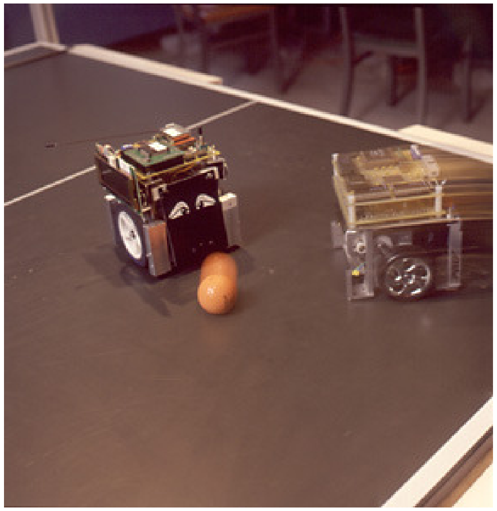
CS 4758/6758: Robot Learning

Spring 2010: Lecture 3.

Ashutosh Saxena

Slides coutesy: Prof Noah Snavely, Yung-Yu Chung, Frédo Durand, Alexei Efros, William Freeman, Svetlana Lazebnik, Srinivasa Narasimhan, Steve Seitz, Richard Szeliski, and Li Zhang


ASIMO
HONDA



The environment

Camera as sensor

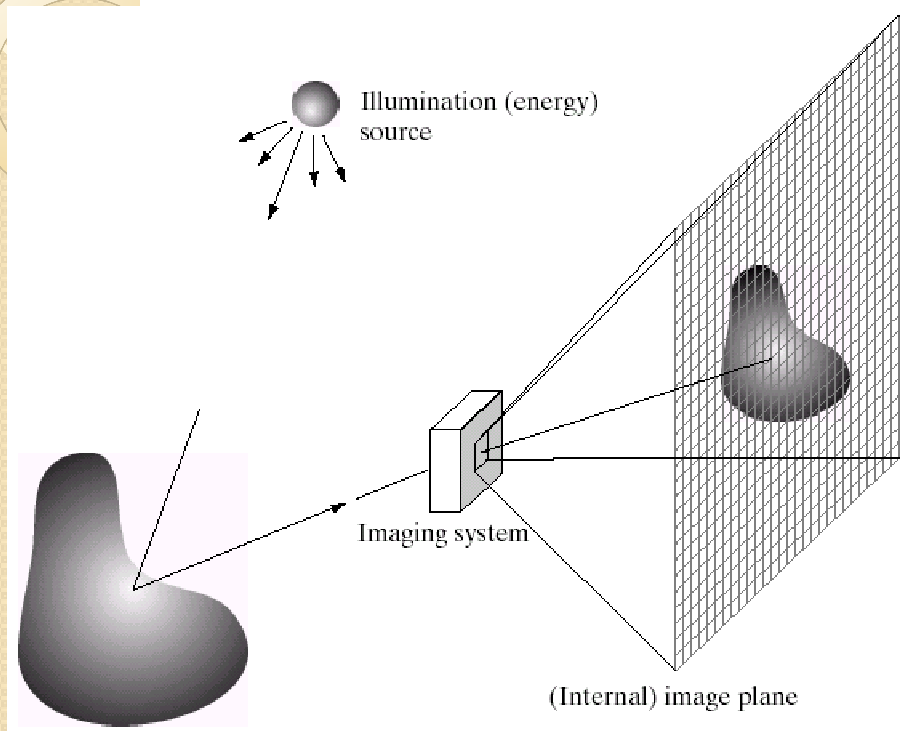
- Image and signal processing.

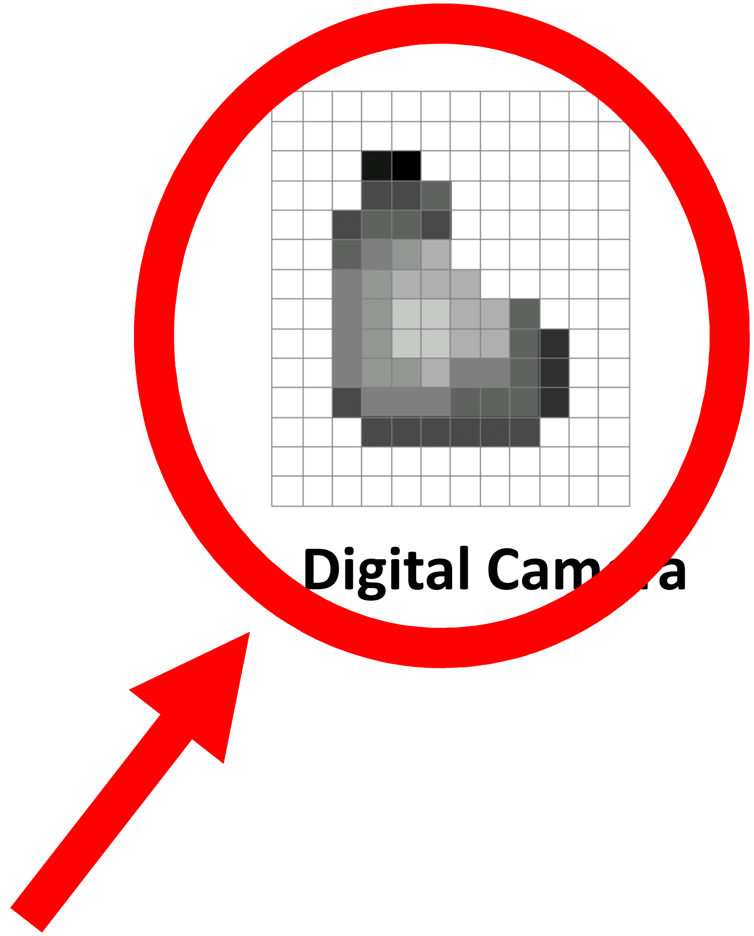
Implementation:

- OpenCV for processing the Image signals.

- Other libraries for processing 1D signals.

What is an image?



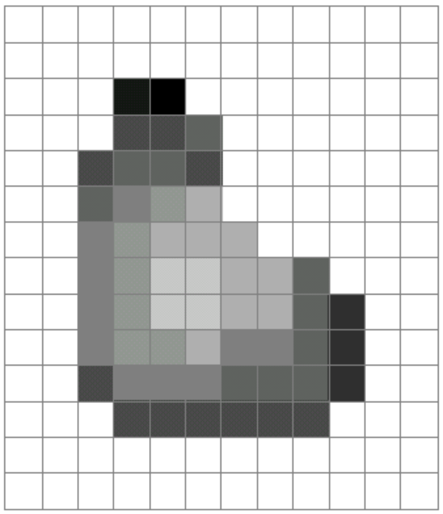


We get this as the input data

Source: A. Efros

What is an image?

- A grid of intensity values

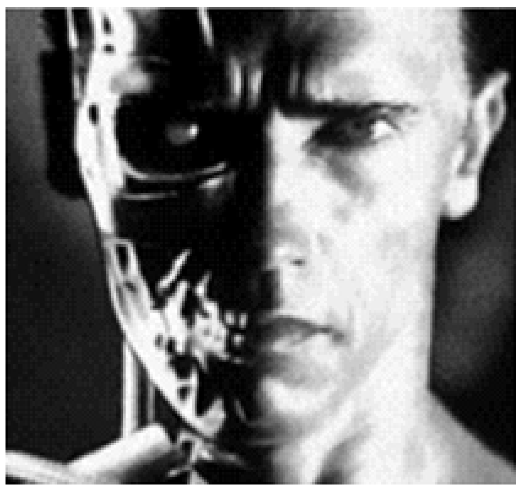


255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	20	0	255	255	255	255	255	255	255
255	255	255	75	75	75	255	255	255	255	255	255
255	255	75	95	95	75	255	255	255	255	255	255
255	255	96	127	145	175	255	255	255	255	255	255
255	255	127	145	175	175	175	255	255	255	255	255
255	255	127	145	200	200	175	175	95	255	255	255
255	255	127	145	200	200	175	175	95	47	255	255
255	255	127	145	145	175	127	127	95	47	255	255
255	255	74	127	127	127	95	95	95	47	255	255
255	255	255	74	74	74	74	74	74	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255

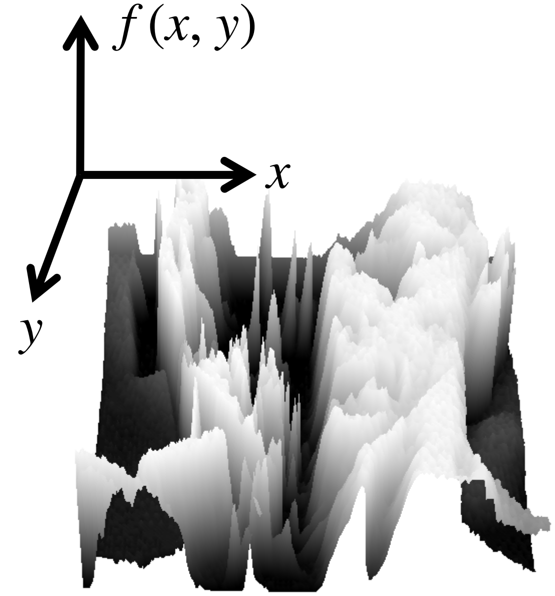
(common to use one byte per value: 0 = black, 255 = white)

What is an image?

- We can think of a (grayscale) image as a **function**, f, from R^2 to R:
  - $f(x,y)$ gives the **intensity** at position (x,y)



snoop

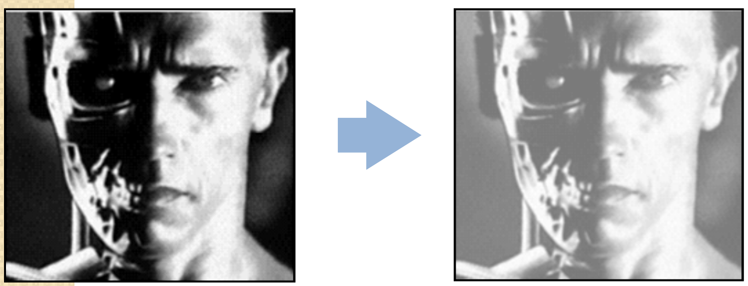


3D view

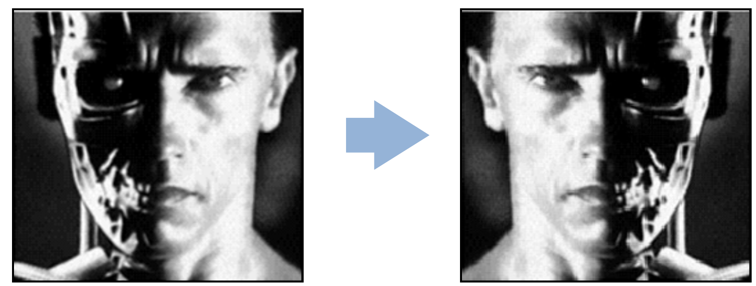
  - A **digital** image is a discrete (**sampled**, **quantized**) version of this function

Image transformations

- As with any function, we can apply operators to an image



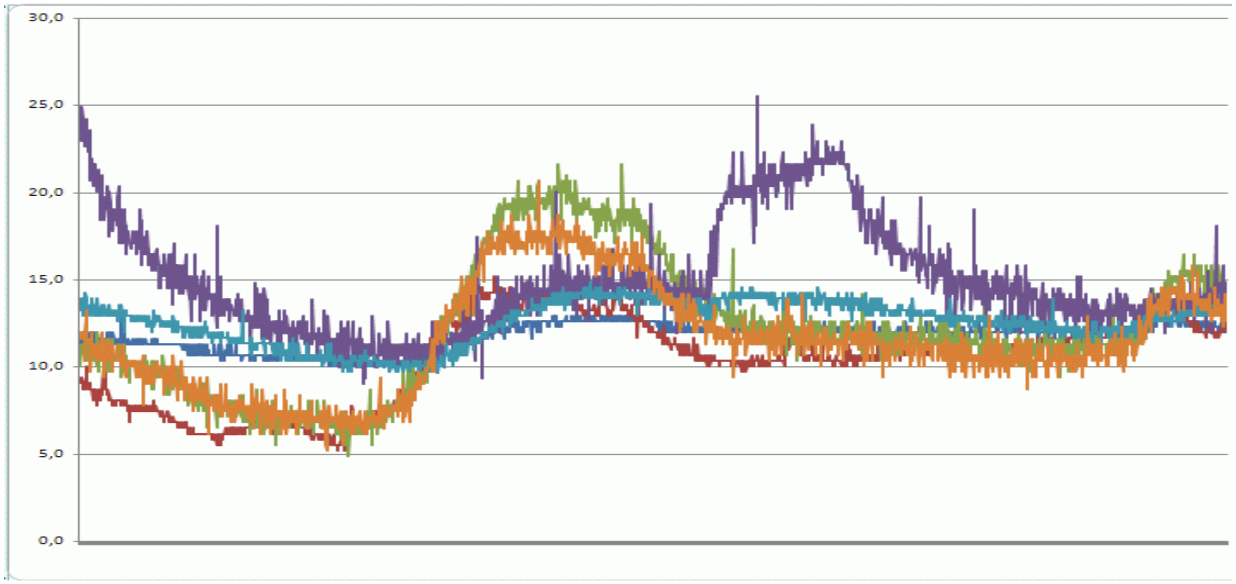
$g(x,y) = f(x,y) + 20$



$g(x,y) = f(-x,y)$

- We'll talk about a special kind of operator, *convolution* (linear filtering)

1D signal



255	200	178	100	74	67	71	101	120	180	211	240

Question: Noise reduction

- Given a camera and a still scene, how can you reduce noise?

Take lots of images and average them!

What’s the next best thing?

Source: S. Seitz

Image filtering

- Modify the pixels in an image based on some function of a local neighborhood of each pixel

Source: L. Zhang

Linear filtering

- One simple version: linear filtering (cross-correlation, convolution)
  - Replace each pixel by a linear combination of its neighbors
- The prescription for the linear combination is called the "kernel" (or "mask", "filter")

10	5	3
4	6	1
1	1	8

Local image data

0	0	0
0	0.5	0
0	1	0.5

kernel

	8	

Modified image data

Cross-correlation

Let F be the image, H be the kernel (of size 2k+1 x 2k+1), and G be the output image

$$G[i,j] = \sum_{u=-k}^{k} \sum_{v=-k}^{k} H[u,v] F[i+u, j+v]$$

This is called a **cross-correlation** operation:

$$G = H \otimes F$$

Convolution

- Same as cross-correlation, except that the kernel is "flipped" (horizontally and vertically)

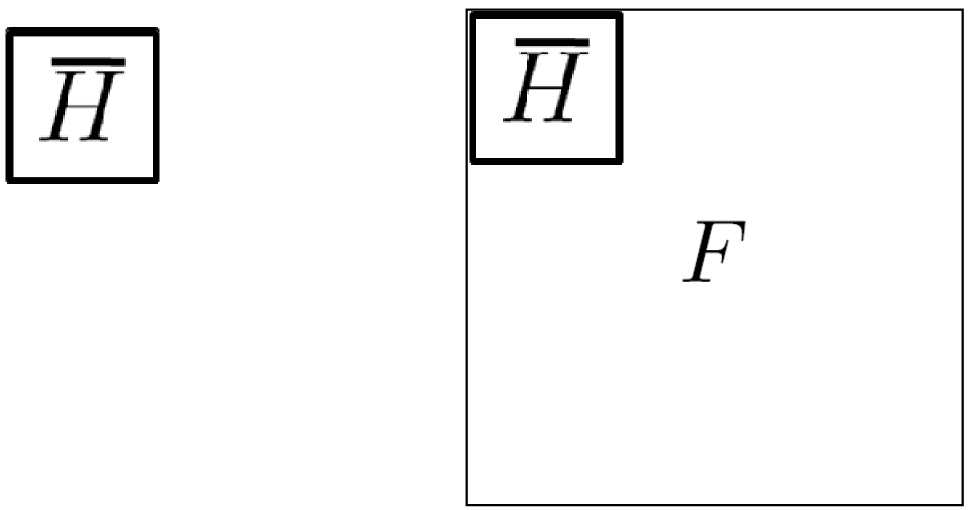
$$G[i,j] = \sum_{u=-k}^{k} \sum_{v=-k}^{k} H[u,v]F[i-u,j-v]$$

This is called a **convolution** operation:

$$G = H * F$$

- Convolution / cross-correlation are **commutative** and **associative**

Convolution



Adapted from F. Durand

Mean filtering

H $*$

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

F

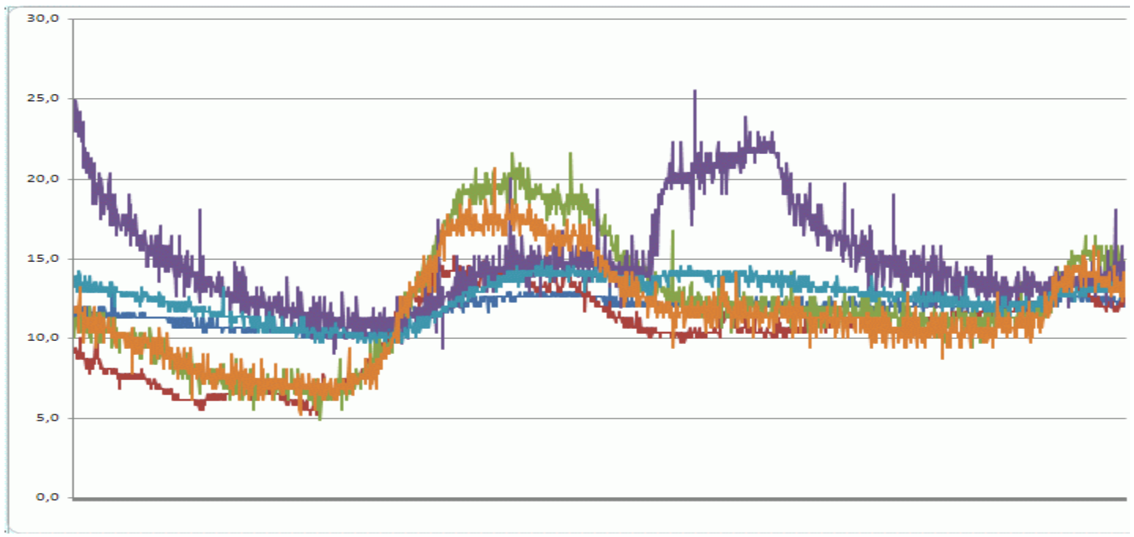
=

	0	10	20	30	30	30	20	10	
	0	20	40	60	60	60	40	20	
	0	30	60	90	90	90	60	30	
	0	30	50	80	80	90	60	30	
	0	30	50	80	80	90	60	30	
	0	20	30	50	50	60	40	20	
	10	20	30	30	30	30	20	10	
	10	10	10	0	0	0	0	0	

G

Mean Filtering: 1-D

One can also apply convolution to 1D signals.



F = [0,10, 12, 20, 8, 12, 0]

H = [.25 .5 .25]

G = ?

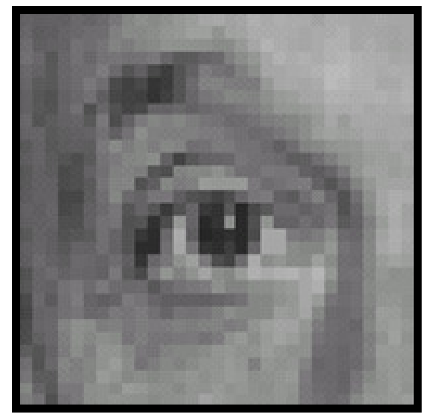
Linear filters: examples

Original

*

0	0	0
0	1	0
0	0	0

=



Identical image

Source: D. Lowe

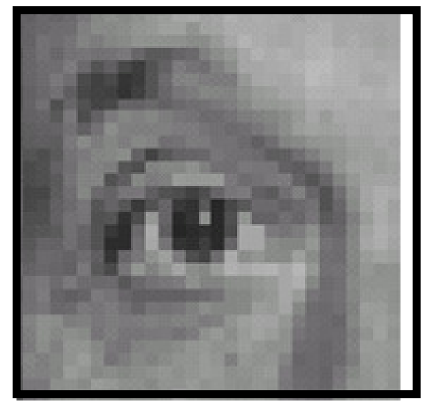
Linear filters: examples

Original

*

0	0	0
1	0	0
0	0	0

=



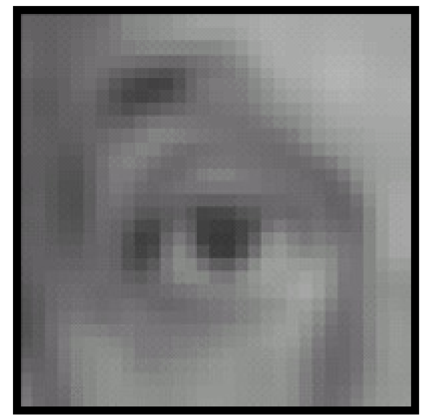
Shifted left
By 1 pixel

Source: D. Lowe

Linear filters: examples

Original

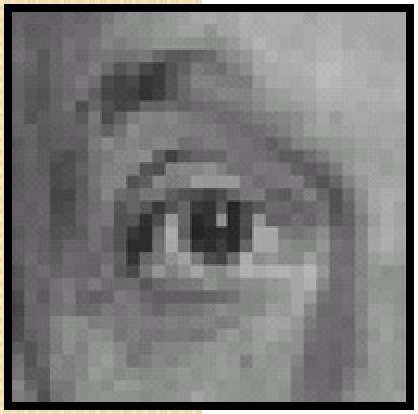
$$* \ \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} =$$



Blur (with a mean filter)

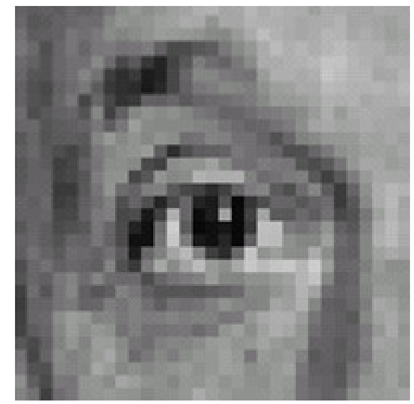
Source: D. Lowe

Linear filters: examples



Original

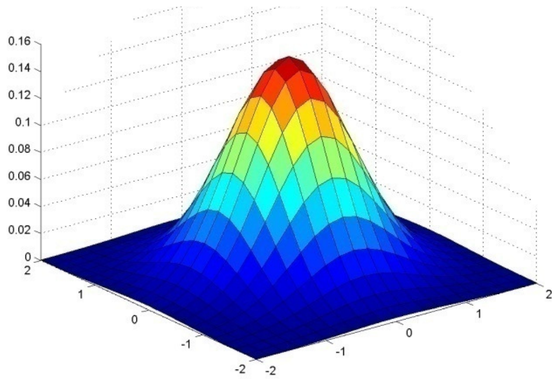
$$* \left(\begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \right) =$$

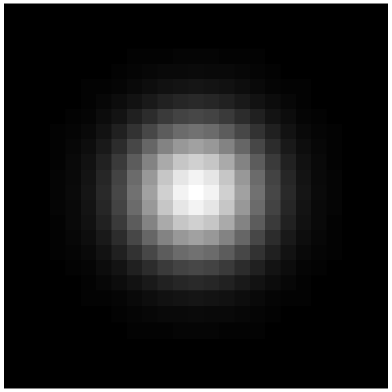


Sharpening filter
(accentuates edges)

Source: D. Lowe

Gaussian Kernel

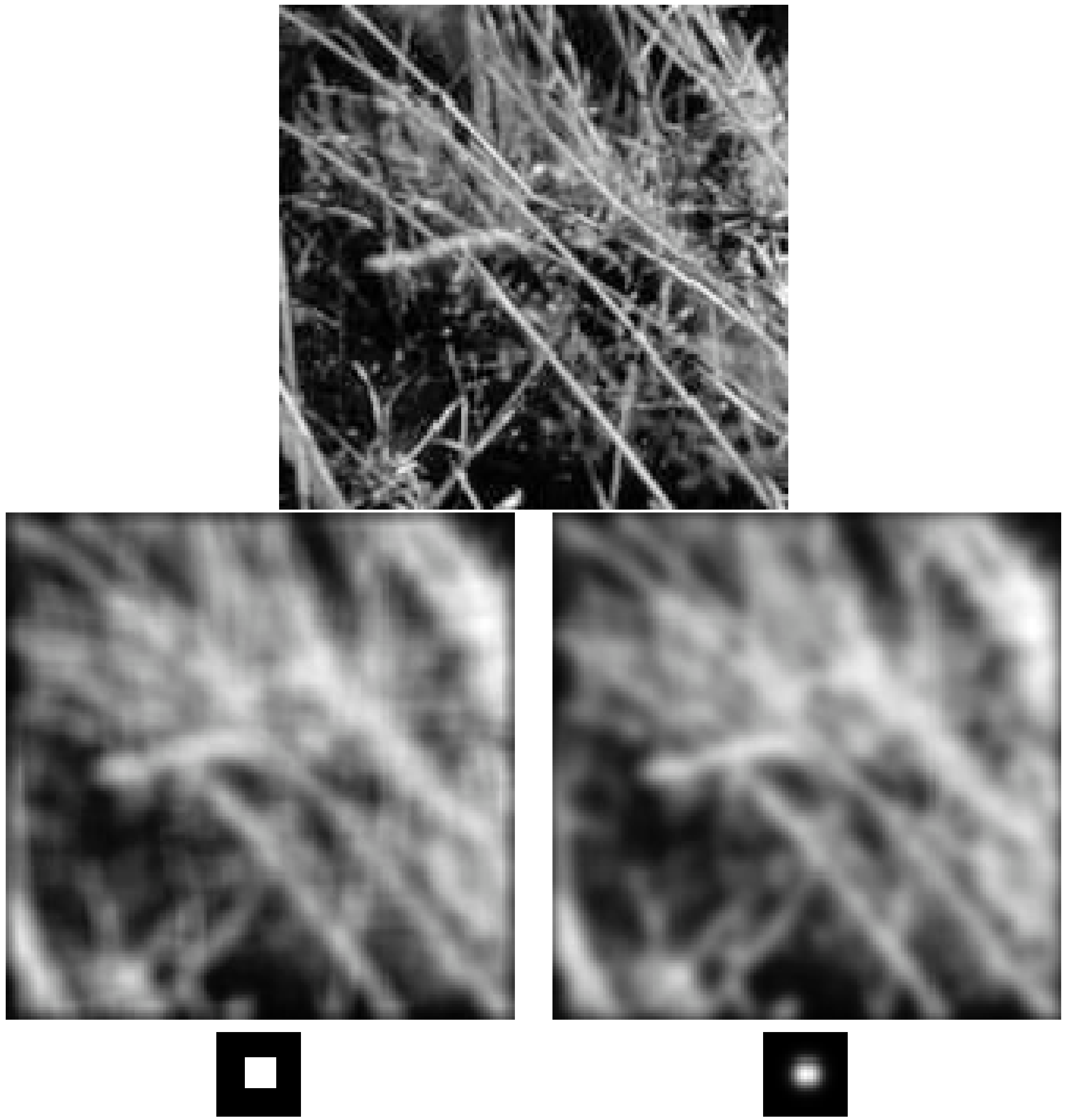




$$G_\sigma = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

Source: C. Rasmussen

Mean vs. Gaussian filtering



Gaussian noise

$F[x, y] + \mathcal{N}(0, 5\%)$

σ = 1 pixel

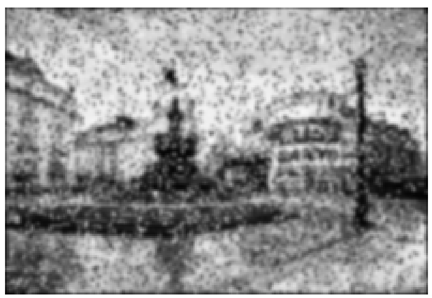
σ = 2 pixels

σ = 5 pixels

Smoothing with larger standard deviations suppresses noise, but also blurs the image

Outliers noise – Gaussian blur

p = 10%



σ = 1 pixel

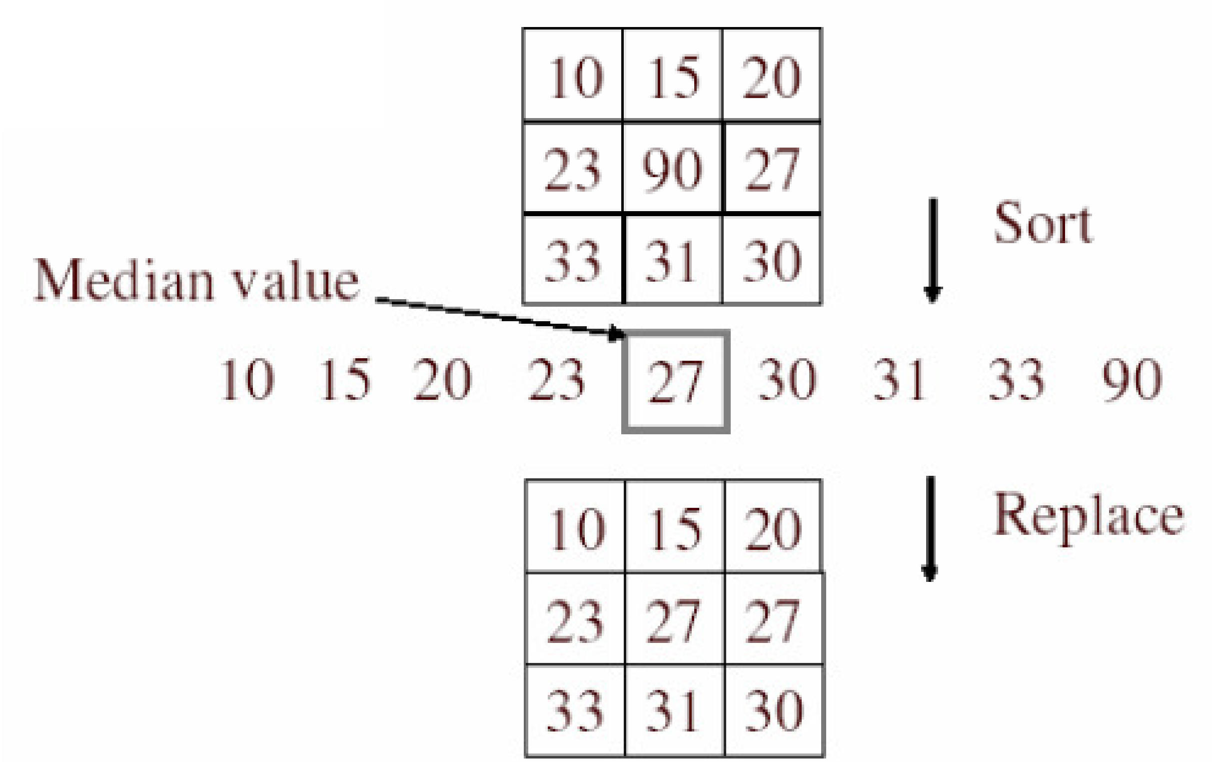
σ = 2 pixels

σ = 5 pixels

- What’s wrong with the results?

Alternative idea: Median filtering

- A **median filter** operates over a window by selecting the median intensity in the window

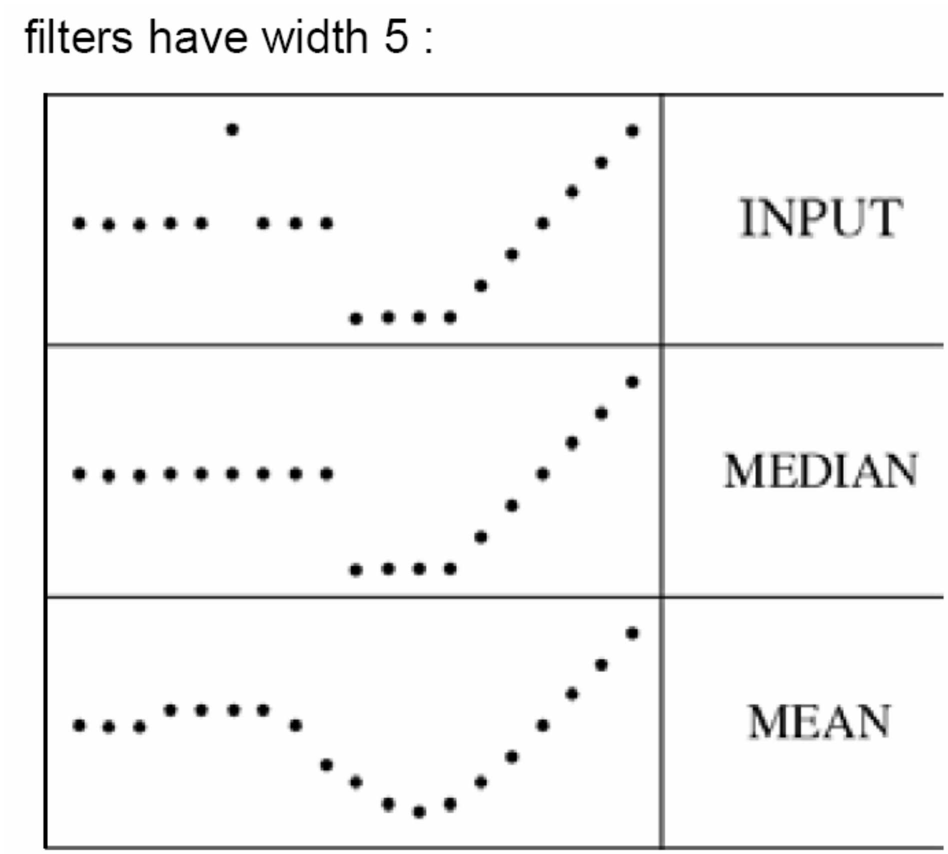


- Is median filtering linear?

Source: K. Grauman

Median filter

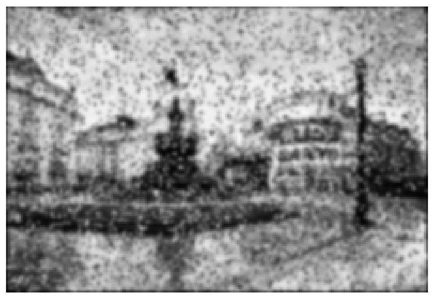
- What advantage does median filtering have over Gaussian filtering?



Source: K. Grauman

Salt & pepper noise – median filtering

p = 10%



σ = 1 pixel

σ = 2 pixels

σ = 5 pixels

3x3 window

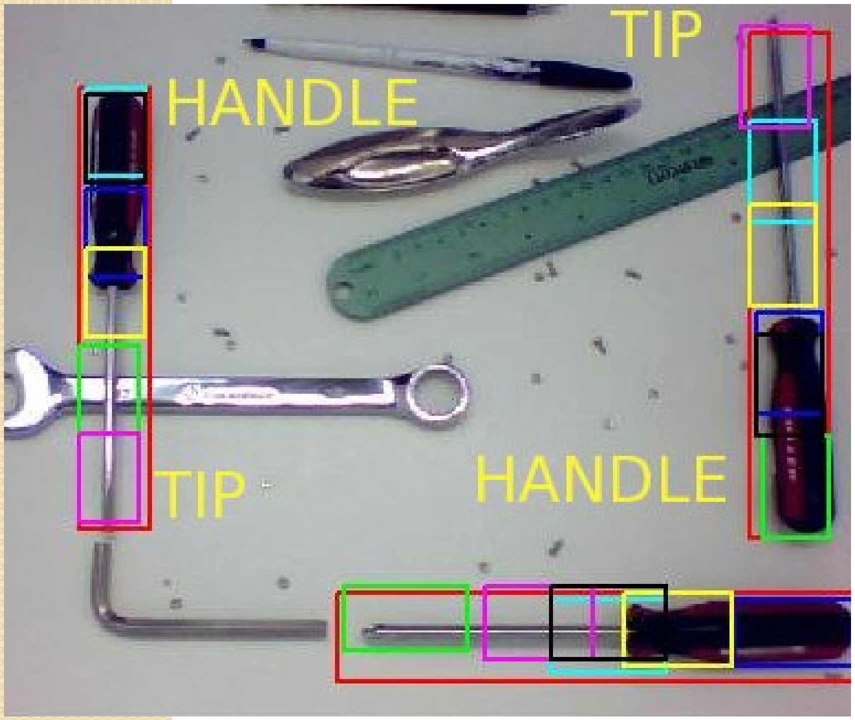
5x5 window

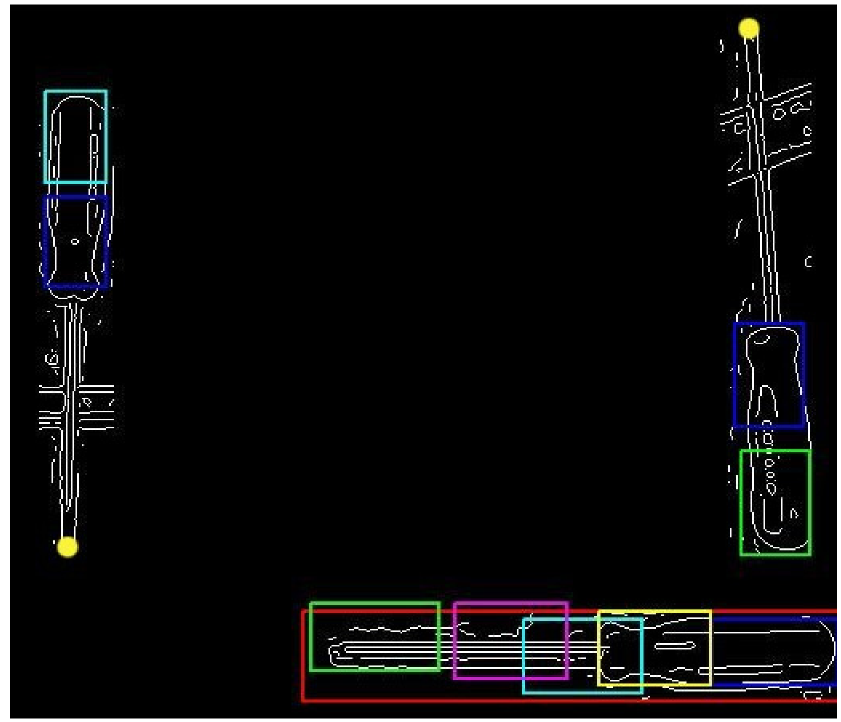
7x7 window

Questions?

Edge Detection

Edge detection

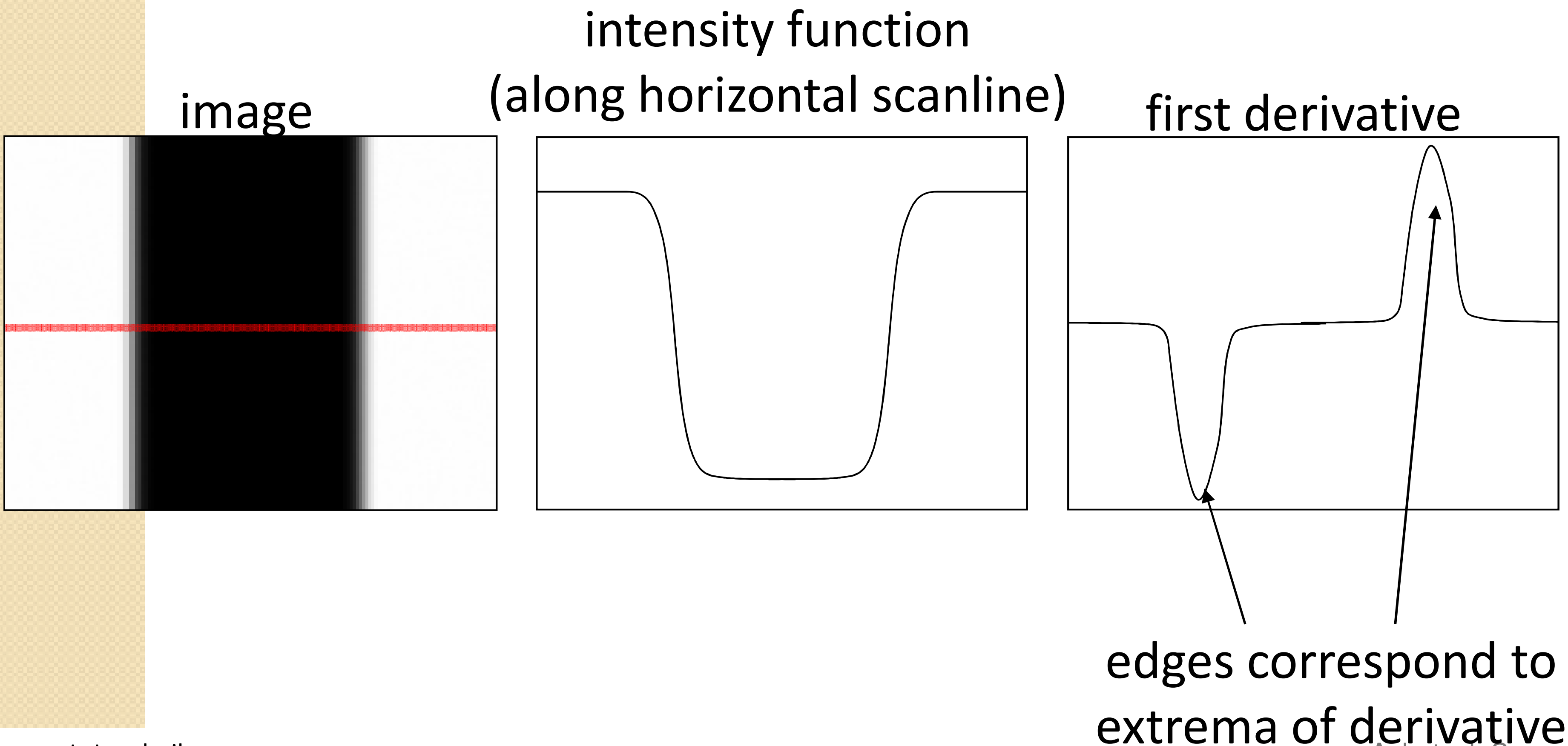




- Convert a 2D image into a set of curves
  - Extracts salient features of the scene
  - More compact than pixels

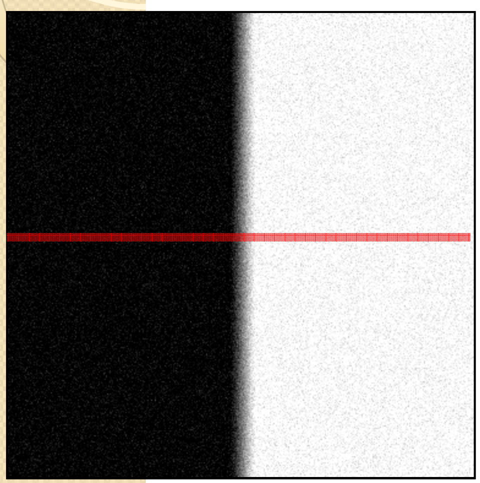
Characterizing edges

- An edge is a place of rapid change in the image intensity function

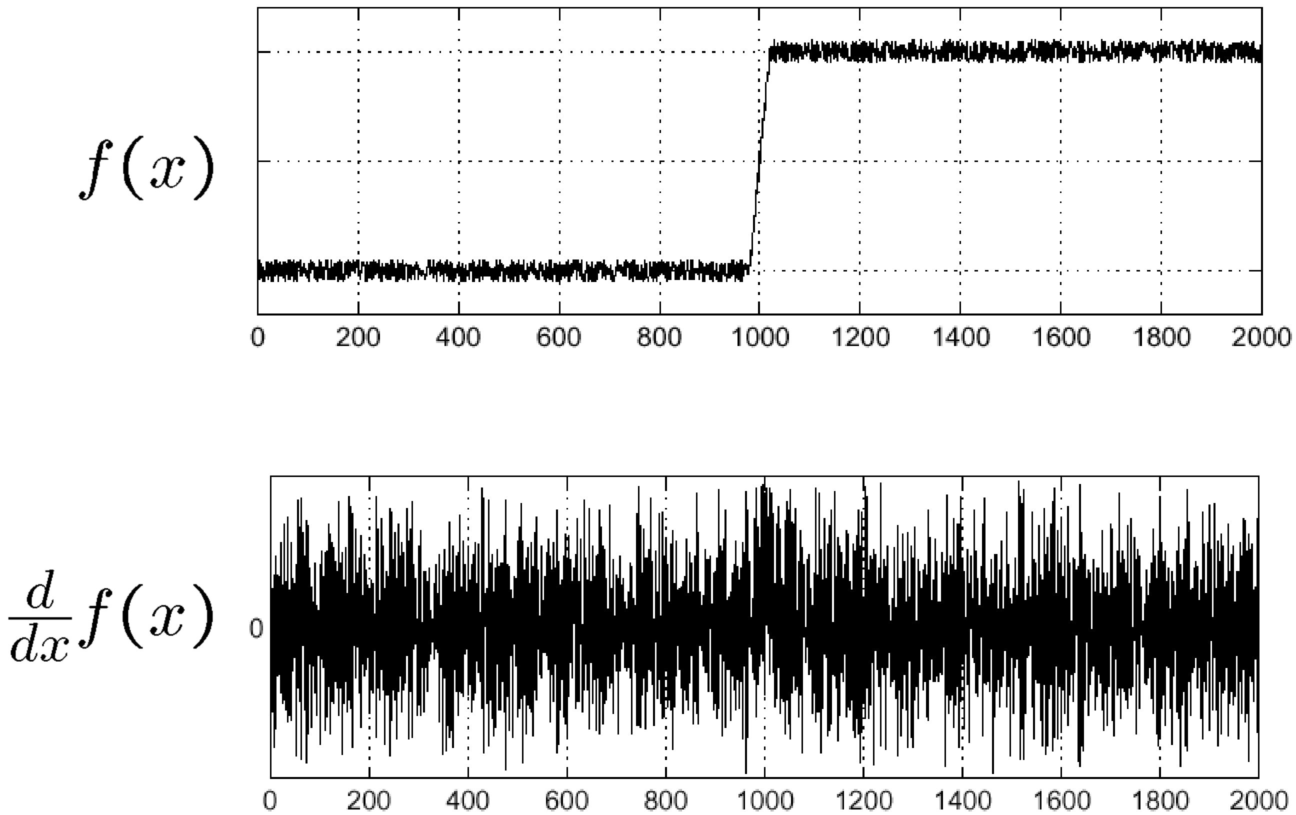


Source: L. Lazebnik

Effects of noise



Noisy input image



Where is the edge?

Source: S. Seitz

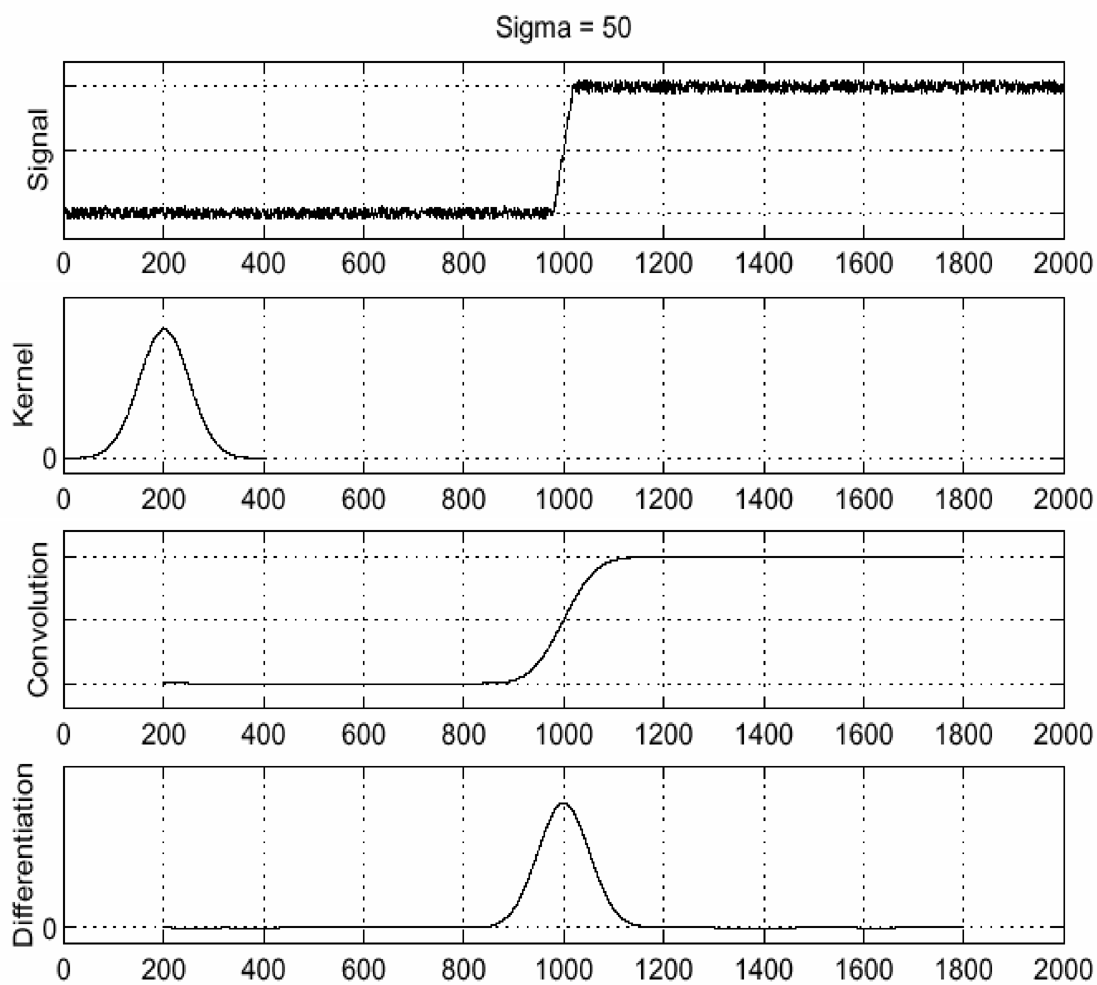
Solution: smooth first

f

h

$f * h$

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



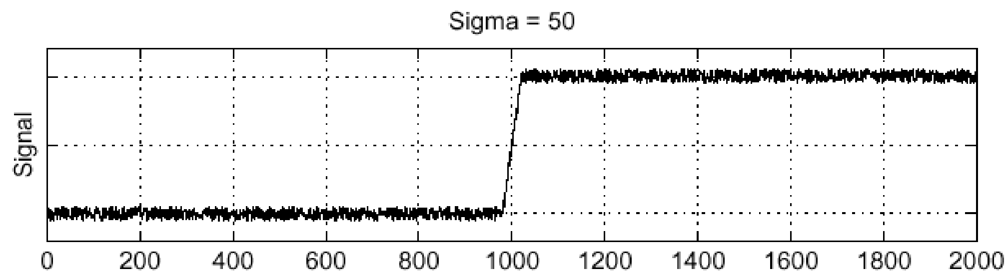
To find edges, look for peaks in $\frac{d}{dx}(f * h)$

Source: S. Seitz

Associative property of convolution

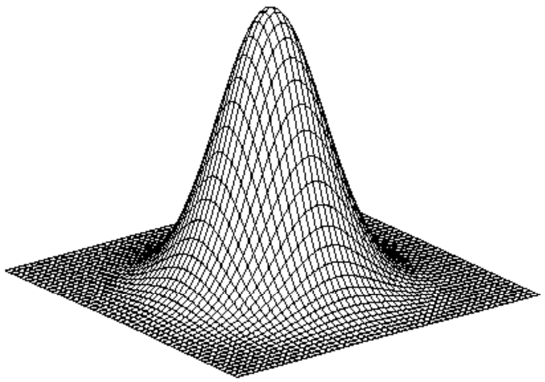
- Differentiation is convolution, and convolution is associative: $\frac{d}{dx}(f * h) = f * \frac{d}{dx}h$
- This saves us one operation:

f



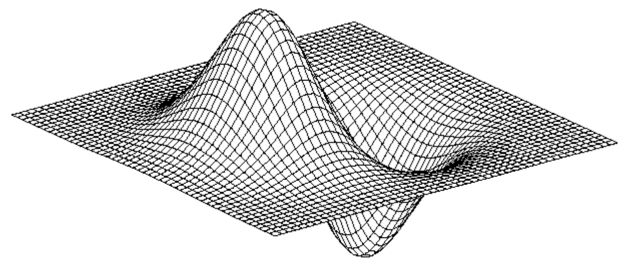
Source: S. Seitz

2D edge detection filters



Gaussian

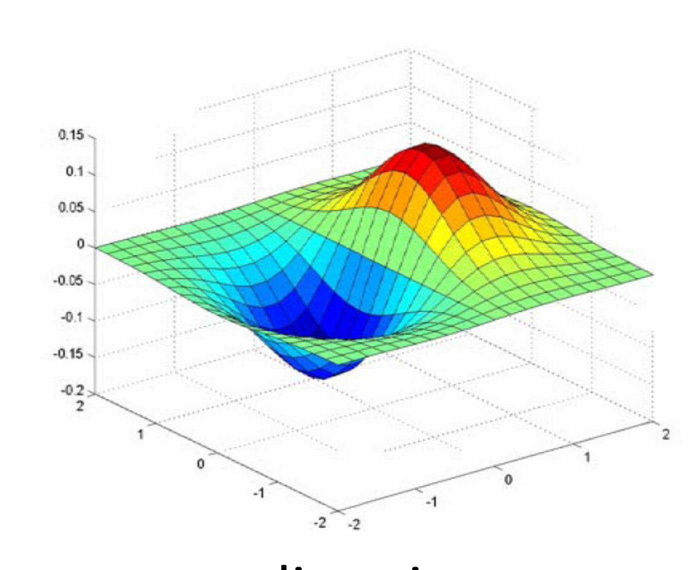
$$h_\sigma(u,v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2\sigma^2}}$$

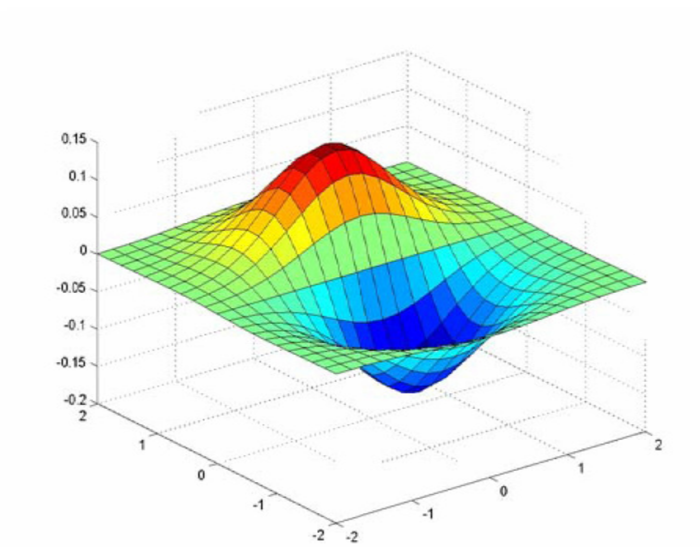


derivative of Gaussian (x)

$$\frac{\partial}{\partial x} h_\sigma(u,v)$$

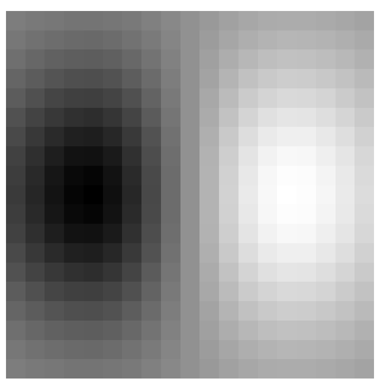
Derivative of Gaussian filter

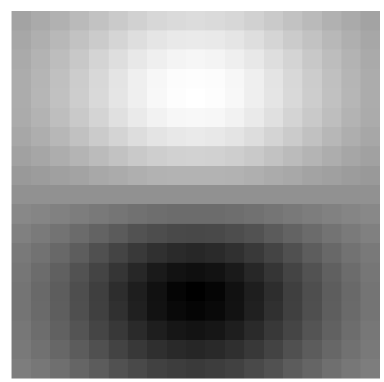




x-direction

y-direction





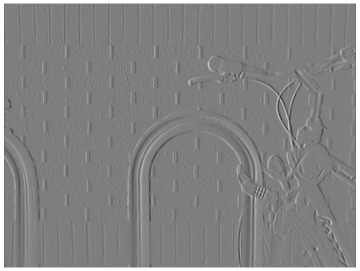
The Sobel operator

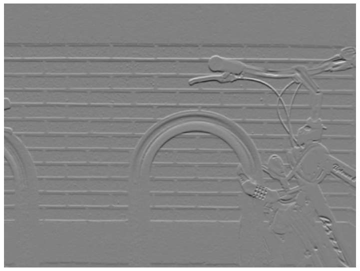
- Common approximation of derivative of Gaussian

$$\frac{1}{8}\begin{array}{|c|c|c|}\hline -1 & 0 & 1 \\ \hline -2 & 0 & 2 \\ \hline -1 & 0 & 1 \\ \hline \end{array} \qquad \frac{1}{8}\begin{array}{|c|c|c|}\hline 1 & 2 & 1 \\ \hline 0 & 0 & 0 \\ \hline -1 & -2 & -1 \\ \hline \end{array}$$

$$s_x \qquad\qquad\qquad s_y$$

Sobel operator: example



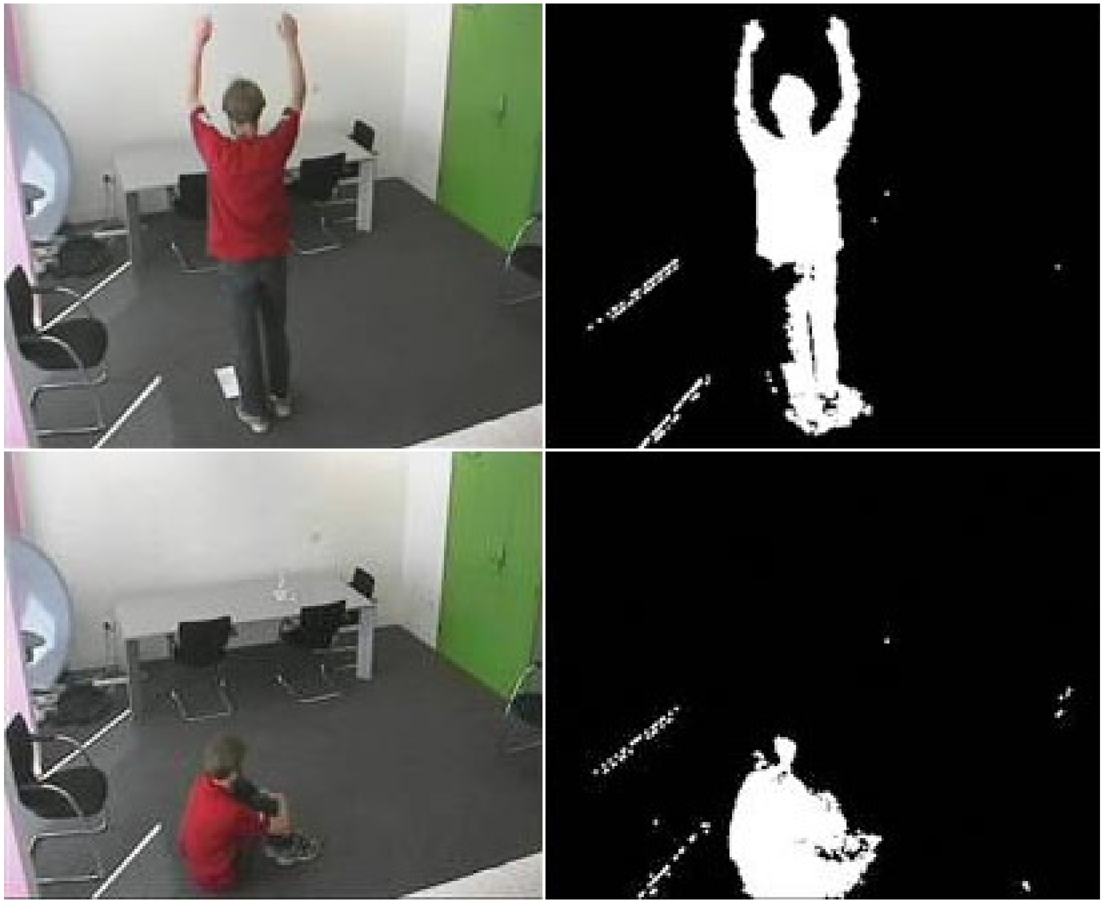


Source: Wikipedia

Questions?

Finding Objects

- Background subtraction



Feature extraction: Corners and blobs

Desirable properties in the features

Distinctiveness:

- can differentiate a large database of objects

Efficiency

- real-time performance achievable

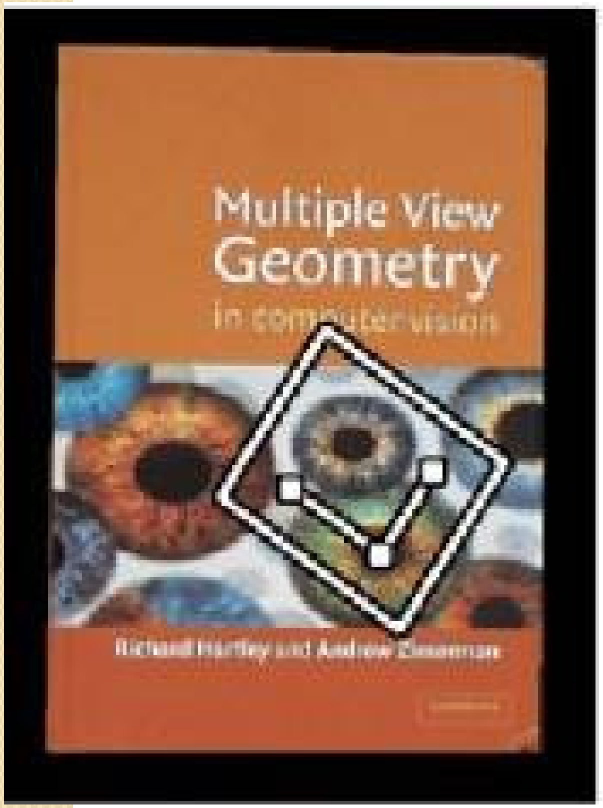
Example of features

A laundry list:

- Corner / edge detectors
- SIFT features

- Output of various filters…

Feature Matching



Feature Matching

Ashutosh Saxena

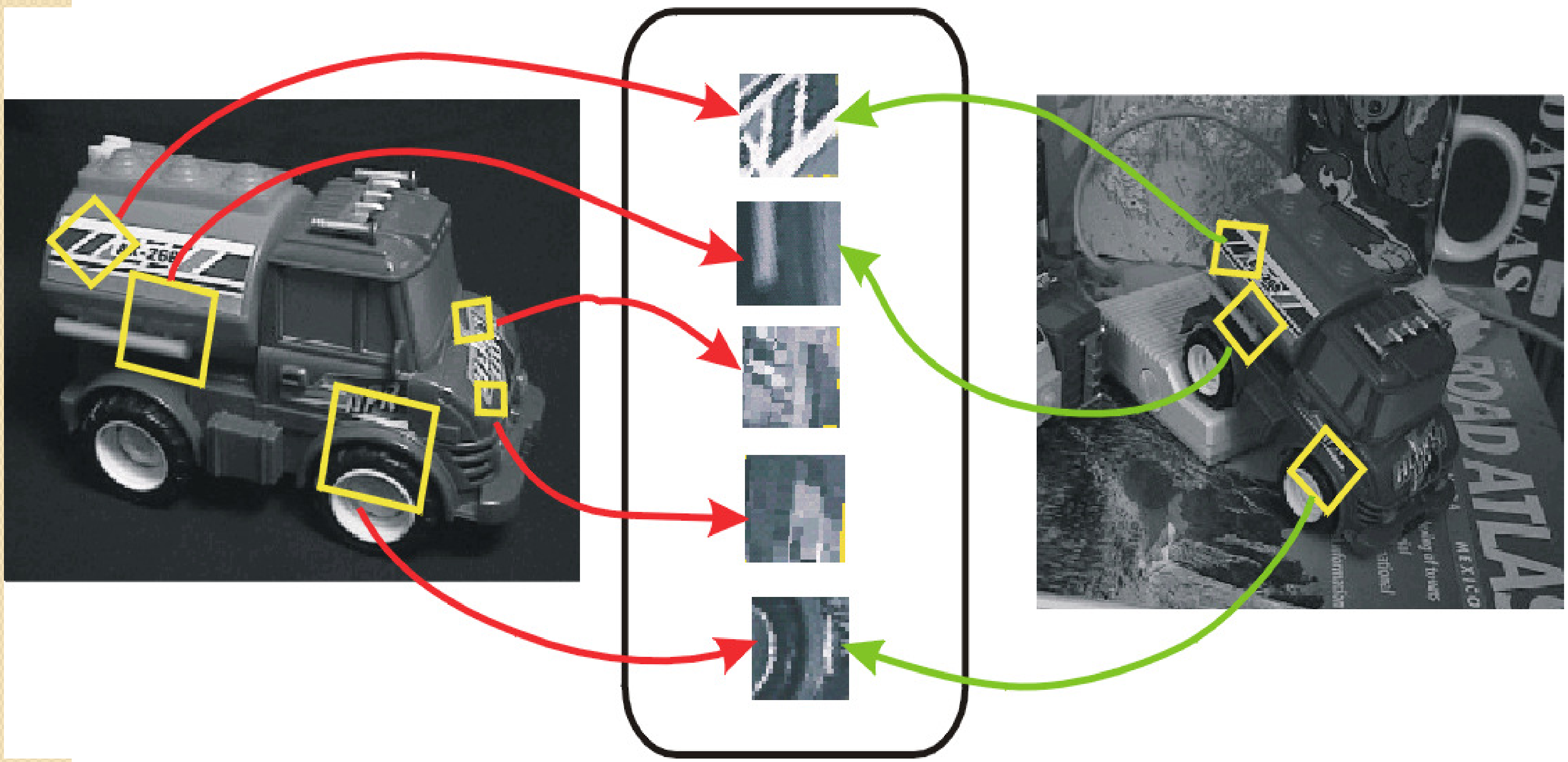
Metric for similarity?

- Vector x_i and x_j .
- What is the distance between them?

Matching using distance between the features

Find features that are invariant to transformations

- geometric invariance: translation, rotation, scale
- photometric invariance: brightness, exposure, …



Feature Descriptors

Object recognition (David Lowe)

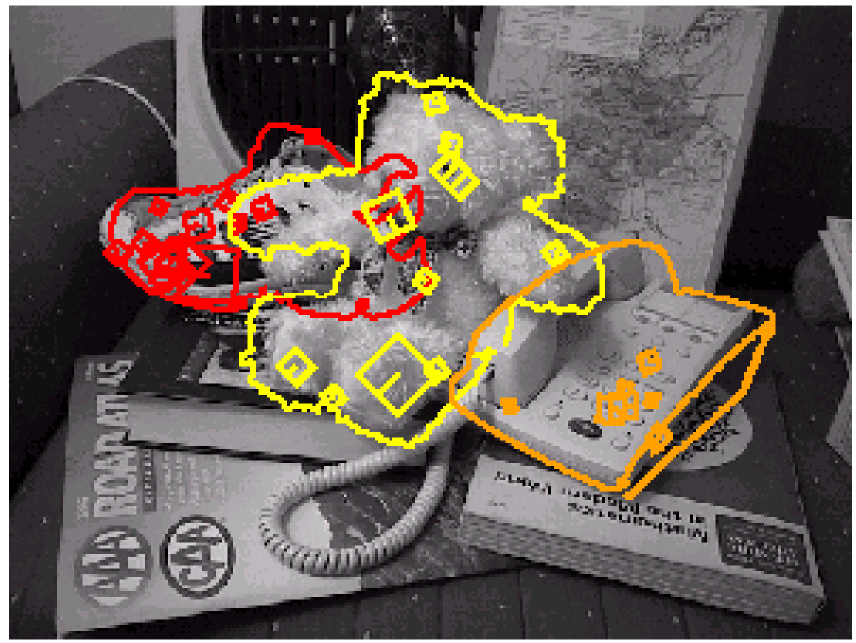


Image matching

by Diva Sian

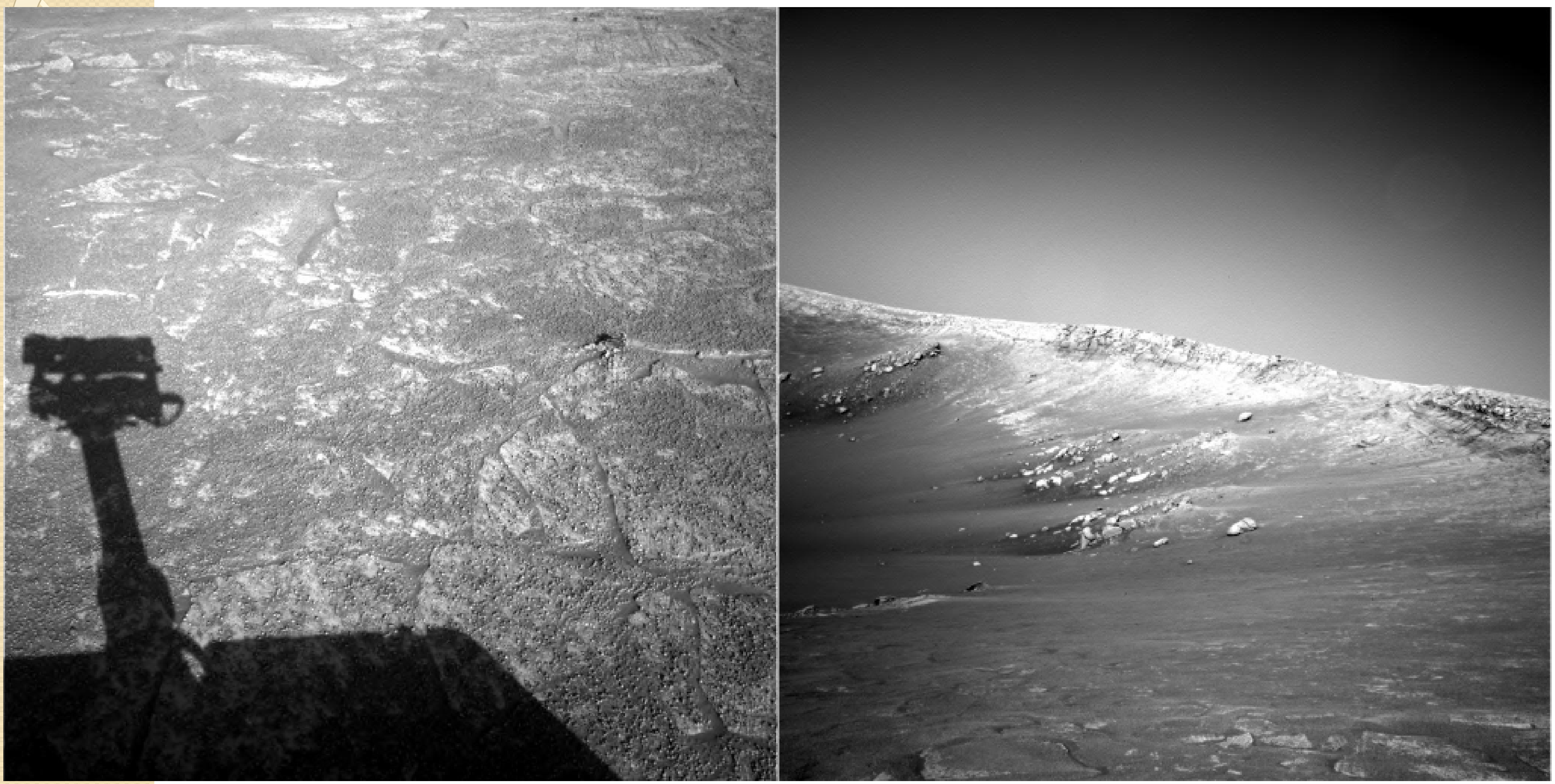
by swashford

Harder case

by Diva Sian

by scgbt

Harder still?



NASA Mars Rover images

How to match features?

- Robustness?

Machine Learning to the rescue.

Supervised Learning: next lecture.

Projects

- Project proposals due **Feb 15**.
  - Brief description of the projects on Thursday lecture.
  - Choose a Topic and a Robot.
- Good time to setup a meeting with the instructor next week.

Questions?