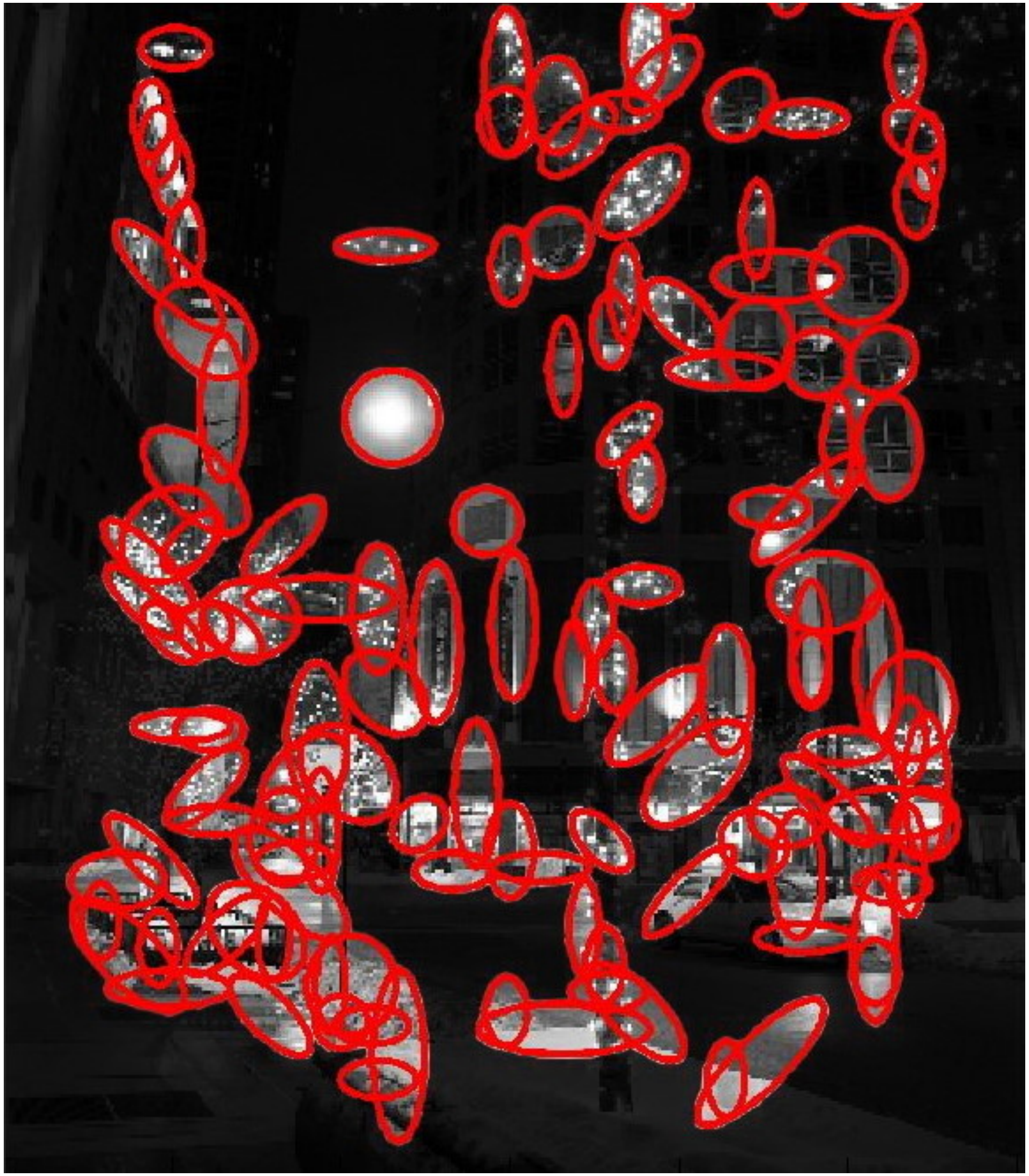
- Properties of detectors
  - Edge detectors
  - Harris
  - DoG
- Properties of descriptors
  - SIFT
  - HOG
  - Shape context

From the 3D to 2D & vice versa



- Let's now focus on 2D

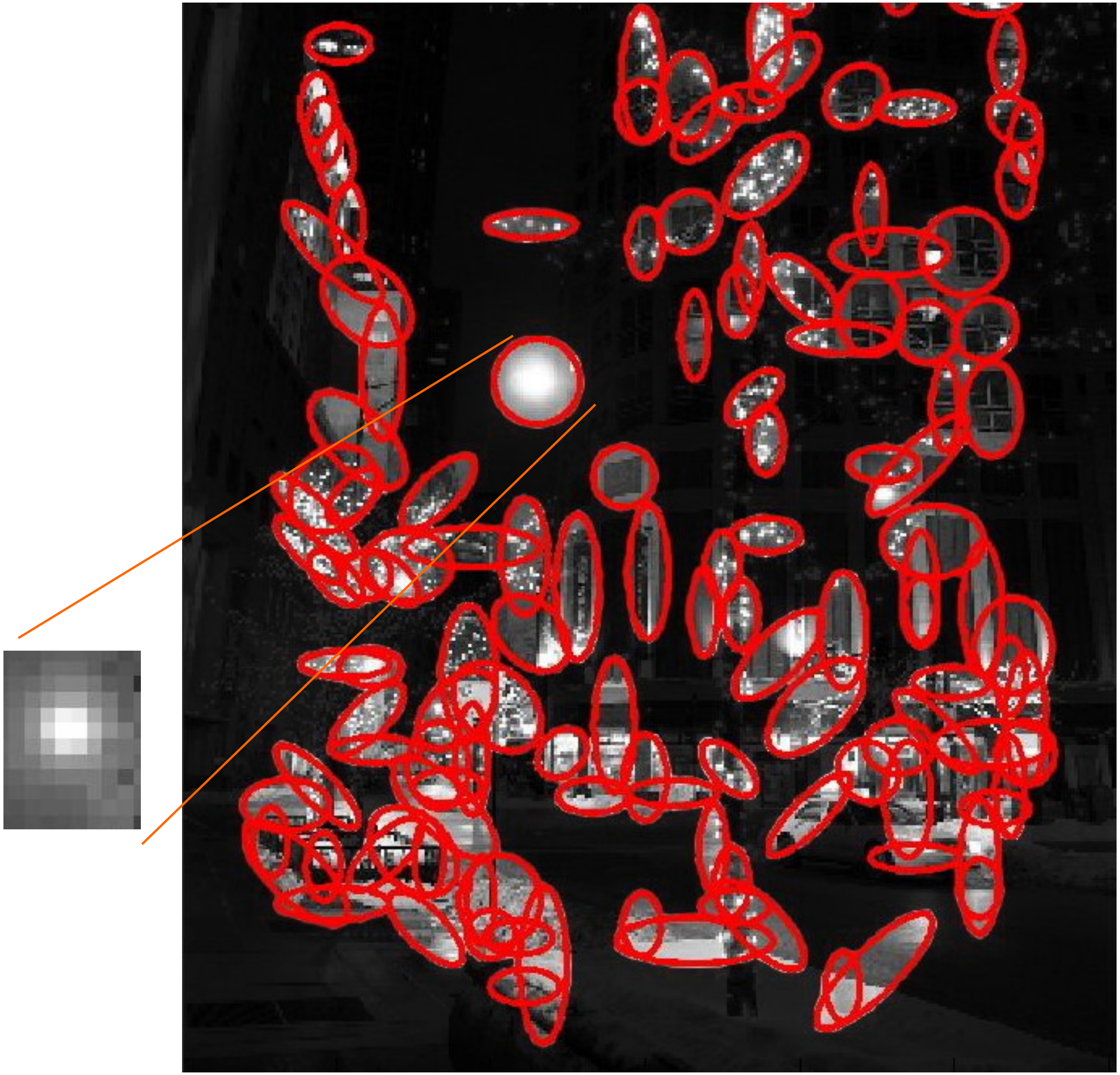
How to represent images?



Feature Detection

e.g. DoG

How to represent images?



Feature Detection

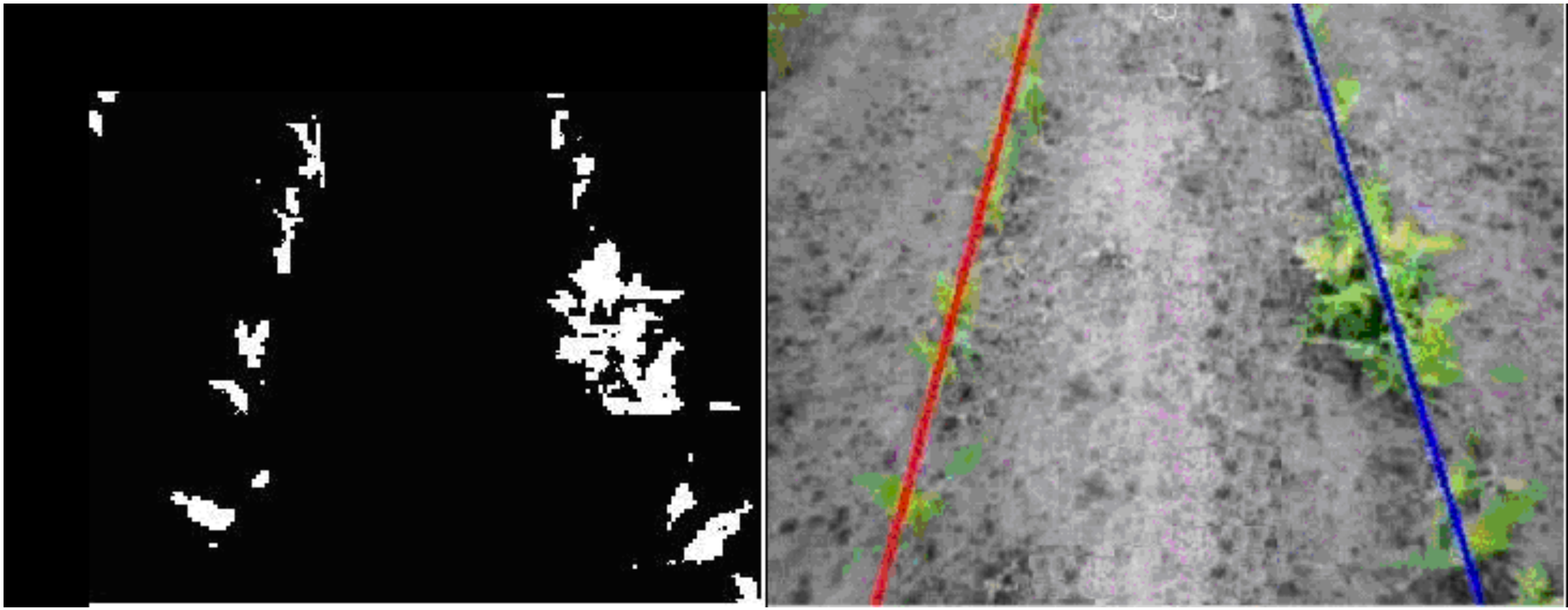
↓ e.g. DoG

Feature Description

↓ e.g. SIFT

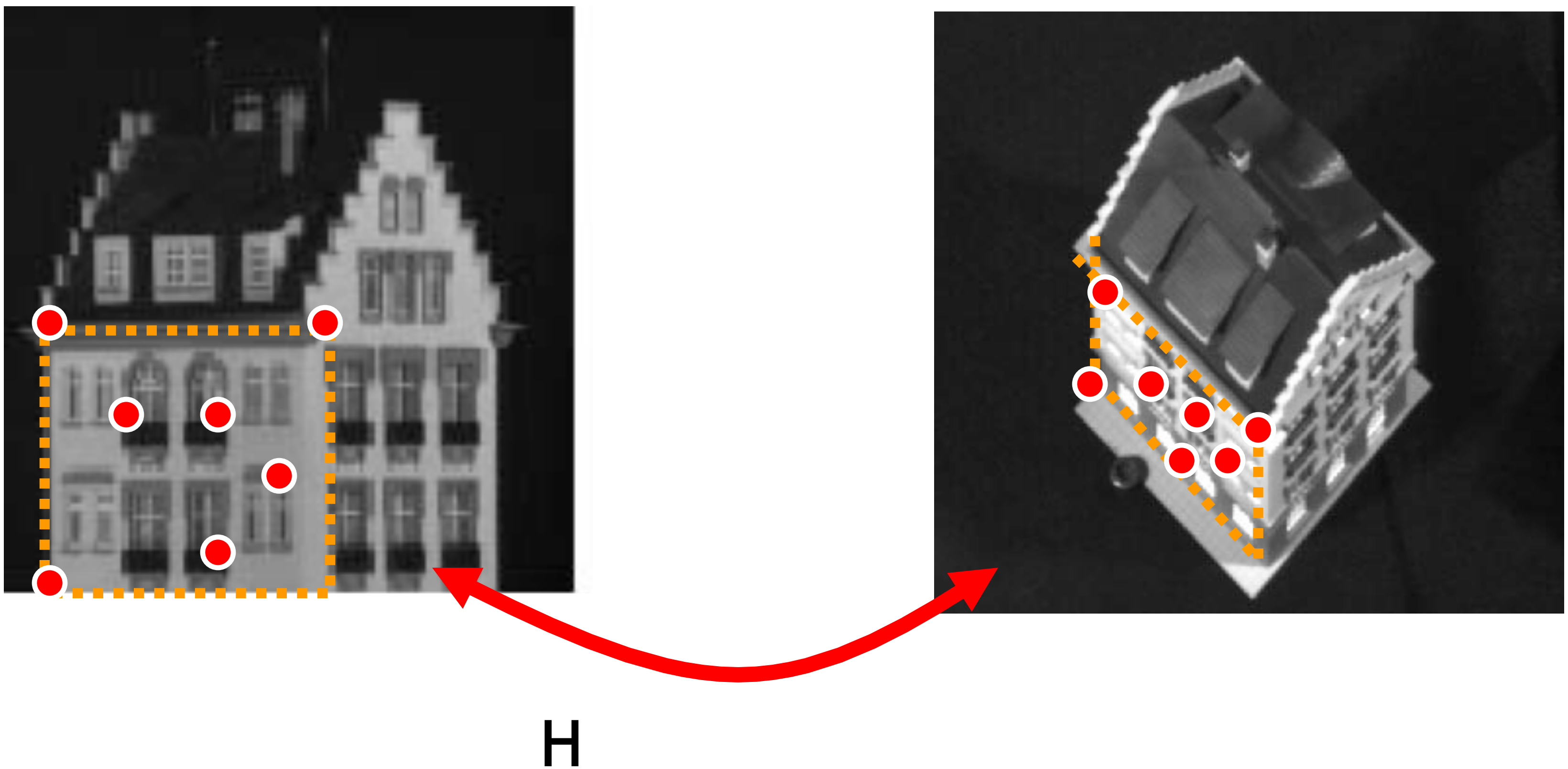
- Estimation
- Matching
- Indexing
- Detection

Estimation

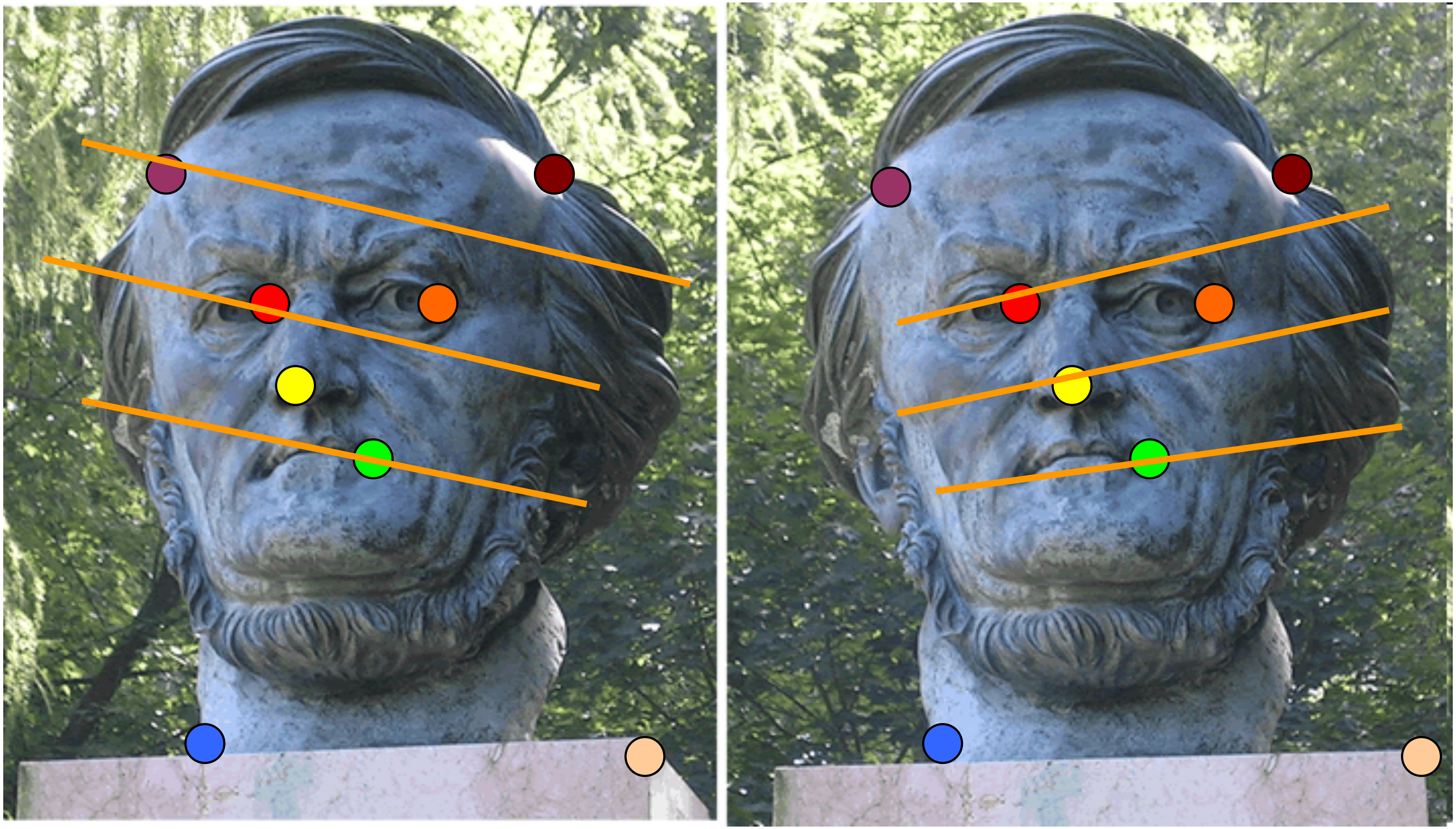


Courtesy of TKK Automation Technology Laboratory

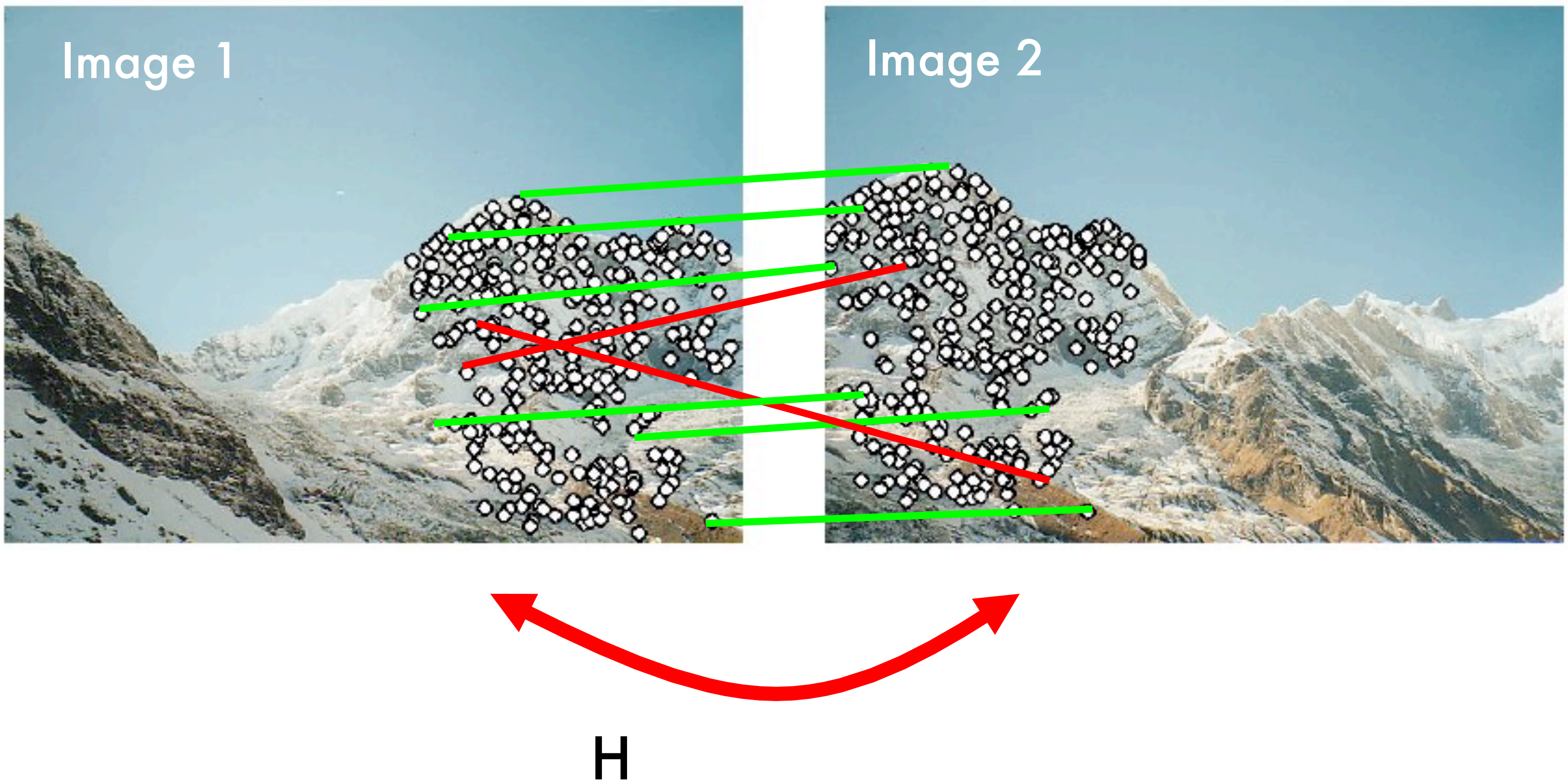
Estimation



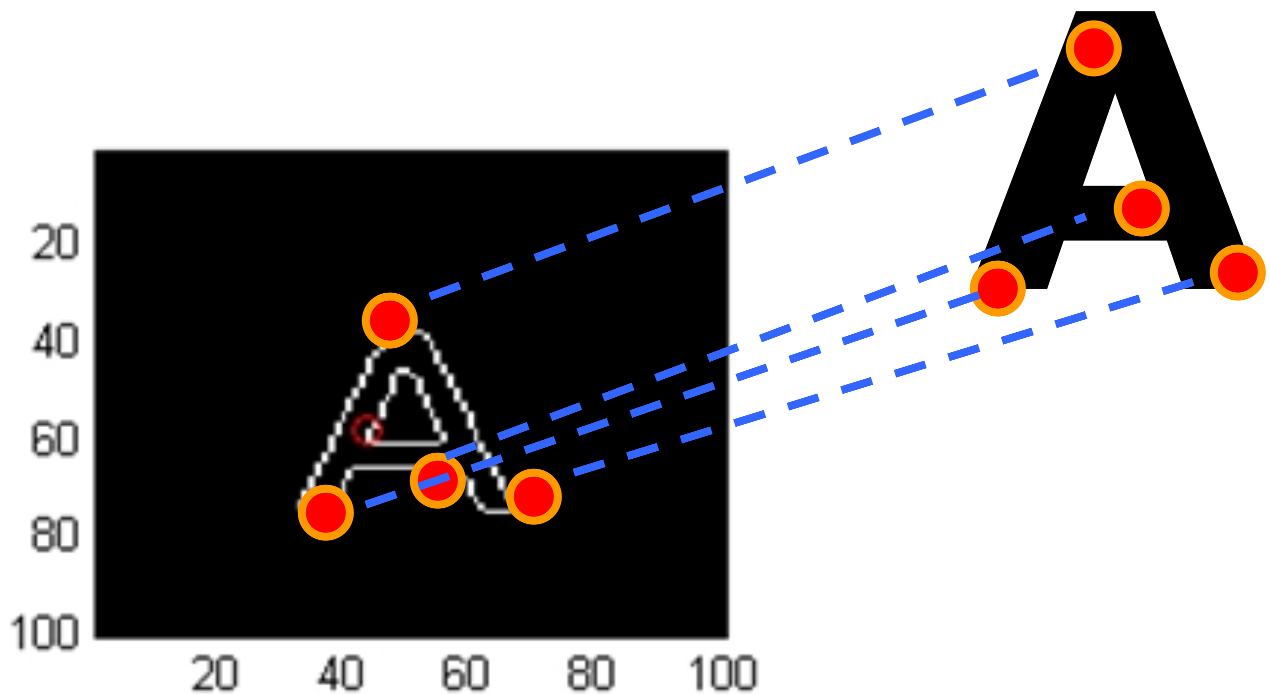
Estimation



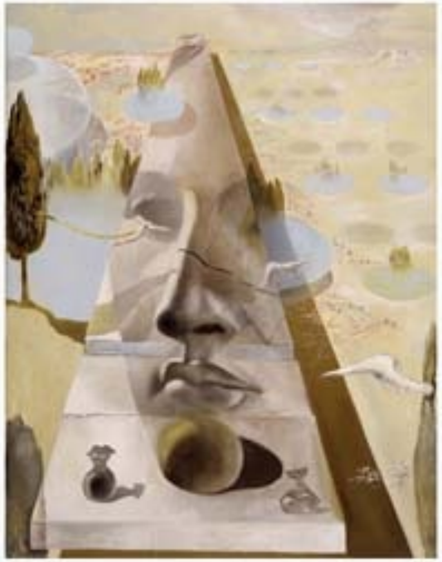
Matching



Object modeling and detection



Lecture 9
Detectors and descriptors



- Properties of detectors
  - Edge detectors
  - Harris
  - DoG
- Properties of descriptors
  - SIFT
  - HOG
  - Shape context

Edge detection

What causes an edge?

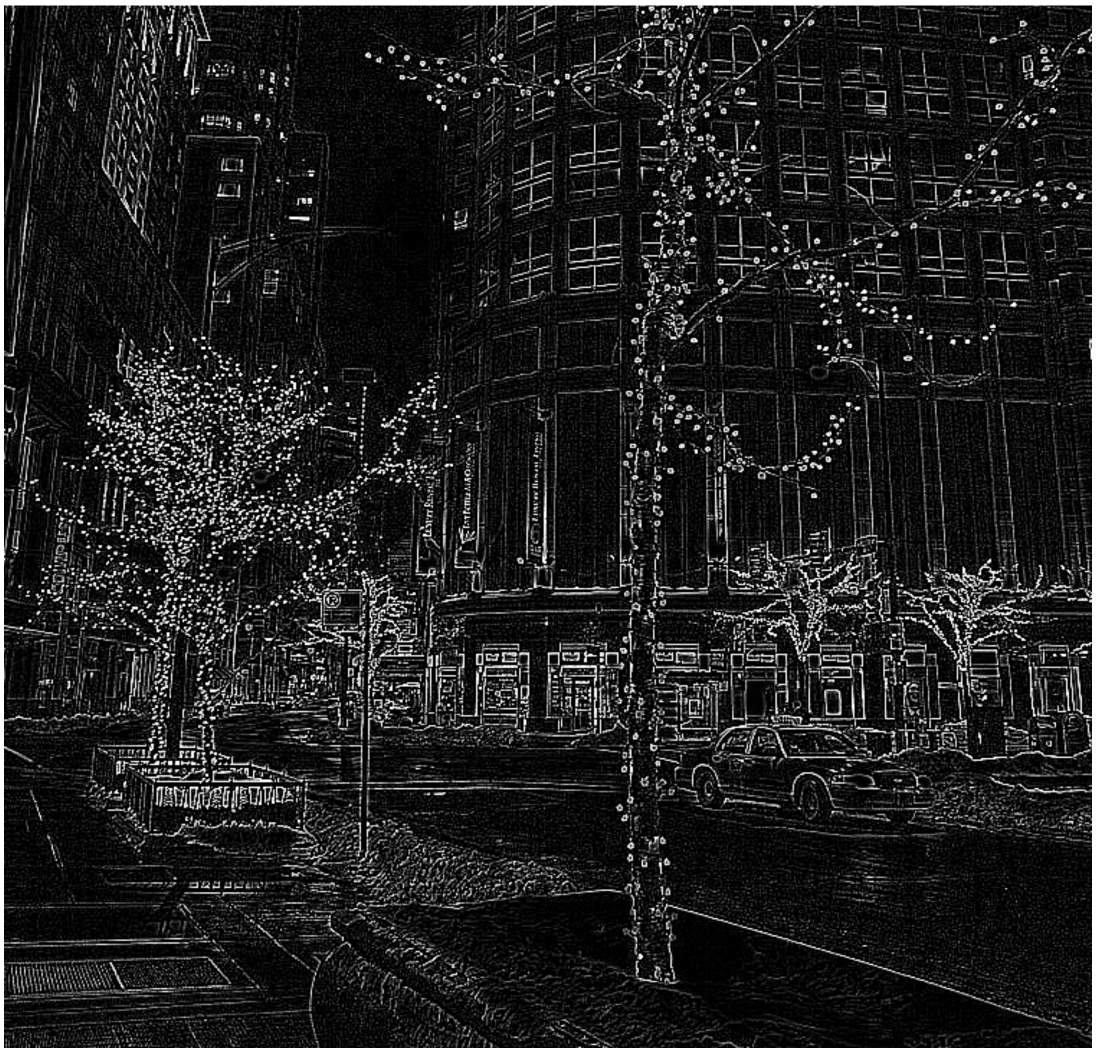
Identifies sudden changes in an image

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., highlights; shadows)

Example of edge detection

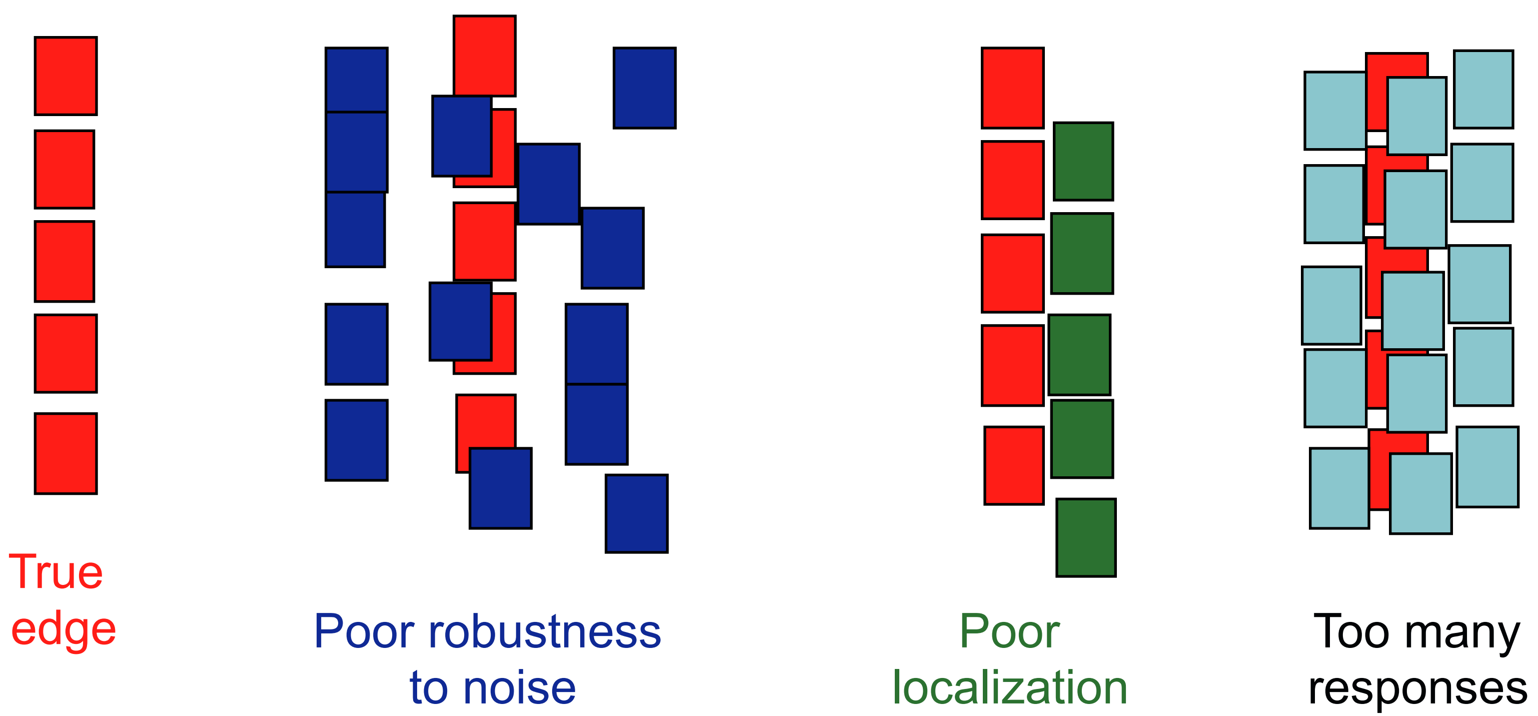


Edge Detection

- 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

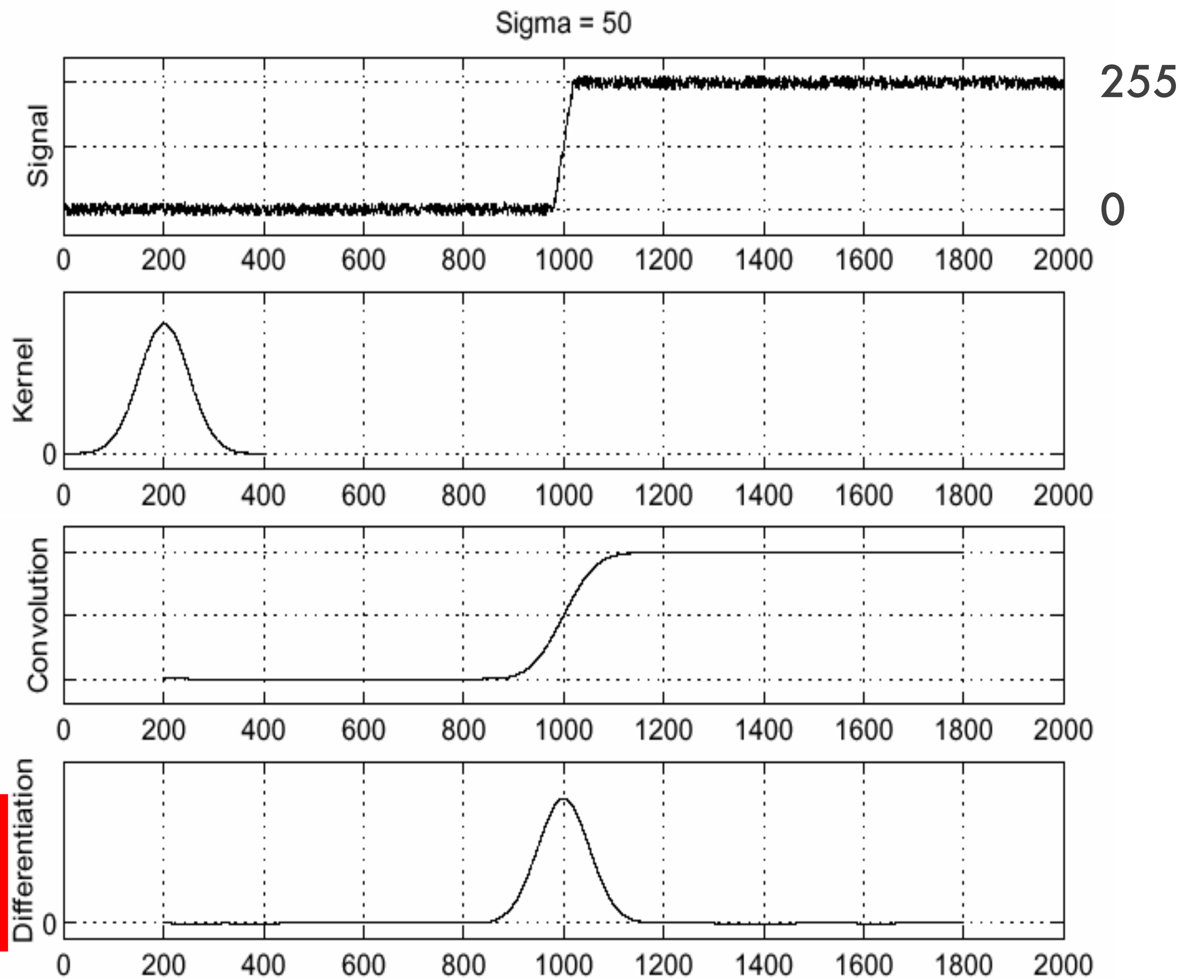
- Examples:



Designing an edge detector

- **Two ingredients:**
- Use derivatives (in x and y direction) to define a location with high gradient .
- Need smoothing to reduce noise prior to taking derivative

Designing an edge detector



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

[Eq. 1]

$$= \frac{dg}{dx} * f$$ = "derivative of Gaussian" filter

[Eq. 2]

See CS231A, lecture 4 for details on convolution and linear filters

Source: S. Seitz

Edge detector in 2D

- Smoothing

$$I' = g(x,y) * I$$ [Eq. 3]

$$g(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$ [Eq. 4]

- Derivative

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

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

[Eq. 6]

$$= \begin{bmatrix} g_x \\ g_y \end{bmatrix} * I = \begin{bmatrix} g_x * I \\ g_y * I \end{bmatrix} = \begin{bmatrix} S_x & S_y \end{bmatrix}$$ = gradient vector

[Eq. 5]

Canny Edge Detection (Canny 86):

See CS131A for details

original

Canny with $\sigma = 1$

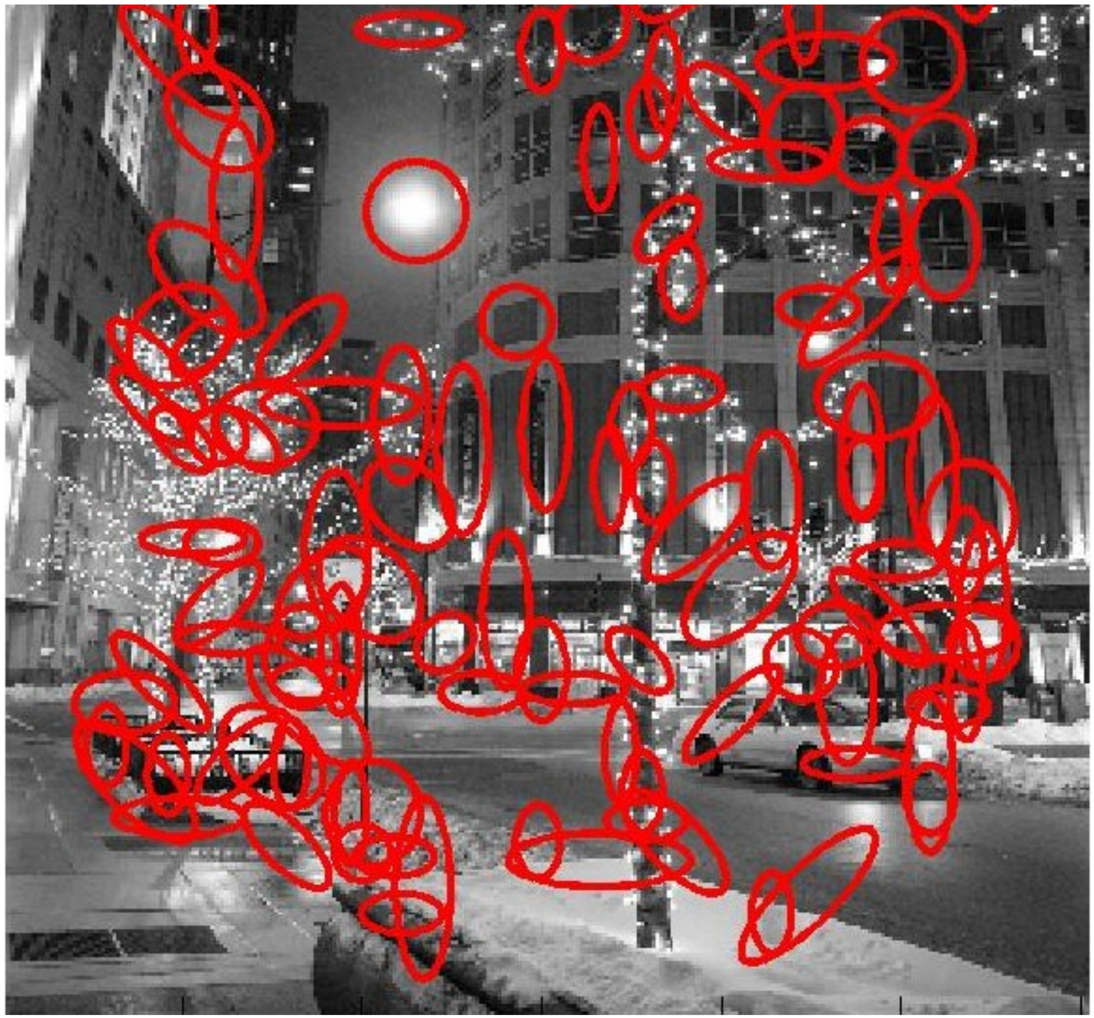
Canny with $\sigma = 2$

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

Other edge detectors:

- Sobel
- Canny-Deriche
- Differential

Corner/blob detectors



Corner/blob detectors

- 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

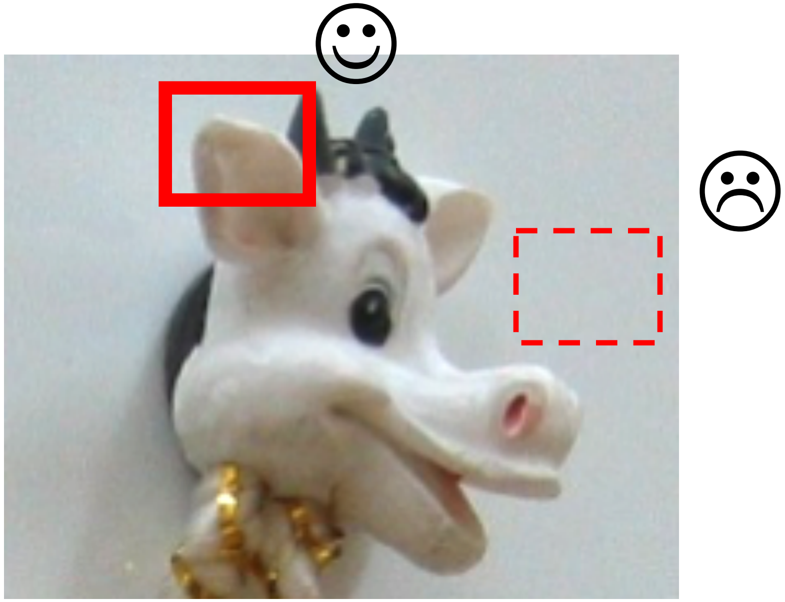
Illumination invariance

Scale invariance

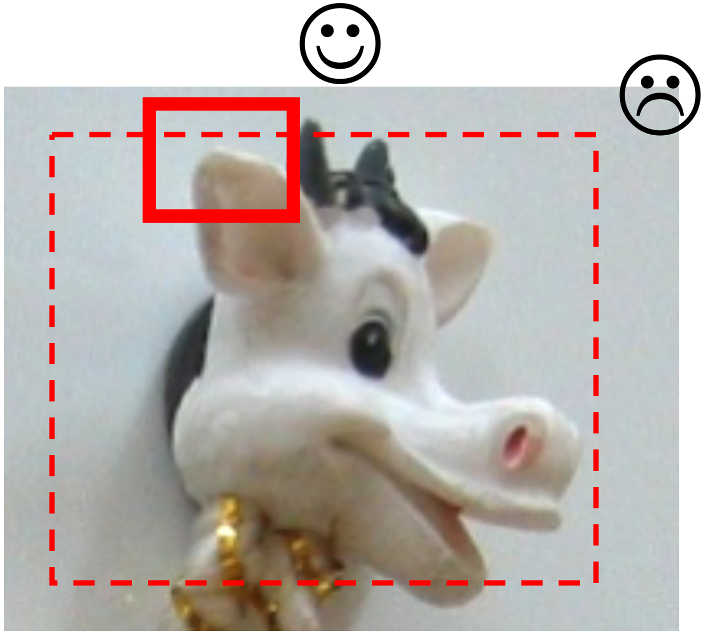
Pose invariance

- Rotation
- Affine

- Saliency



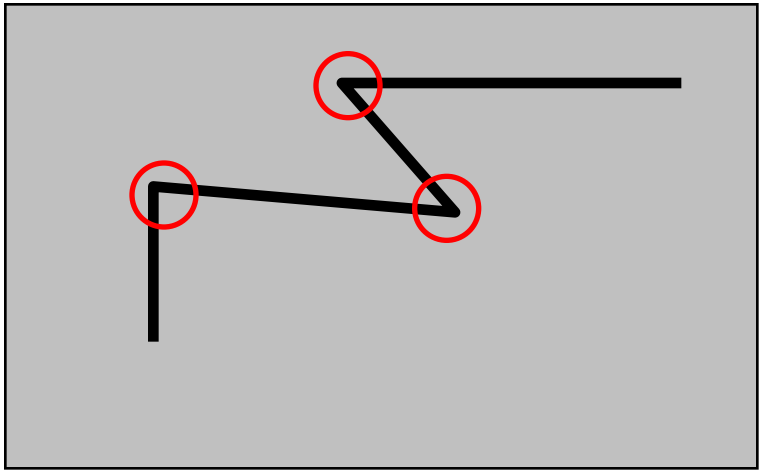
- Locality



Harris corner detector

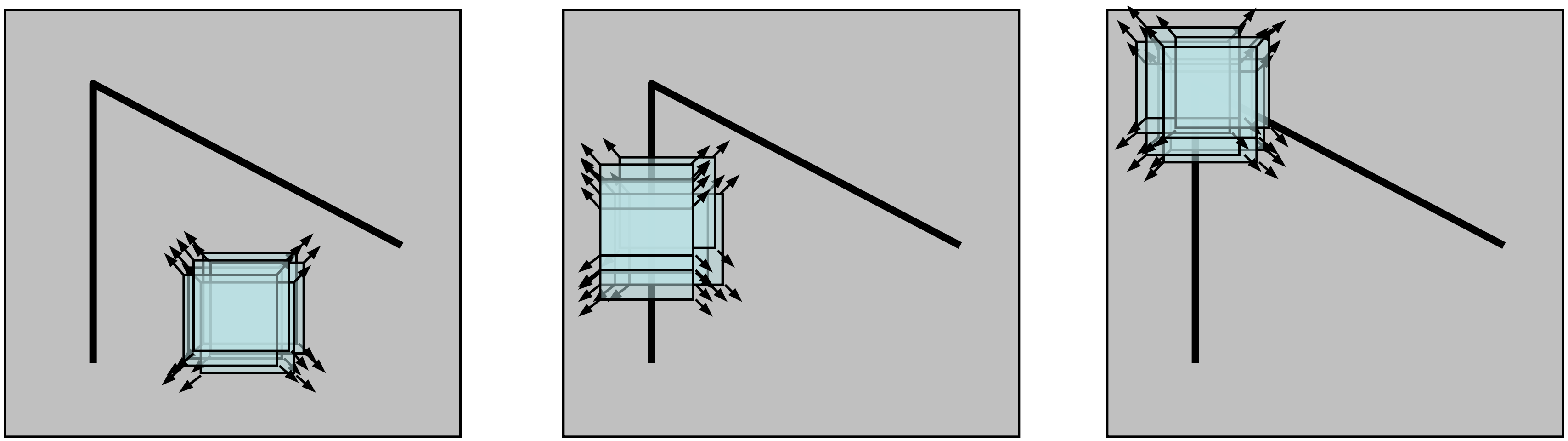
C.Harris and M.Stephens. "A Combined Corner and Edge Detector.“ *Proceedings of the 4th Alvey Vision Conference*: pages 147--151.

See CS131A for details



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

Results

Harris corner doesn't tell us the scale of the corner!

Blob detectors

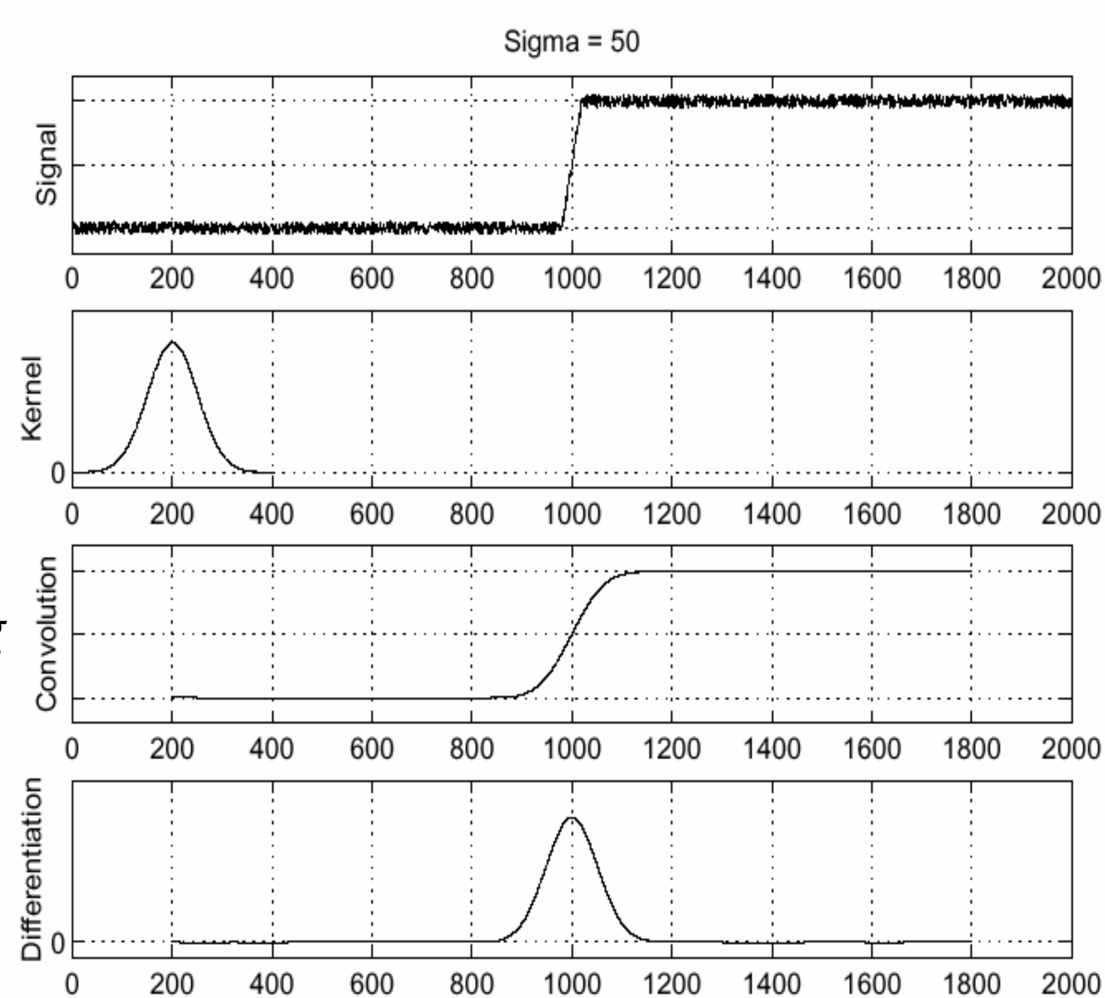
Edge detection

f

g

$f * g$

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



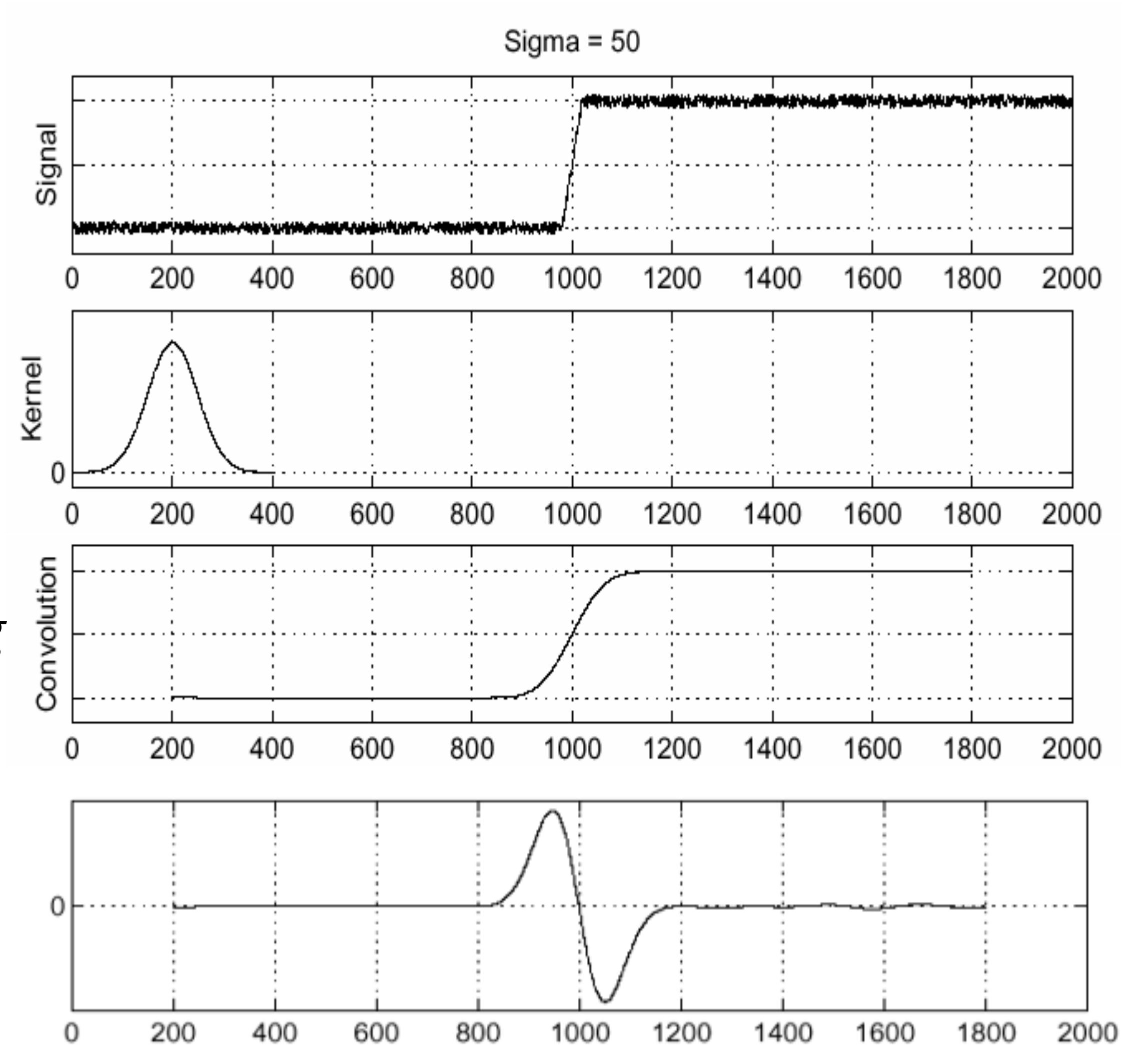
Source: S. Seitz

Edge detection

f

g

$f * g$



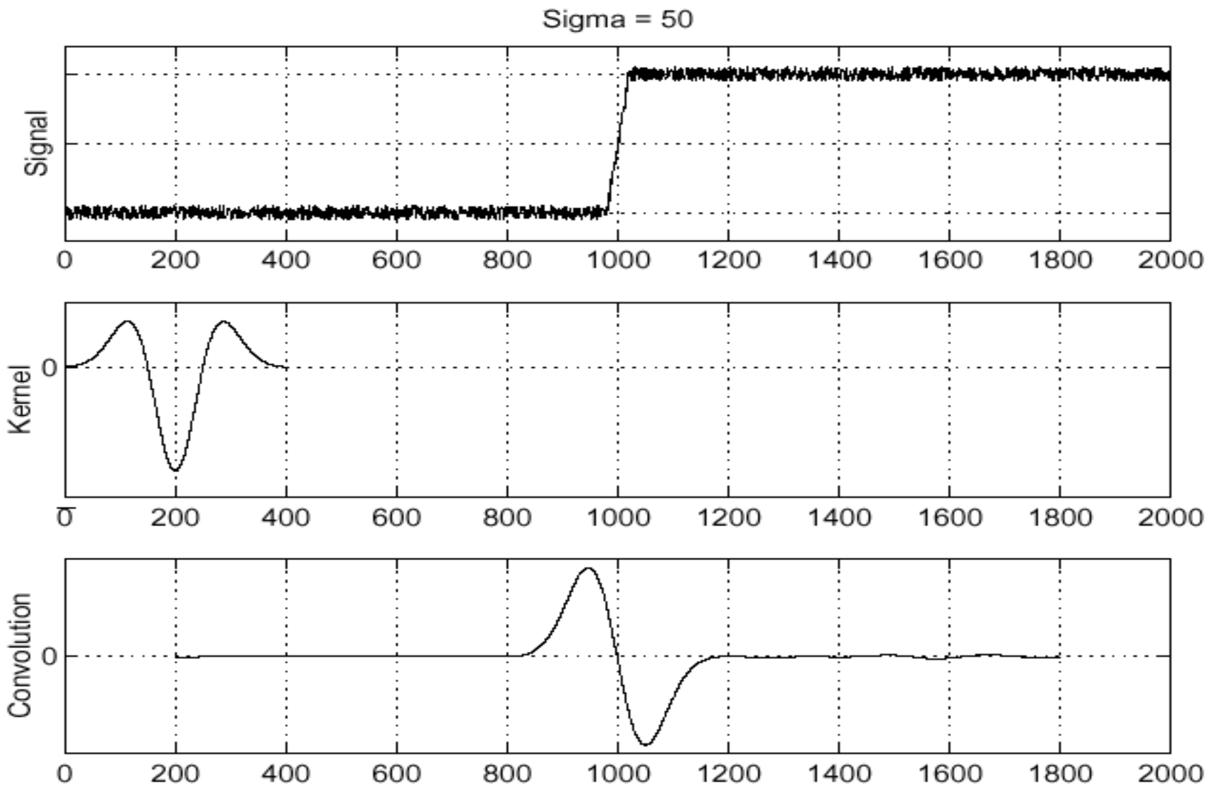
[Eq. 7] $\boxed{\frac{d^2}{dx^2}(f * g)}$

[Eq. 8] $f * \frac{d^2}{dx^2} g$ = "second derivative of Gaussian" filter = Laplacian of the gaussia

Edge detection as zero crossing

f

Edge



$$\frac{d^2}{dx^2}g$$

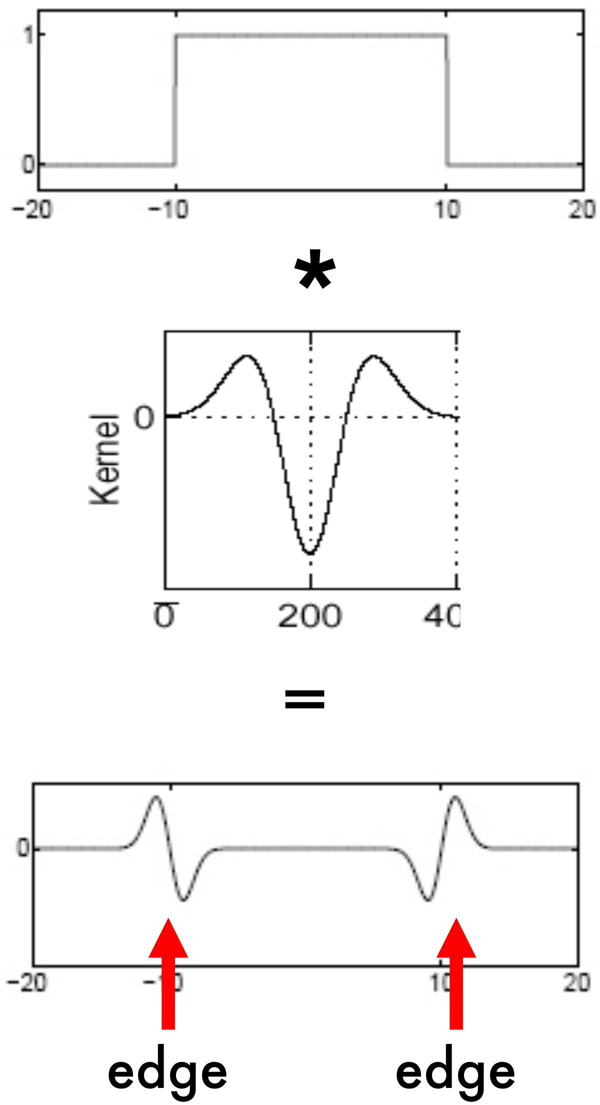
Laplacian

$$f * \frac{d^2}{dx^2}g$$

[Eq. 8]

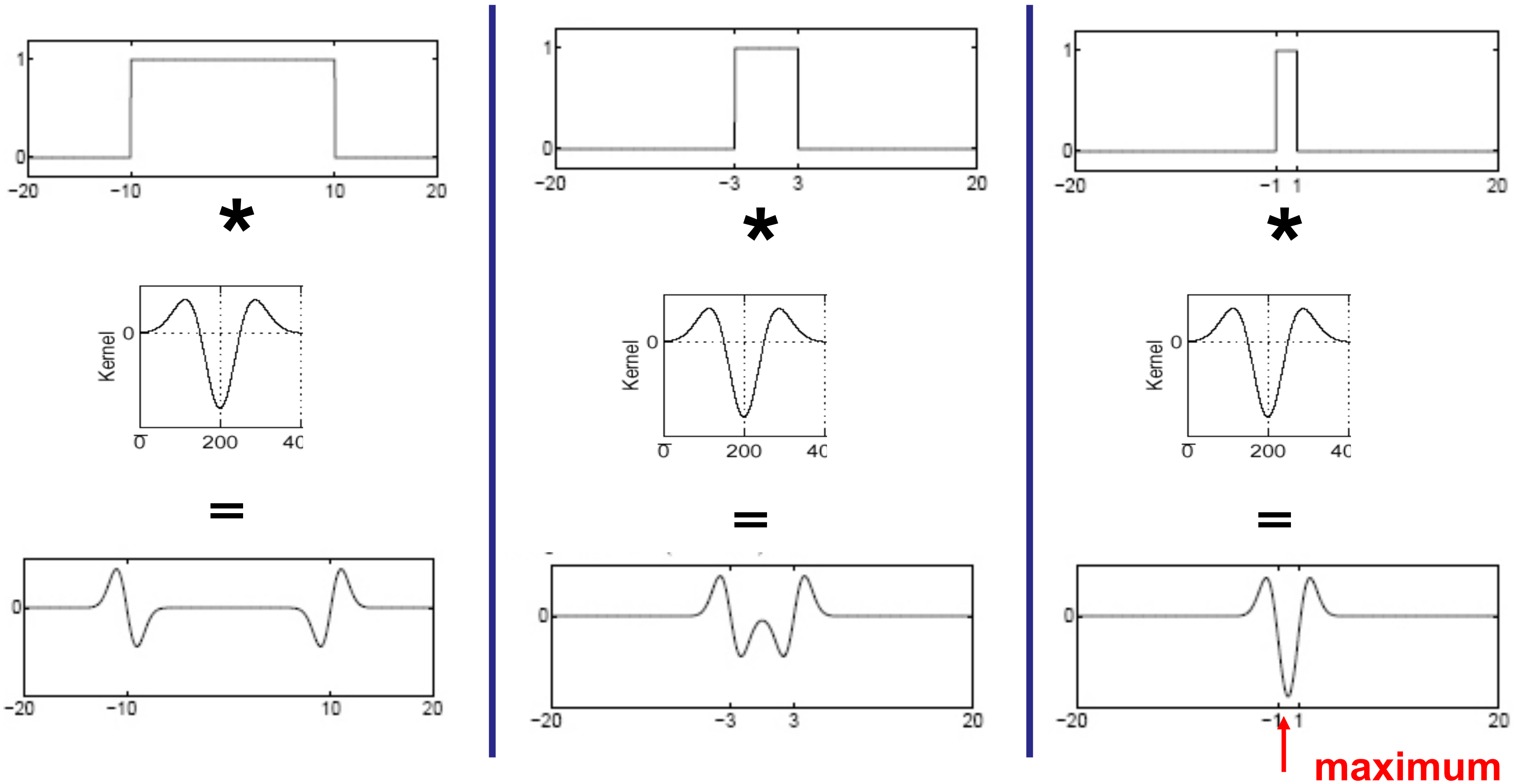
Edge = zero crossing of the second derivative

Edge detection as zero crossing



From edges to blobs

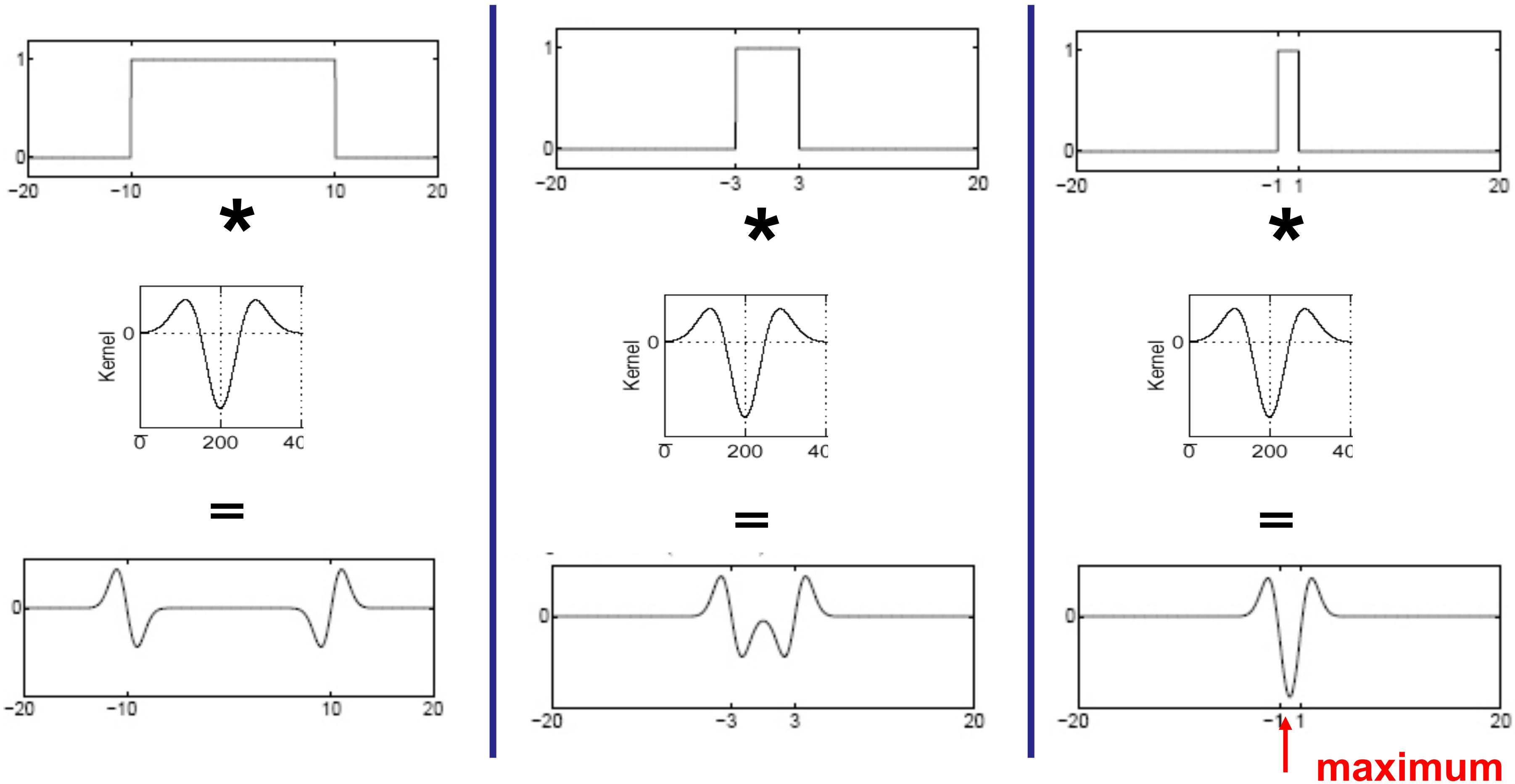
- Can we use the laplacian to find a blob (RECT function)?



Magnitude of the Laplacian response achieves a maximum at the center of the blob, provided the scale of the Laplacian is “matched” to the scale of the blob

From edges to blobs

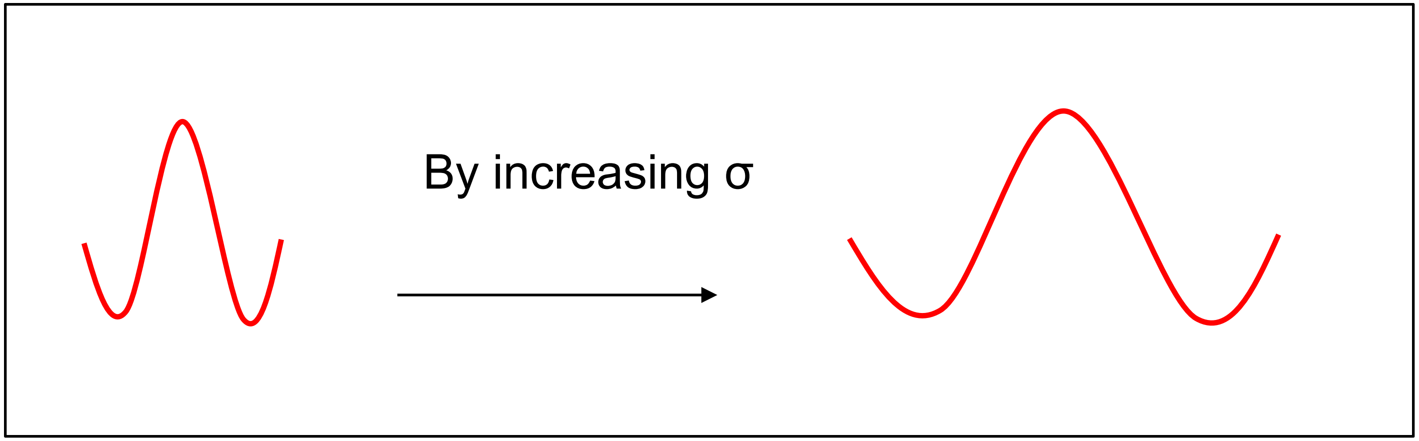
- Can we use the laplacian to find a blob (RECT function)?



What if the blob is slightly thicker or slimmer?

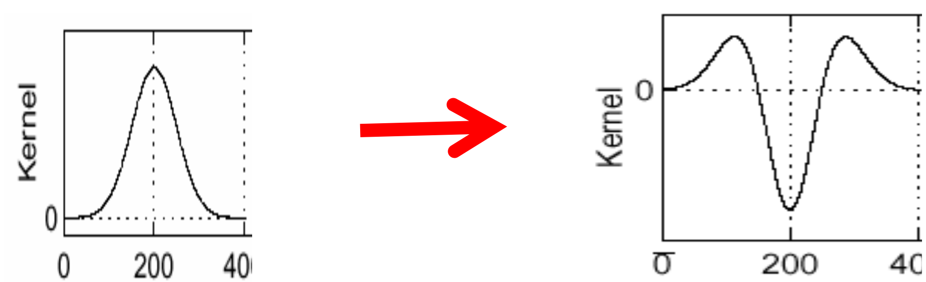
Scale selection

Convolve signal with Laplacians at several scales and looking for the maximum response. How in increase the scale??



Scale normalization

- To keep the energy of the response the same, must multiply Gaussian kernel by σ
- Laplacian is the second Gaussian derivative, so it must be multiplied by σ²



Normalized Laplacian

$$g(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{x^2}{2\sigma^2}}$$

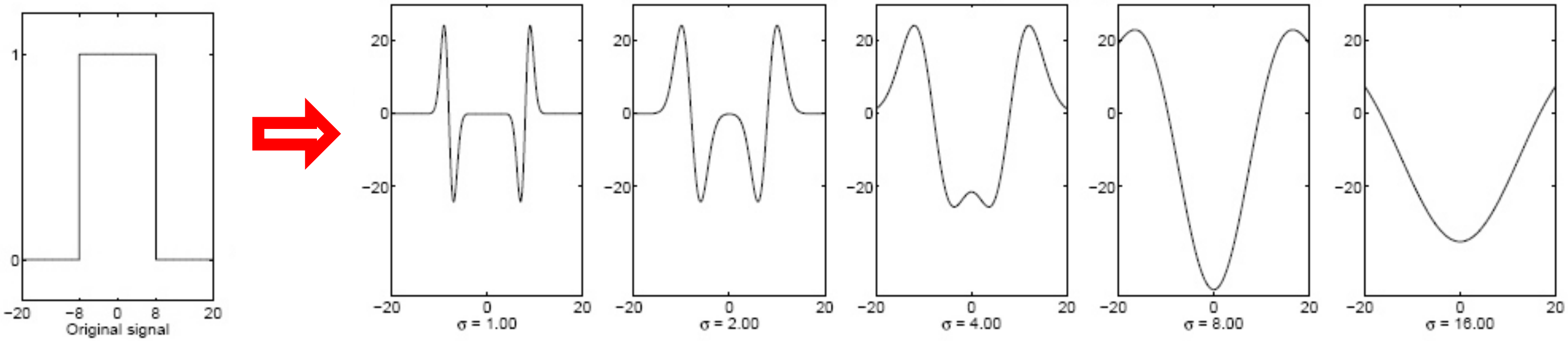
$$\sigma^2 \frac{d^2}{dx^2} g_n$$

Characteristic scale

Scale-normalized Laplacian response

Original signal

$\sigma = 1$ $\sigma = 2$ $\sigma = 4$ $\sigma = 8$ $\sigma = 16$



Maximum ☺

The characteristic scale is the scale that produces peak of Laplacian response

This procedure allows us to:

1) detect the blob
2) estimate the size of the blob!

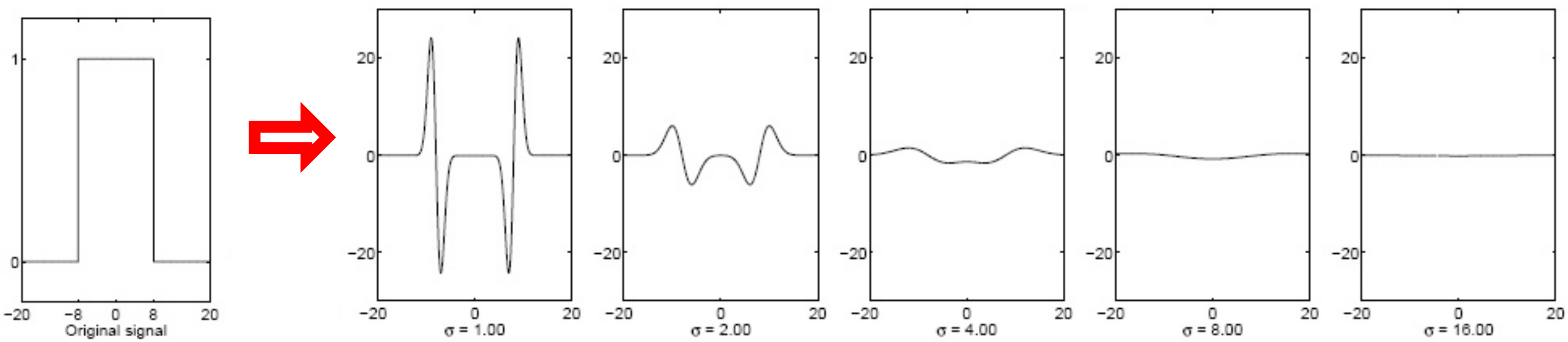
T. Lindeberg (1998). "Feature detection with automatic scale selection." *International Journal of Computer Vision* **30** (2): pp 77--116.

Characteristic scale

Here is what happens if we don't normalize the Laplacian:

Original signal

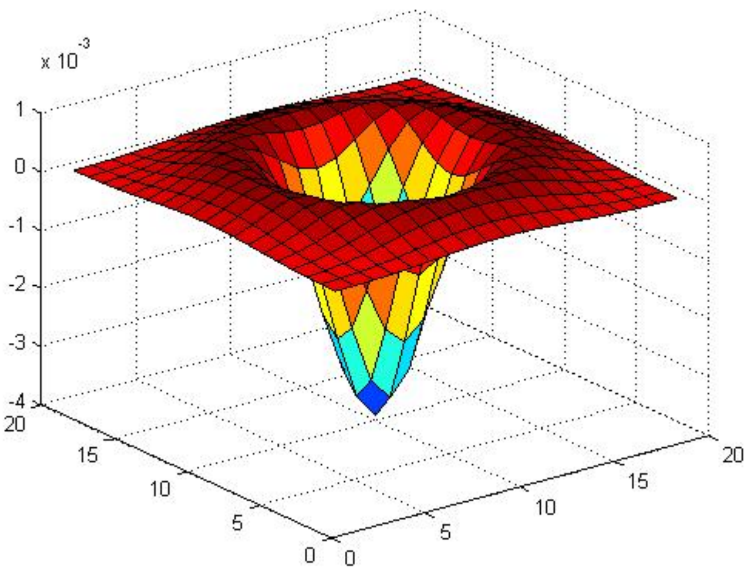
σ = 1 σ = 2 σ = 4 σ = 8 σ = 16

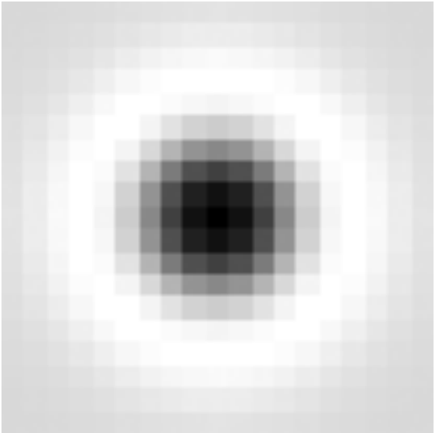


This should give the max response ☹

Blob detection in 2D

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



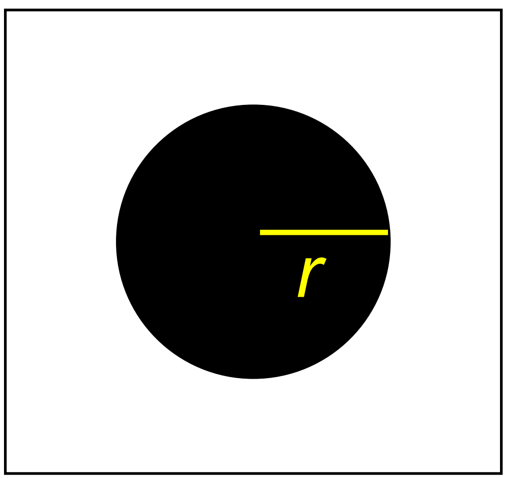


Scale-normalized: $\nabla^2_{\text{norm}} g = \sigma^2 \left(\frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2} \right) = \sigma^2 (g_{xx} + g_{yy})$

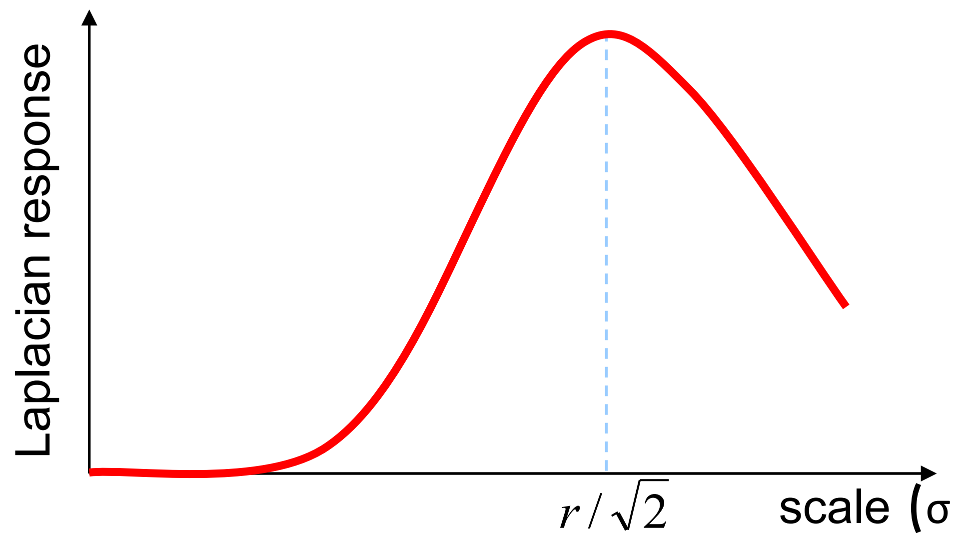
[Eq. 9]

Scale selection

- For a binary circle of radius r, the Laplacian achieves a maximum at $\sigma = r/\sqrt{2}$



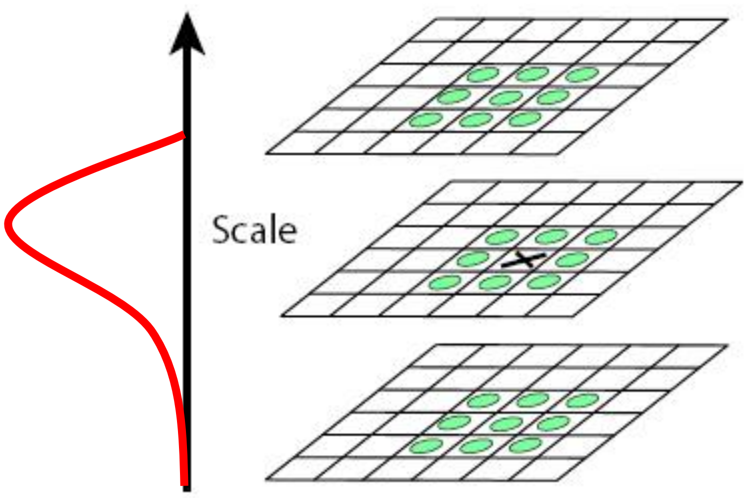
image



Scale-space blob detector

1. Convolve image with scale-normalized Laplacian at several scales
2. Find maxima of squared Laplacian response in scale-space

The maxima indicate that a blob has been detected and what's its intrinsic scale

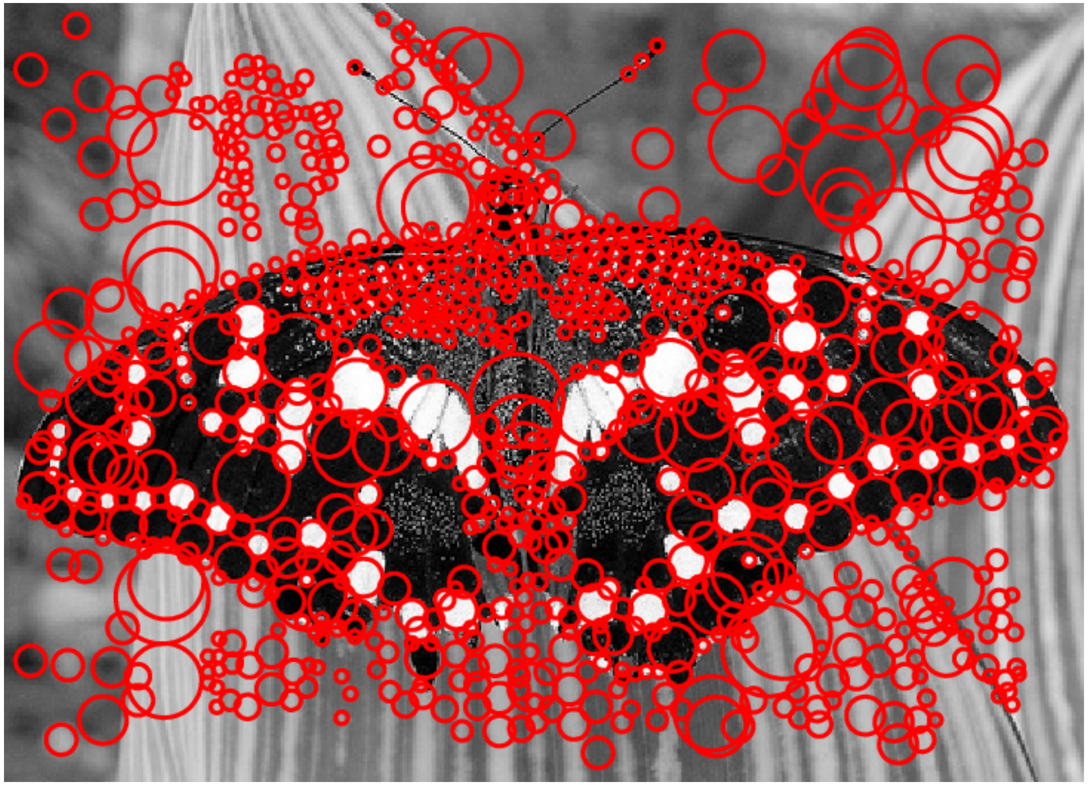


Scale-space blob detector: Example

Scale-space blob detector: Example

sigma = 11.9912

Scale-space blob detector: Example



Difference of Gaussians (DoG)

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

- Approximating the Laplacian with a difference of Gaussians:

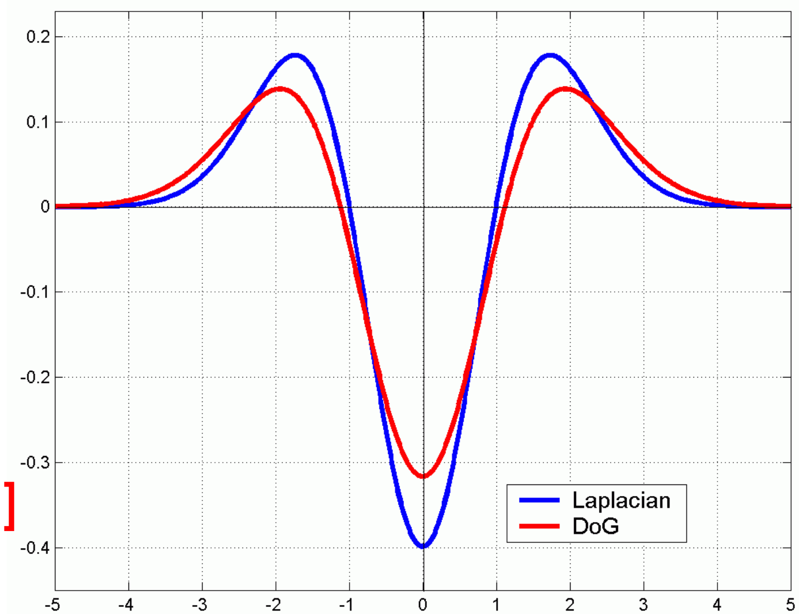
$$L = \sigma^2 \left(g_{xx}(x,y,\sigma) + g_{yy}(x,y,\sigma) \right)$$

(Laplacian) [Eq. 10]

$$DoG = g(x,y,2\sigma) - g(x,y,\sigma)$$

[Eq. 11]

Difference of gaussian with scales 2 σ and σ



In general:

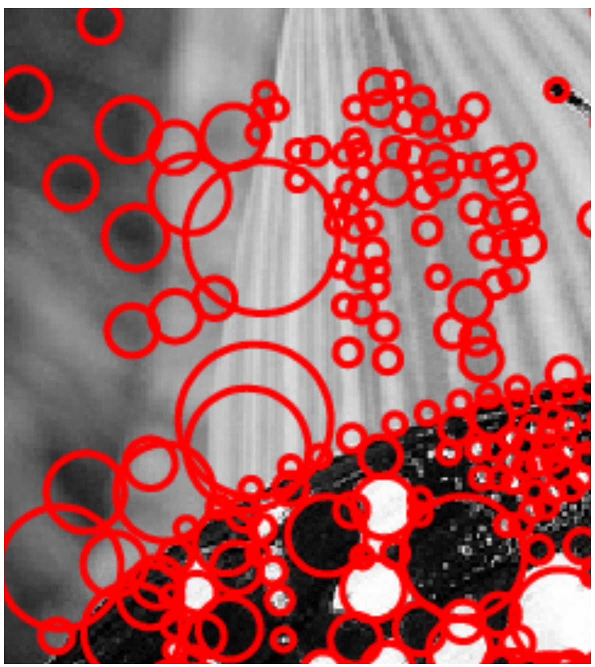
$$DoG = g(x,y,k\sigma) - g(x,y,\sigma) \approx (k-1)\sigma^2 \mathbf{L}$$

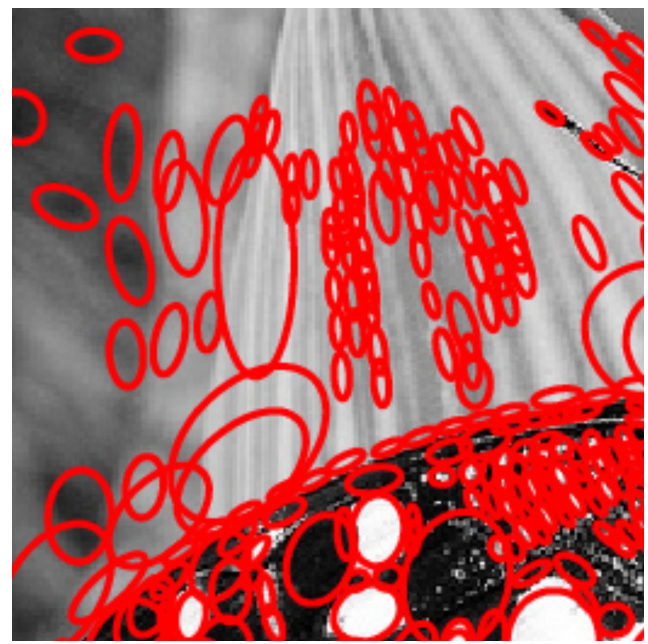
[Eq. 12]

Affine invariant detectors

K. Mikolajczyk and C. Schmid, Scale and Affine invariant interest point detectors, IJCV 60(1):63-86, 2004.

Similarly to characteristic scale, we can define the characteristic shape of a blob

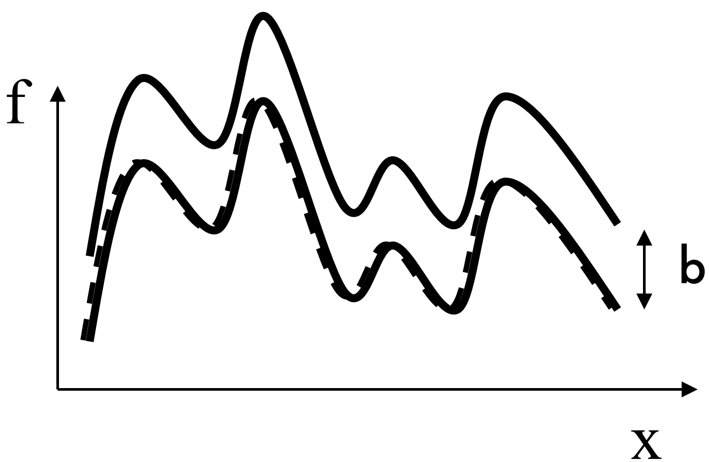




Properties of detectors

Detector	Illuminatio n	Rotation	Scale	View point
Lowe '99 (DoG)	Yes*			

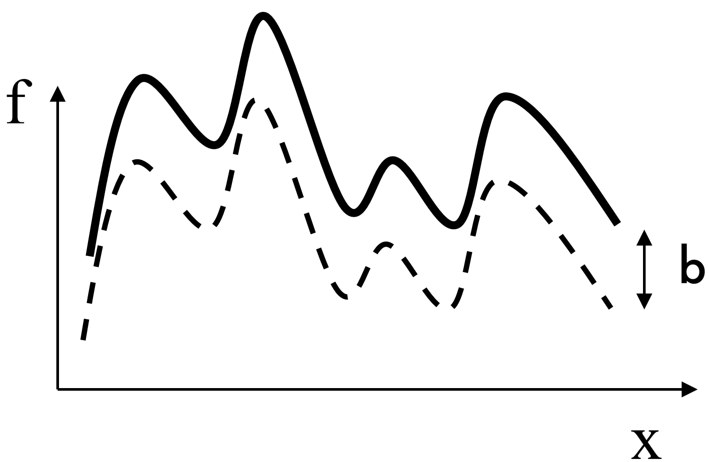
$$f \rightarrow f + b$$



Properties of detectors

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

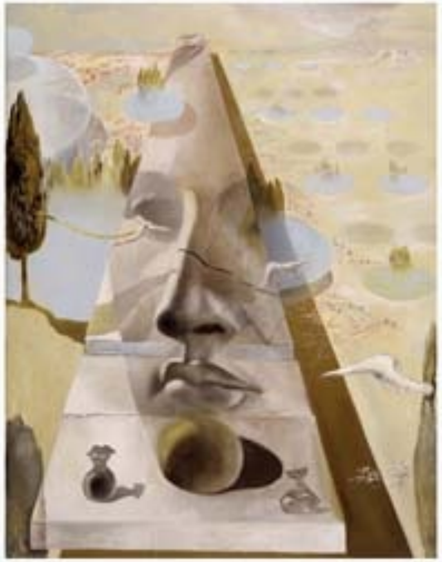
$$f \rightarrow f + b$$



Properties of detectors

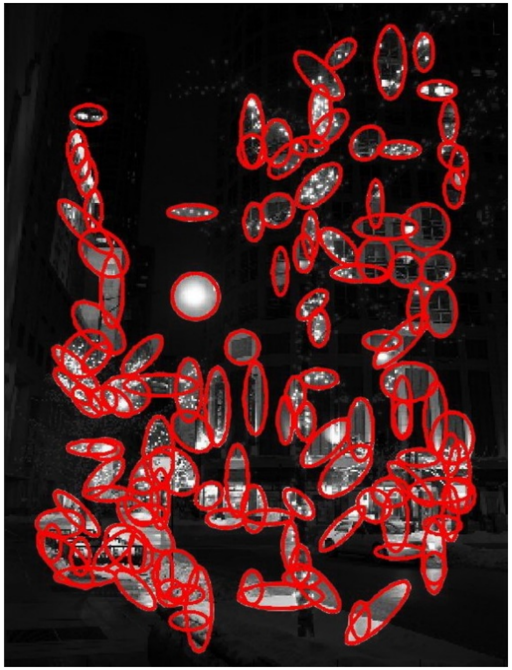
Detector	Illumination	Rotation	Scale	View point
Lowe '99 (DoG)	Yes*	Yes	Yes	No
Harris corner	Yes*	Yes	No	No
Mikolajczyk & Schmid '01, '02	Yes*	Yes	Yes	Yes
Tuytelaars, '00	Yes*	Yes	No (Yes '04)	Yes
Kadir & Brady, 01	Yes*	Yes	Yes	no
Matas, '02	Yes*	Yes	Yes	no

Lecture 9
Detectors and descriptors



- Properties of detectors
  - Edge detectors
  - Harris
  - DoG
- Properties of descriptors
  - SIFT
  - HOG
  - Shape context

The big picture...



Feature Detection

e.g. DoG

Feature Description

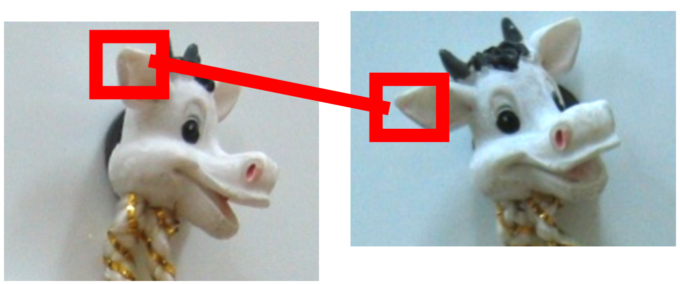
e.g. SIFT

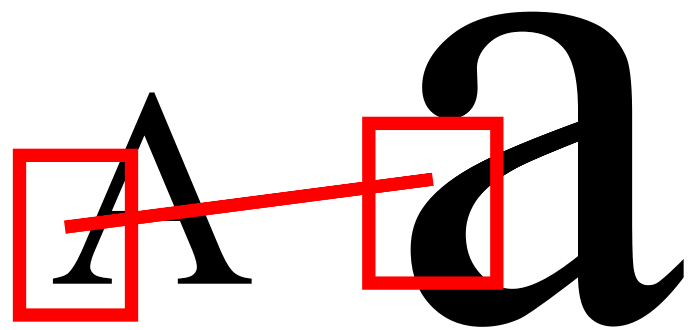
- Estimation
- Matching
- Indexing
- Detection

Properties

Depending on the application a descriptor must incorporate information that is:

- Invariant w.r.t:
  - Illumination
  - Pose
  - Scale
  - Intraclass variability

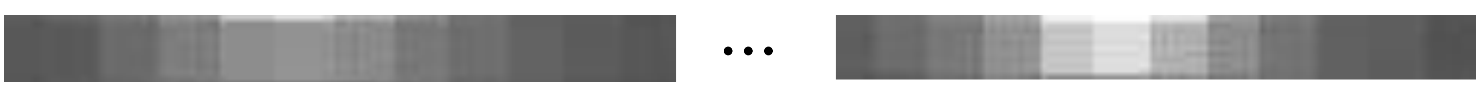




- **Highly distinctive** (allows a single feature to find its correct match with good probability in a large database of features)

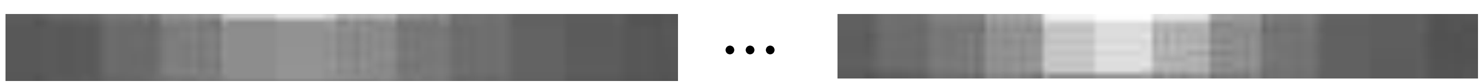
The simplest descriptor

1 x NM vector of pixel intensities

w= []

Normalized vector of intensities

1 x NM vector of pixel intensities

w= []

$$w_n = \frac{(w - \overline{w})}{\left\|(w - \overline{w})\right\|}$$

Makes the descriptor invariant with respect to affine transformation of the illumination condition

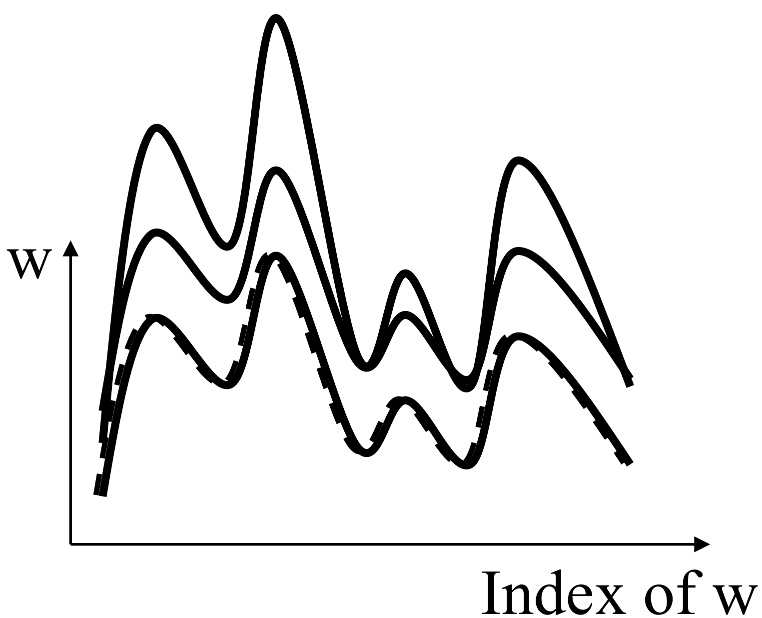
[Eq. 13]

Illumination normalization

- *Affine intensity* change:

$$w \rightarrow w + b \quad \text{[Eq. 14]}$$
$$\rightarrow a\,w + b$$

$$w_n = \frac{(w - \overline{w})}{\left\|(w - \overline{w})\right\|}$$

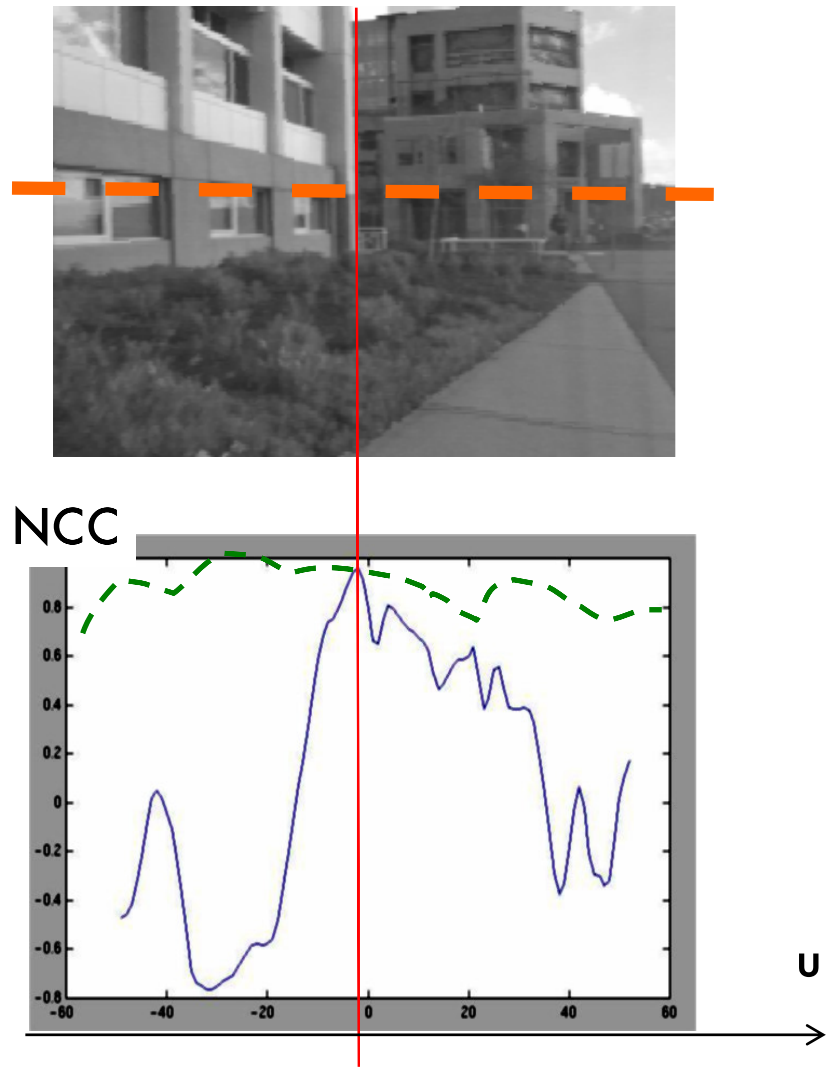


- Make each patch zero mean: remove b
- Make unit variance: remove a

Why can't we just use this?

- Sensitive to small variation of:
    - location
    - Pose
    - Scale
    - intra-class variability
- Poorly distinctive

Sensitive to pose variations



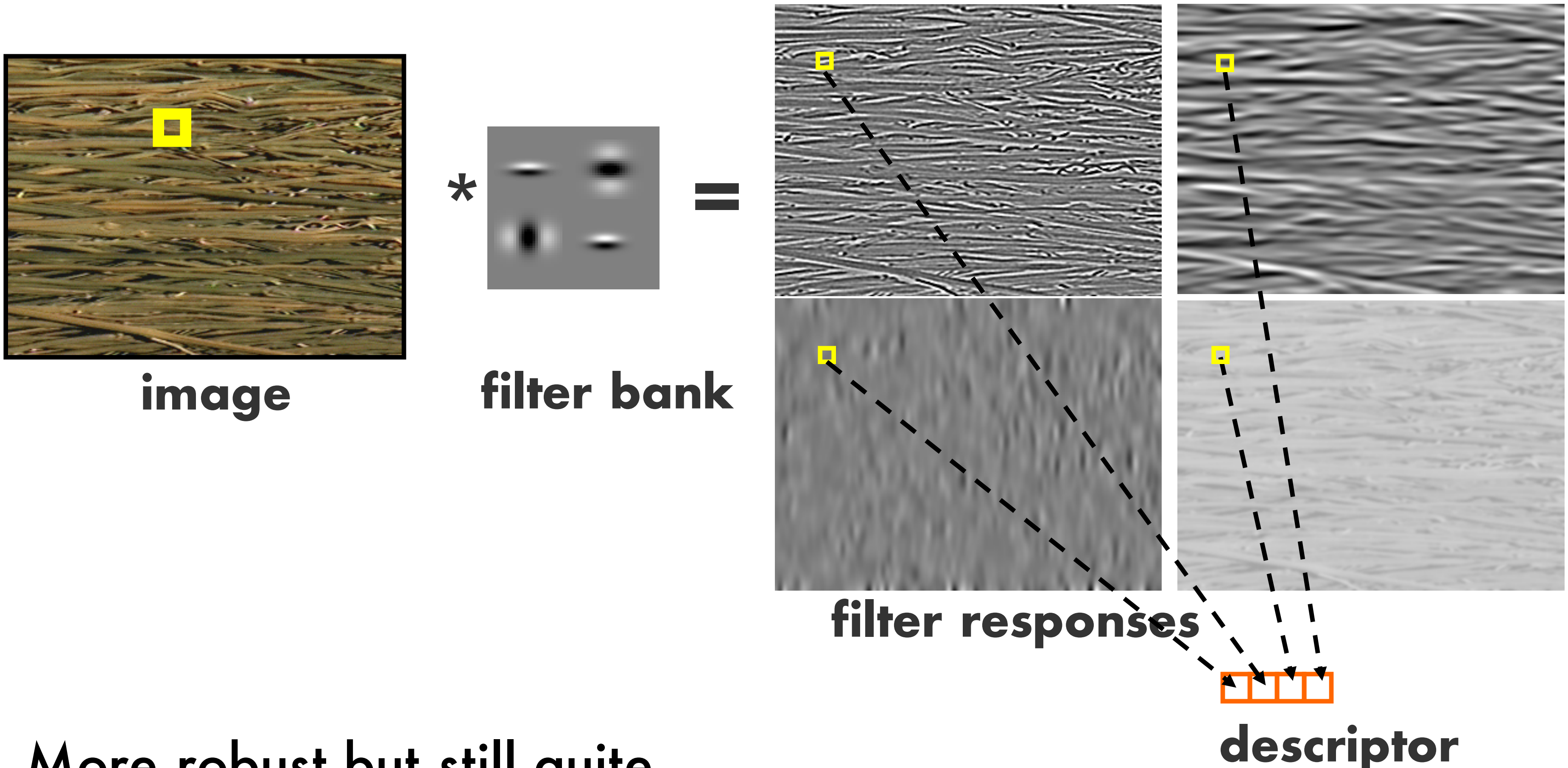
Normalized Correlation:

$$w_n \cdot w'_n = \frac{(w - \overline{w})(w' - \overline{w}')}{\left\|(w - \overline{w})(w' - \overline{w}')\right\|}$$

Properties of descriptors

Descripto r	Illuminatio n	Pose	Intra-class variab.
PATCH	Good	Poor	Poor

Bank of filters



More robust but still quite sensitive to pose variations

http://people.csail.mit.edu/billf/papers/steerpaper91FreemanAdelson.pdf

A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. IJCV, 2001.

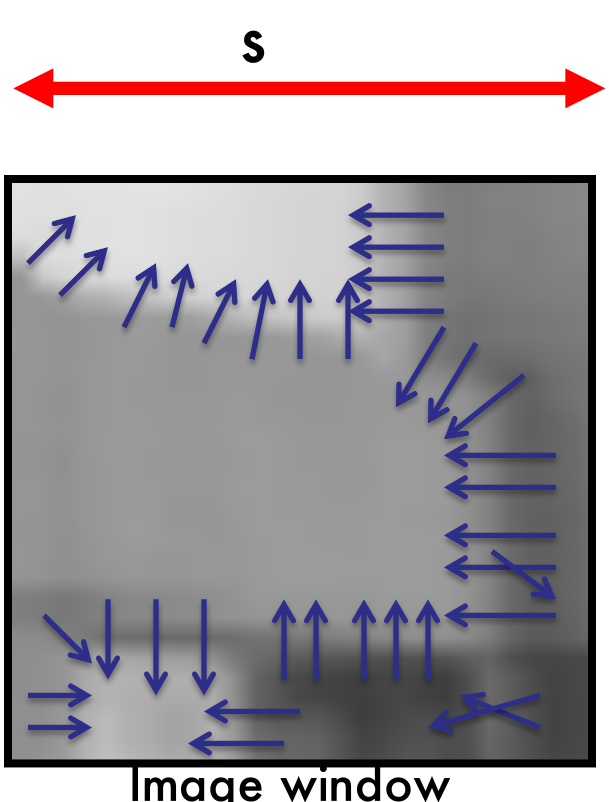
Properties of descriptors

Descriptor	Illumination	Pose	Intra-class variab.
PATCH	Good	Poor	Poor
FILTERS	Good	Medium	Medium

SIFT descriptor

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

- Alternative representation for image regions
- Location and characteristic scale s given by DoG detector



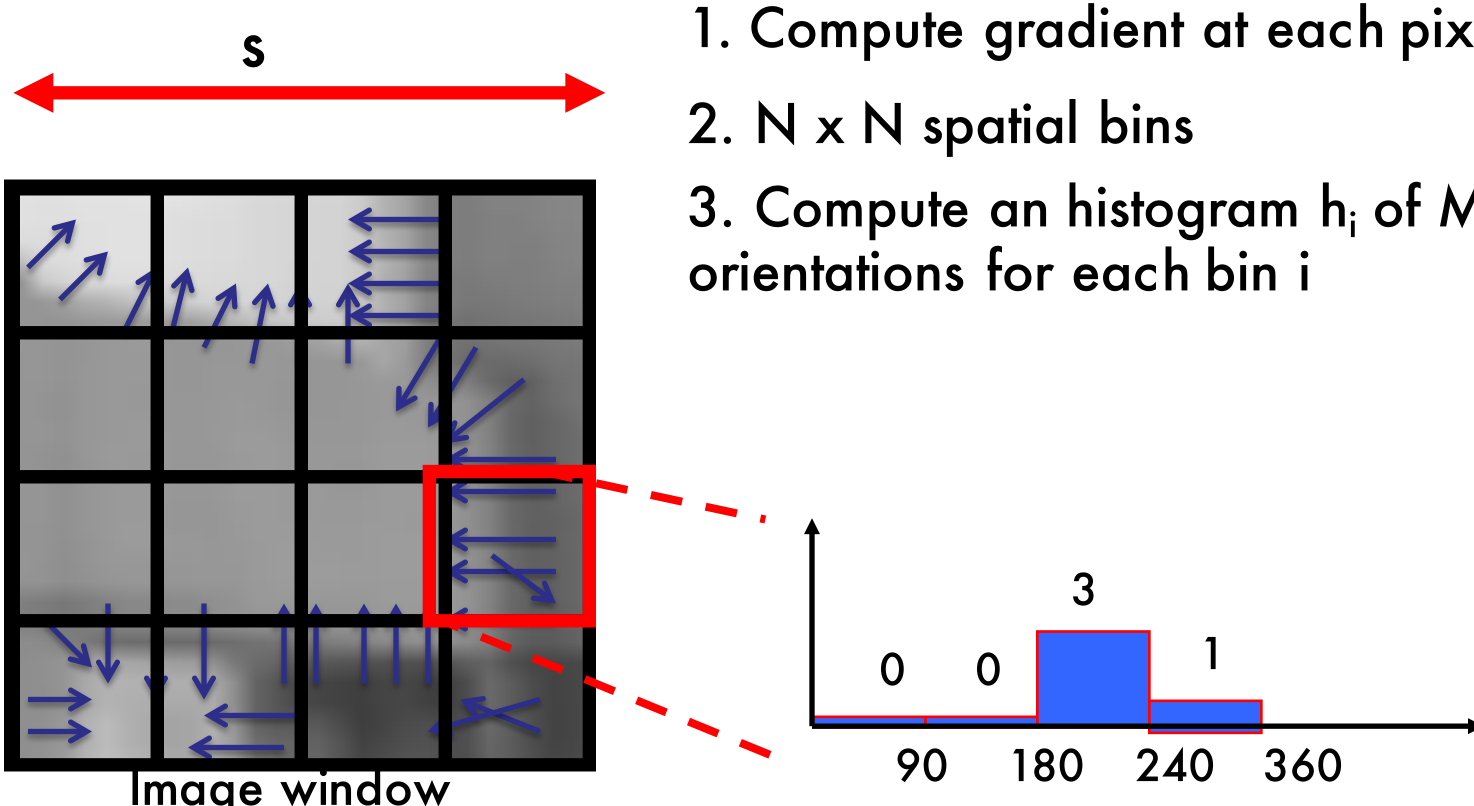
- Compute gradient at each pixel

SIFT descriptor

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

- Alternative representation for image regions
- Location and characteristic scale s given by DoG detector

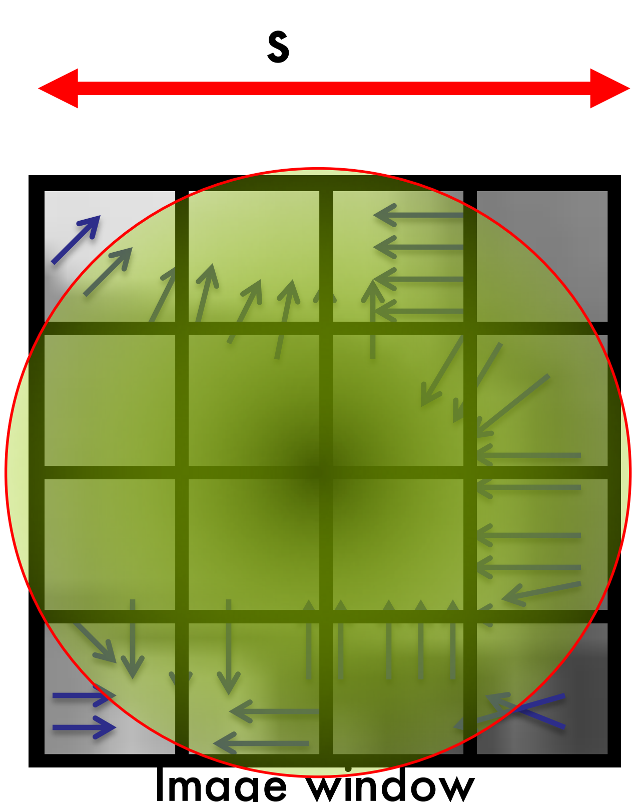
1. Compute gradient at each pixel
2. N x N spatial bins
3. Compute an histogram h_i of M orientations for each bin i



SIFT descriptor

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

- Alternative representation for image regions
- Location and characteristic scale s given by DoG detector



1 Compute gradient at each pixel

2 N x N spatial bins

3 Compute an histogram h_i of M orientations for each bin i

4 Concatenate h_i for i=1 to N^2 to form a $1xMN^2$ vector H

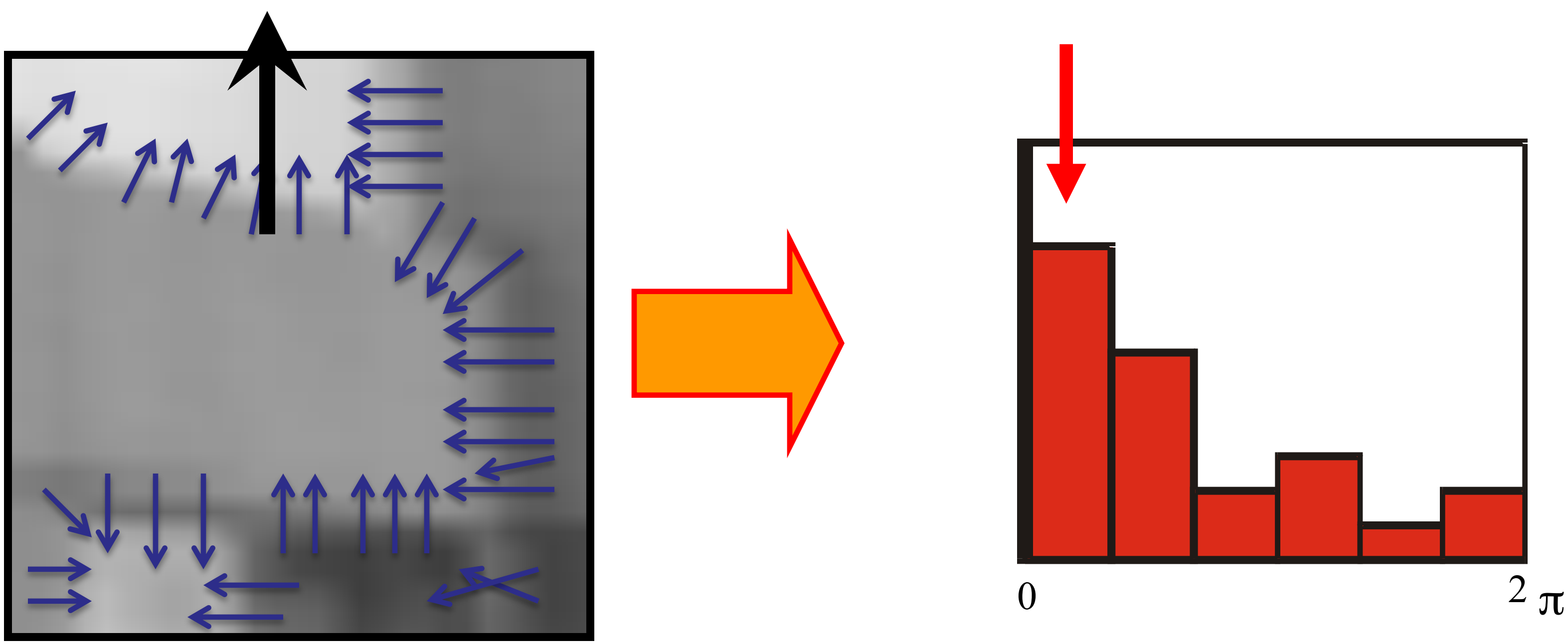
5 Gaussian center-weighting

6 Normalize to unit norm

Typically M = 8; N= 4
H = 1 x 128 descriptor

Rotational invariance

- Find dominant orientation by building a orientation histogram
- Rotate all orientations by the dominant orientation



This makes the SIFT descriptor rotational invariant

Properties of descriptors

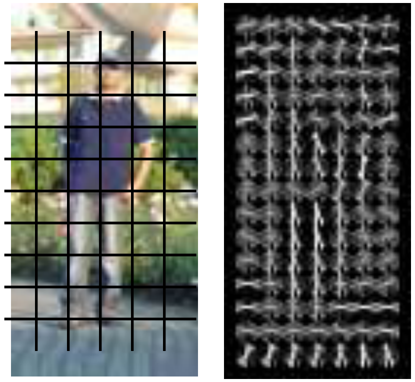
Descriptor	Illumination	Pose	Intra-class variab.
PATCH	Good	Poor	Poor
FILTERS	Good	Medium	Medium
SIFT	Good	Good	Medium

- SIFT is robust w.r.t. small variation in:
  - Illumination (thanks to gradient & normalization)
  - Pose (small affine variation thanks to orientation histogram)
  - Scale (scale is fixed by DOG)
  - Intra-class variability (small variations thanks to histograms)

HoG = Histogram of Oriented Gradients

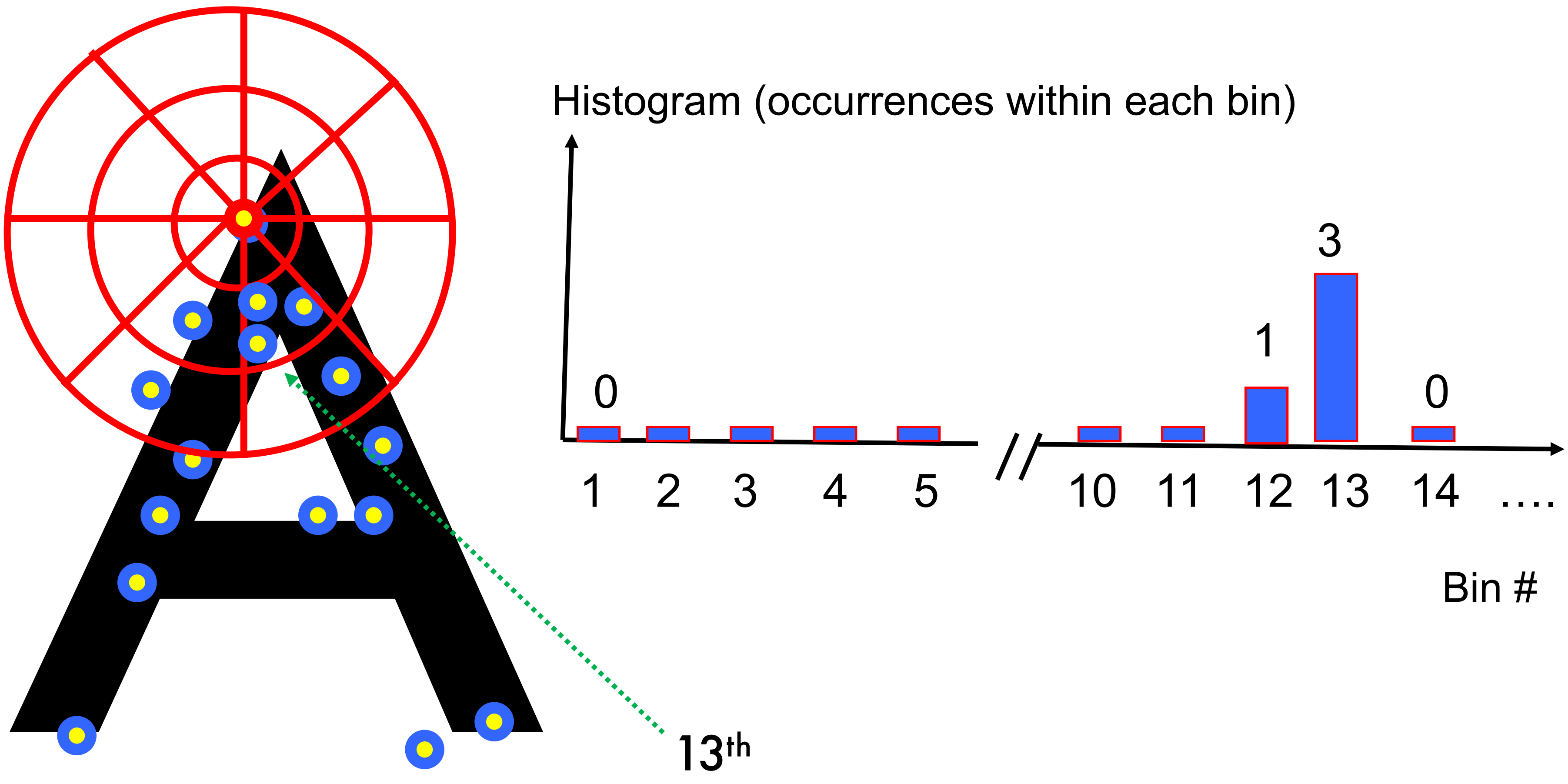
Navneet Dalal and Bill Triggs, Histograms of Oriented Gradients for Human Detection, CVPR05

- Like SIFT, but…
  - Sampled on a dense, regular grid around the object
  - Gradients are contrast normalized in overlapping blocks

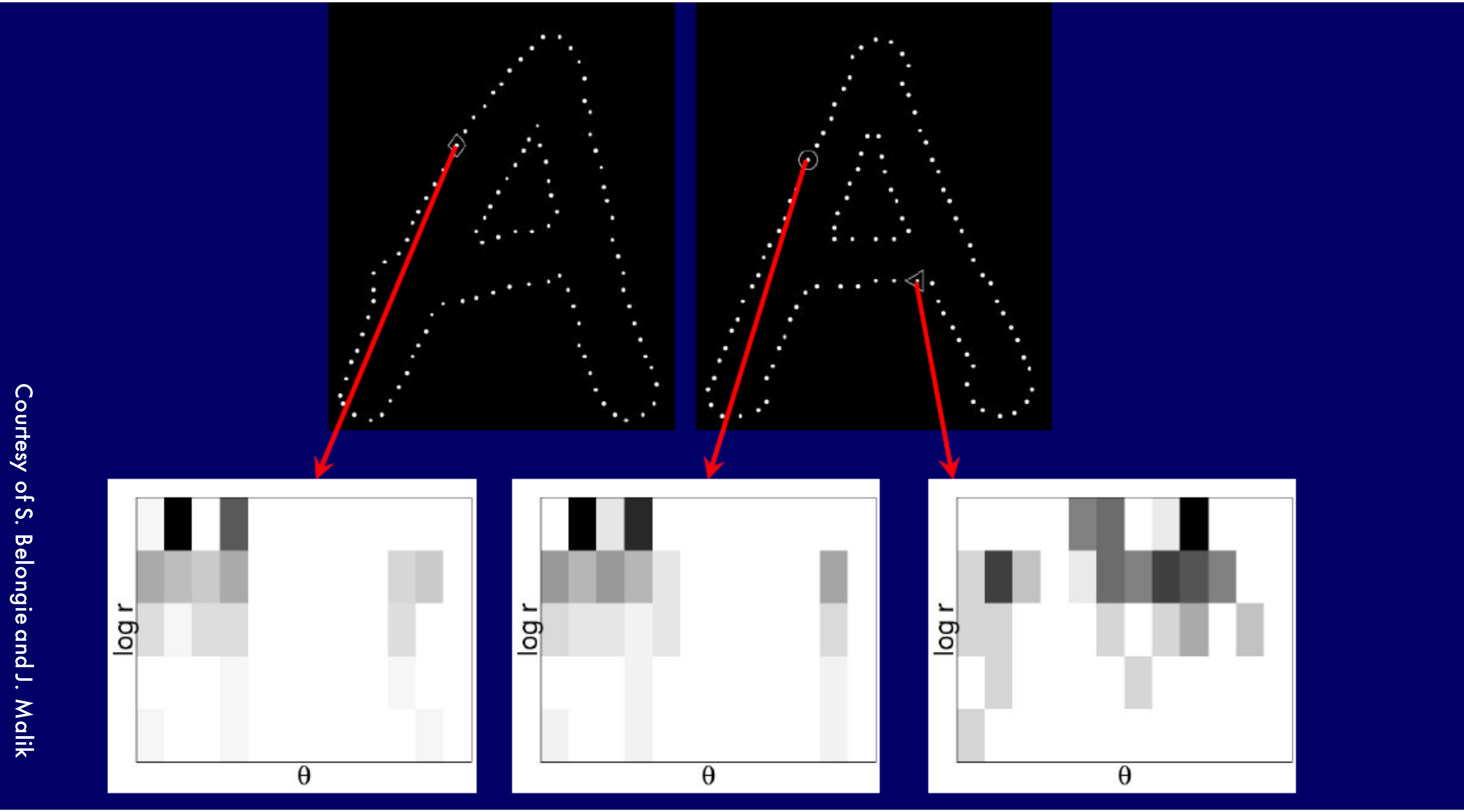


Shape context descriptor

Belongie et al. 2002



Shape context descriptor



descriptor 1 descriptor 2 descriptor 3

Other detectors/descriptors

- **HOG: Histogram of oriented gradients**

  Dalal & Triggs, 2005

- **SURF: Speeded Up Robust Features**

  Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc Van Gool, "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346--359, 2008

- **FAST (corner detector)**

  Rosten. Machine Learning for High-speed Corner Detection, 2006.

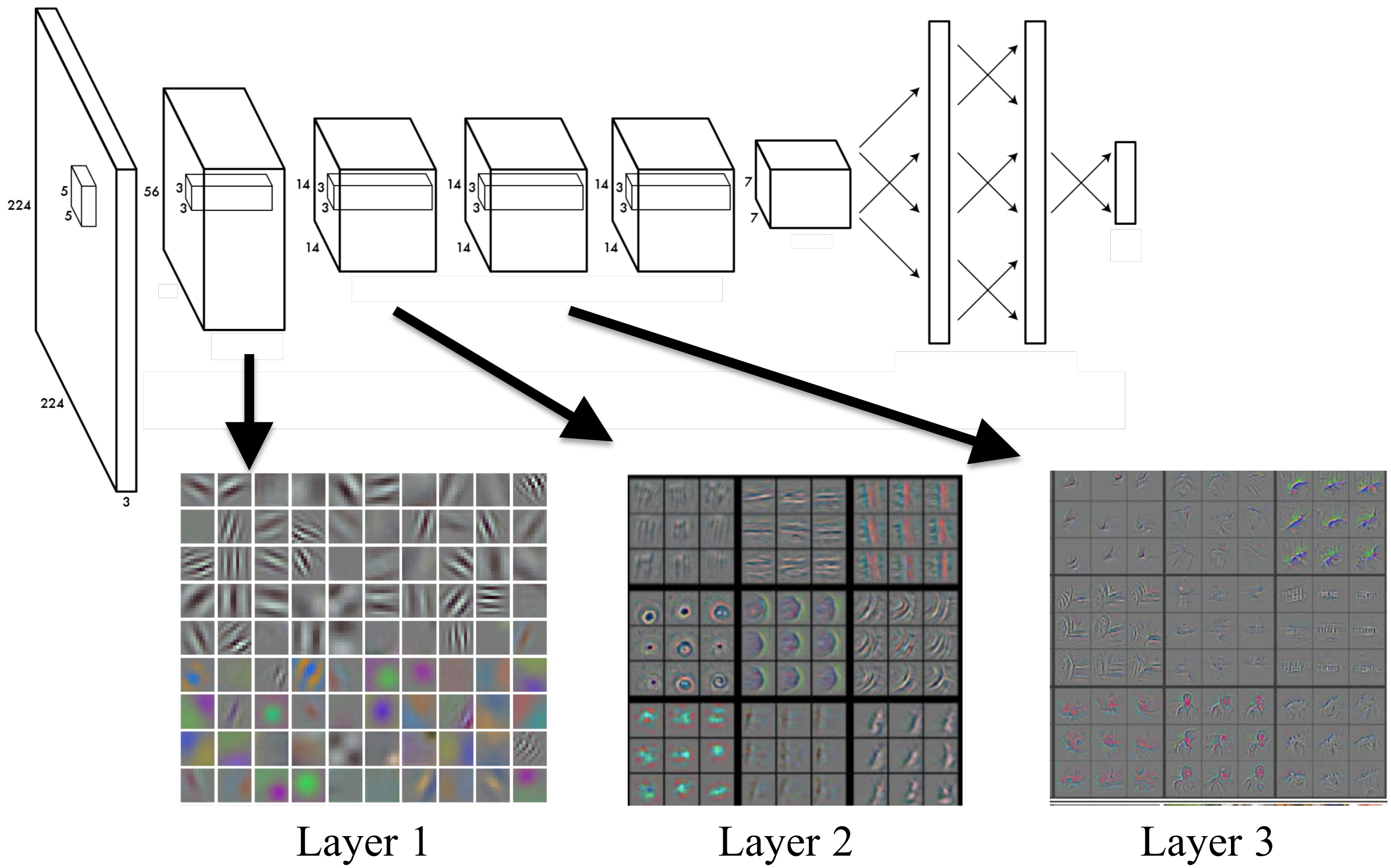
- **ORB: an efficient alternative to SIFT or SURF**

  Ethan Rublee, Vincent Rabaud, Kurt Konolige, Gary R. Bradski: ORB: An efficient alternative to SIFT or SURF. ICCV 2011

- **Fast Retina Key- point (FREAK)**

  A. Alahi, R. Ortiz, and P. Vandergheynst. FREAK: Fast Retina Keypoint. In IEEE Conference on Computer Vision and Pattern Recognition, 2012. CVPR 2012 Open Source Award Winner.

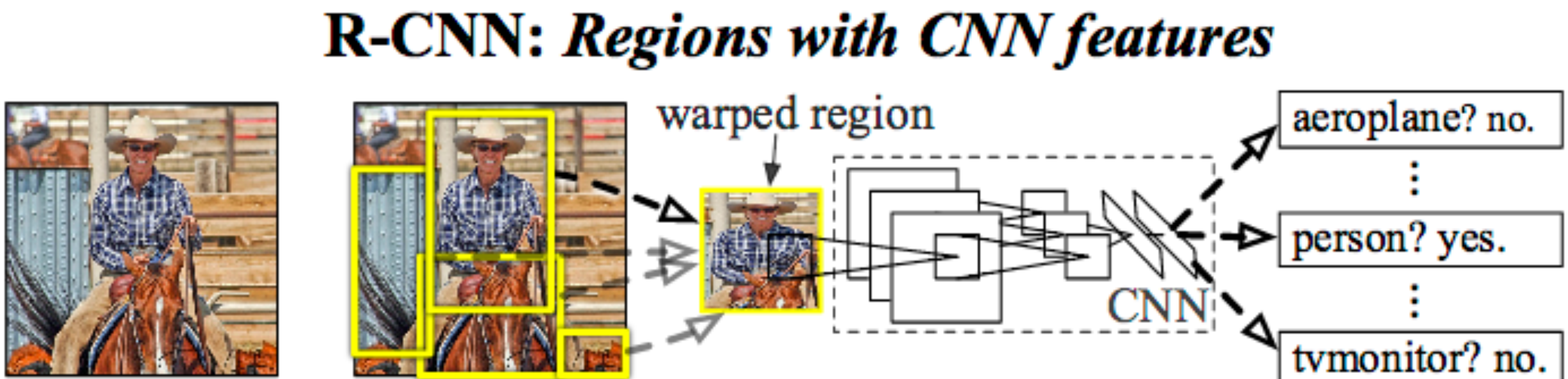
Using CNNs to detect and describe features



[Zeiler & Fergus ECCV 14]

Object detection using CNN features!

Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. R. Girshick, J. Donahue, T. Darrell, J. Malik, 2014



Next lecture:

Introduction to recognition