Course: Deep Learning

Unit 2: Computer Vision

Fundamentals of image processing and object recognition

Luis Baumela

Universidad Politécnica de Madrid

Fundamentals of Imge processing and Object Recognition

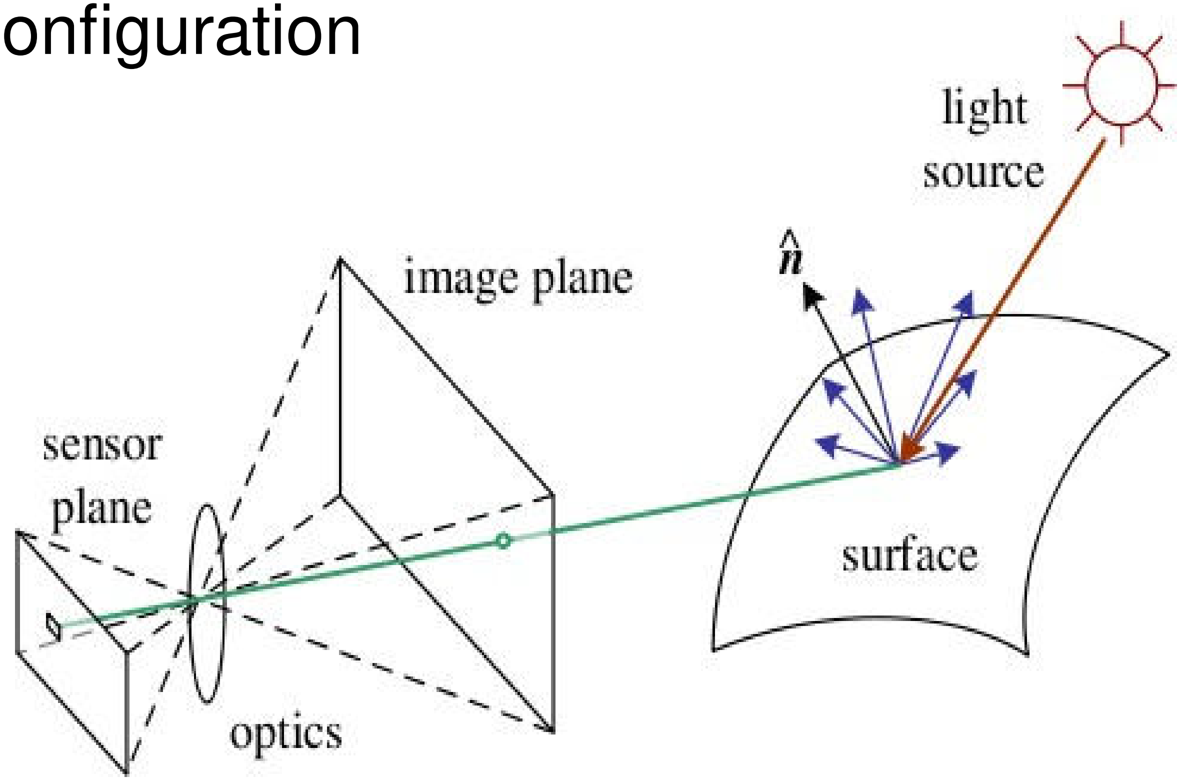
1. Introduction to image processing
    - Image formation
    - Convolution
    - Gradient and Laplacian of an image

2. Shallow object recognition approach
    - Bag of words

Introduction to image processing

- Image formation

  Image gray levels depend on:
  - Illumination
  - Scene geometry
  - Object surface properties
  - Camera configuration



Introduction to image processing

- Digital image

Let $\mathcal{F}$ be the set of image **rows** and $\mathcal{C}$ the set of **colums**

$$\mathcal{F} = \{0, \ldots, f-1\}$$
$$\mathcal{C} = \{0, \ldots, c-1\}$$

$f \times c$ is the **spatial resolution**.

	0							$c-1$
0	45	60	98	127	132	133	137	133
	46	65	98	123	126	128	131	133
	47	65	96	115	119	123	135	137
	47	63	91	107	113	122	138	134
	50	59	80	97	110	123	133	134
	49	53	68	83	97	113	128	133
	50	50	58	70	84	102	116	126
$f-1$	50	50	52	58	69	86	101	120

A digital image is a function

$$\mathrm{I} : \mathcal{F} \times \mathcal{C} \rightarrow \mathcal{D}$$

where

$$\mathcal{D} = \{0, \ldots, d-1\}$$

is the image **digital resolution**, typically

$$d = 256$$

Introduction to image processing

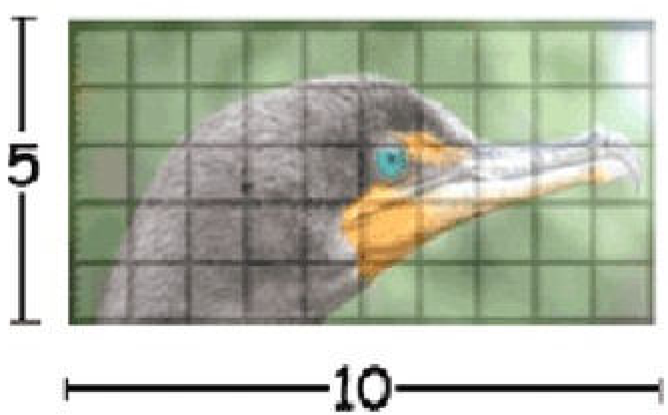
- Colour digital image

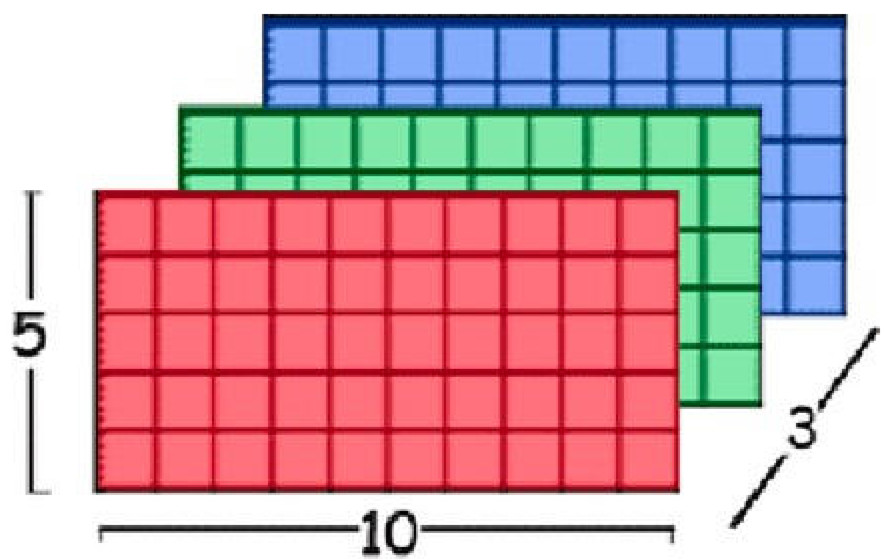
Let $\mathcal{F}$ be the set of image **rows** and $\mathcal{C}$ the set of **colums**

$\mathcal{F} = \{0, \ldots, f-1\}$ $\quad$ $f \times c$ is the **spatial resolution**.

$\mathcal{C} = \{0, \ldots, c-1\}$

A colour digital image is a function $\mathtt{I} : \mathcal{F} \times \mathcal{C} \rightarrow \mathcal{D}^3$

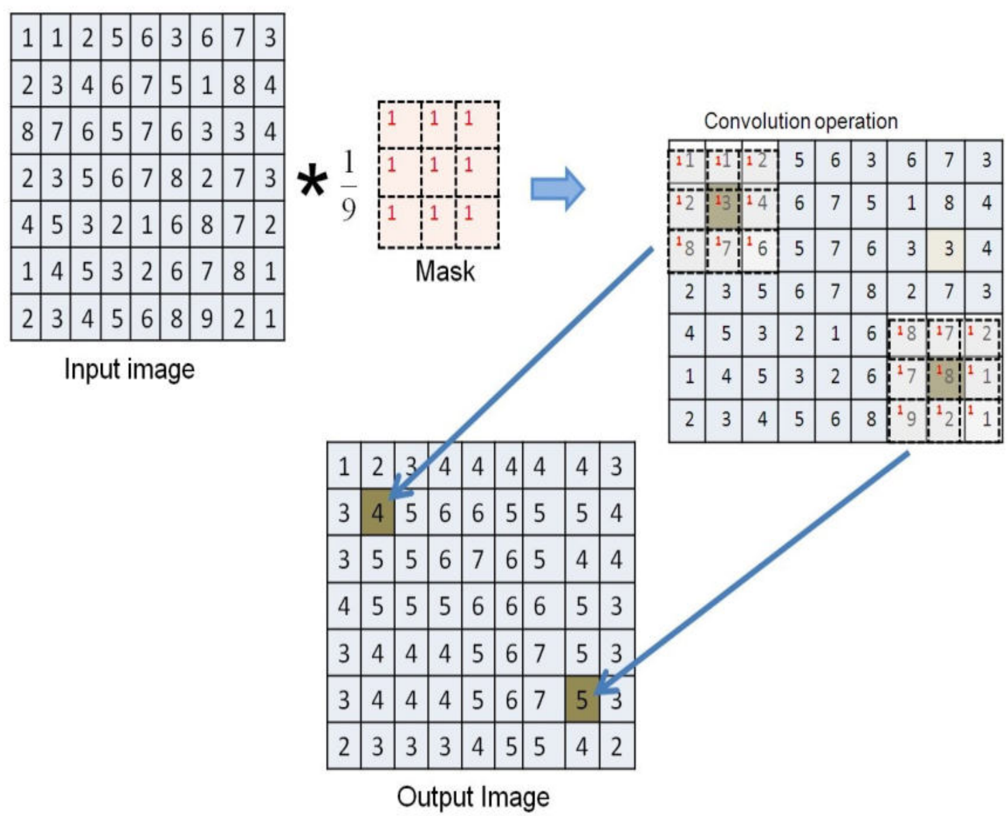




Introduction to image processing

- Discrete convolution

Let e and h be two discrete functions defined respectively in the domains $M \times N$ and $m \times n$. Convolution $o = h * e$ is a function of dimension $M + m - 1 \times N + n - 1$

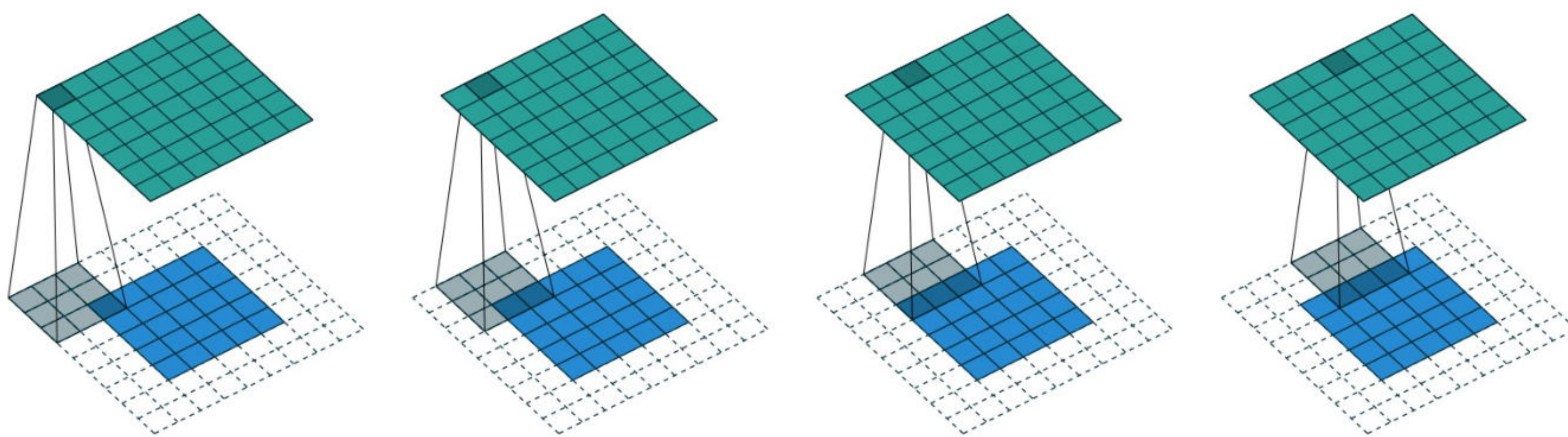


Introduction to image processing

- Discrete convolution

Let e and h be two discrete functions defined respectively in the domains $M \times N$ and $m \times n$. Convolution $o = h * e$ is a function of dimension $M + m - 1 \times N + n - 1$

$$o(r,t) = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} h(i,j) e_e[r + i - (m-1), t + j - (n-1)]$$



https://github.com/vdumoulin/conv_arithmetic

Introduction to image processing

- Discrete convolution with padding

Let e and h be two discrete functions defined respectively in the domains $M \times N$ and $m \times n$. Convolution $o = h * e$ is a function of dimension $M + m - 1 \times N + n - 1$

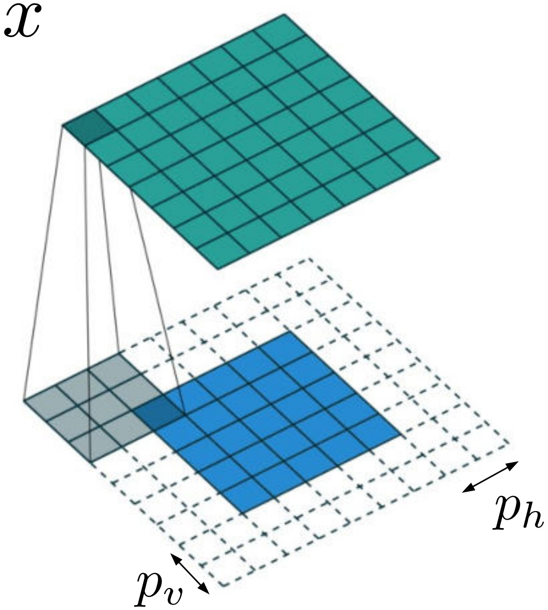
$$o(r,t) = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} h(i,j) e_e[r + i - (m-1), t + j - (n-1)]$$

Let p_x be the **padding** amount in dimension x where $x \in \{h, v\}$, o is a function of dimension

$$M + 2p_h - m + 1 \times N + 2p_v - n + 1$$

in this case, $p_h = m - 1,\ p_v = n - 1.$

termed **full / strict padding**



https://github.com/vdumoulin/conv_arithmetic

Introduction to image processing

- Discrete convolution / correlation

Half padding convolution

$$o(r,t) = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} h(i,j) e_e[r+i-(m-1)/2, t+j-(n-1)/2]$$

where o has dimension $M \times N$.

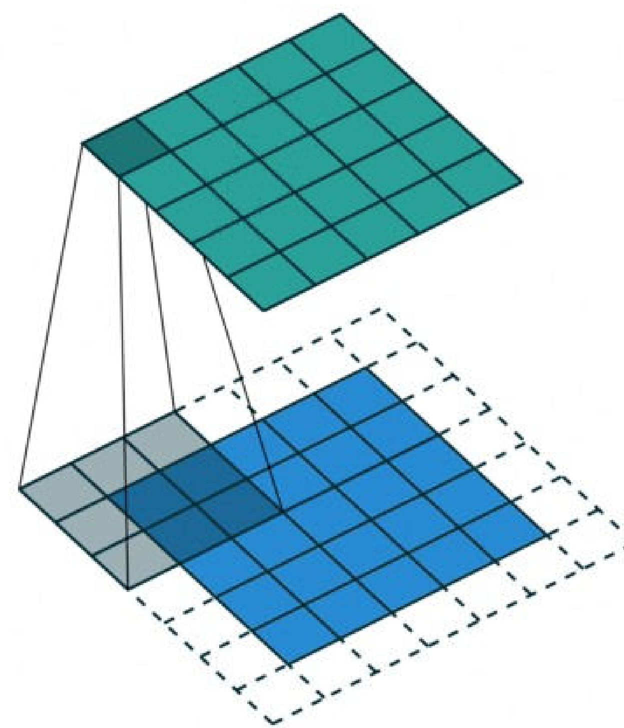
In the case of half padding

$$p_h = \frac{m-1}{2}, \qquad p_v = \frac{n-1}{2}$$

$$M + 2p_h - m + 1 = M$$

$$N + 2p_v - n + 1 = N$$

half padding / same convolution



https://github.com/vdumoulin/conv_arithmetic

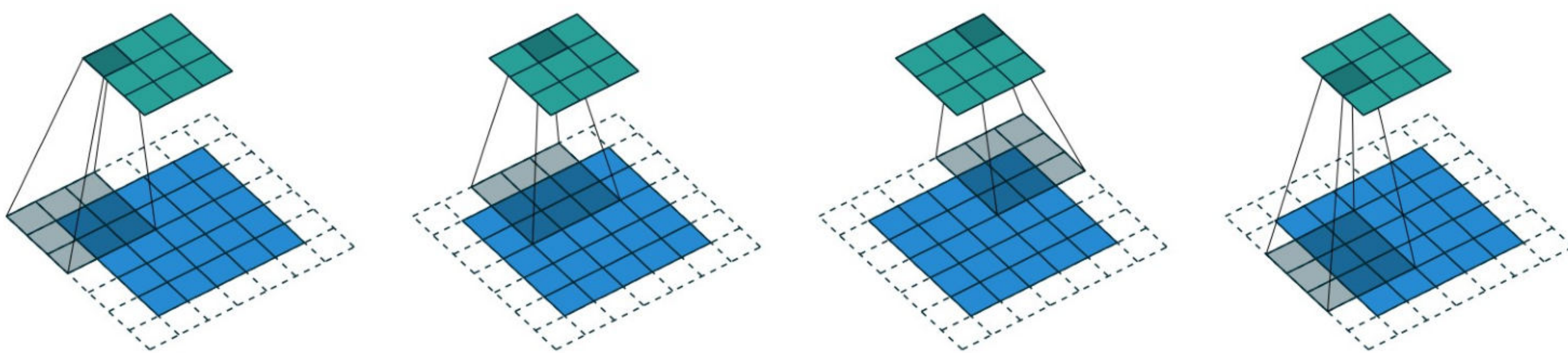
Introduction to image processing

- Strided convolution / correlation

  Skip s positions in the input signal when convolving the kernel.

  Half padding strided convolution

$$o(r,t) = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} h(i,j) e_e[sr + i - (m-1)/2, st + j - (n-1)/2]$$



https://github.com/vdumoulin/conv_arithmetic

Introduction to image processing

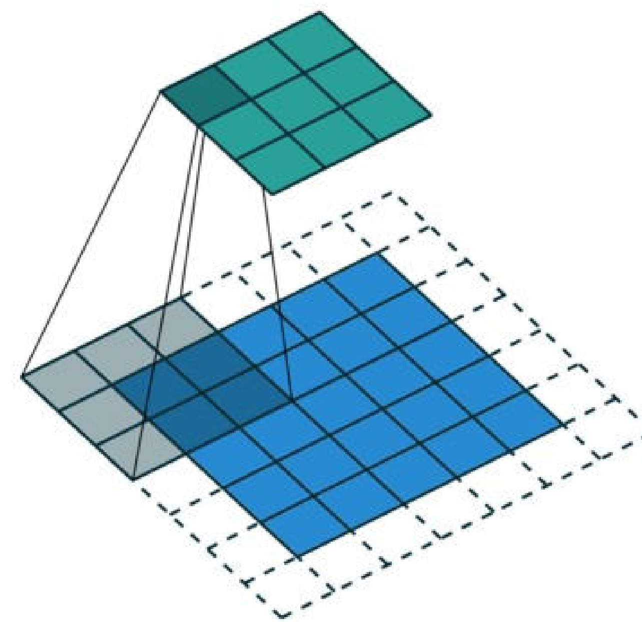
- Strided convolution / correlation

Skip s positions in the input signal when convolving the kernel.

Half padding strided convolution

$$o(r,t) = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} h(i,j) e_e[sr + i - (m-1)/2, st + j - (n-1)/2]$$

$$\frac{M + 2p - m}{s} + 1$$



If half padded and $s = 2$ the ouput is half the input

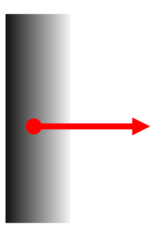
$$\frac{M-1}{2} + 1 \times \frac{N-1}{2} + 1$$

https://github.com/vdumoulin/conv_arithmetic

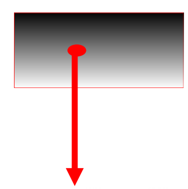
Introduction to image processing

- Gradient filter

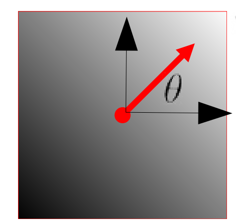
The image gradient at pixel $\mathtt{I}(x, y)$ is a vector $[\mathtt{I}_x, \mathtt{I}_y]$ pointing in the direction of maximum growth in the image gray value



$$\nabla \mathbf{I} = \left[\frac{\partial \mathtt{I}}{\partial x}, 0\right]$$



$$\nabla \mathbf{I} = \left[0, \frac{\partial \mathtt{I}}{\partial y}\right]$$



$$\nabla \mathbf{I} = \left[\frac{\partial \mathtt{I}}{\partial x}, \frac{\partial \mathtt{I}}{\partial y}\right]$$

$$|\bar{\nabla} \mathtt{I}| = \sqrt{\mathtt{I}_x^2 + \mathtt{I}_y^2}$$

$$\theta = \arctan\left(\frac{\mathtt{I}_y}{\mathtt{I}_x}\right)$$

Introduction to image processing

- Gradient filter

How do I compute the gradient of an image?

$$\frac{\partial \mathtt{I}(x,y)}{\partial x} \equiv \mathtt{I}_x \approx \frac{\mathtt{I}(x+\delta x, y) - \mathtt{I}(x-\delta x, y)}{2\delta x}$$

$$\mathtt{I}_x \approx \frac{1}{2}\left[\mathtt{I}(x+1,y) - \mathtt{I}(x-1,y)\right] = h_{dx} * \mathtt{I}$$

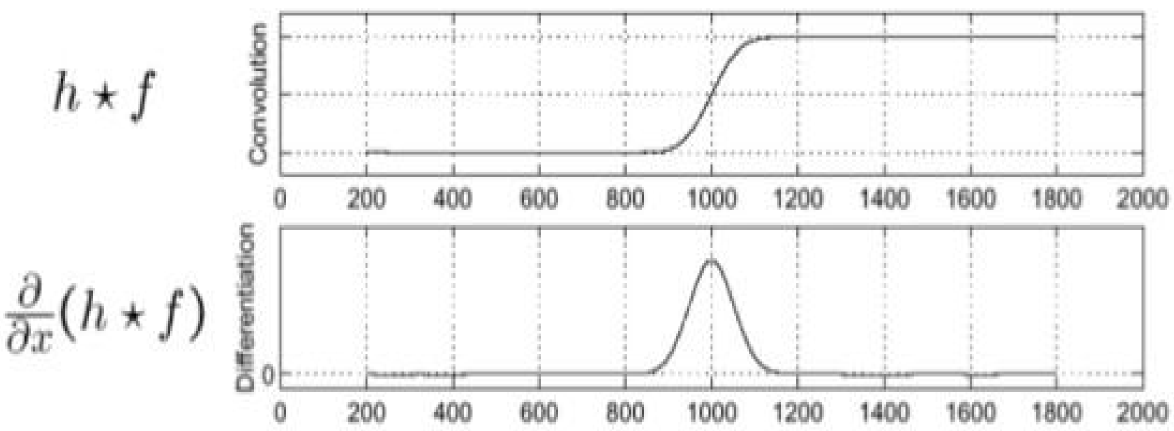
$$h_{dx} = \frac{1}{2}\left[\begin{array}{ccc} 1 & 0 & -1 \end{array}\right] \; h_{dy} = \frac{1}{2}\left[\begin{array}{c} 1 \\ 0 \\ -1 \end{array}\right]$$

Then $\mathtt{I}_x = h_{dx} * \mathtt{I}$ and $\mathtt{I}_y = h_{dy} * \mathtt{I}$

Introduction to image processing

- Gradient filter

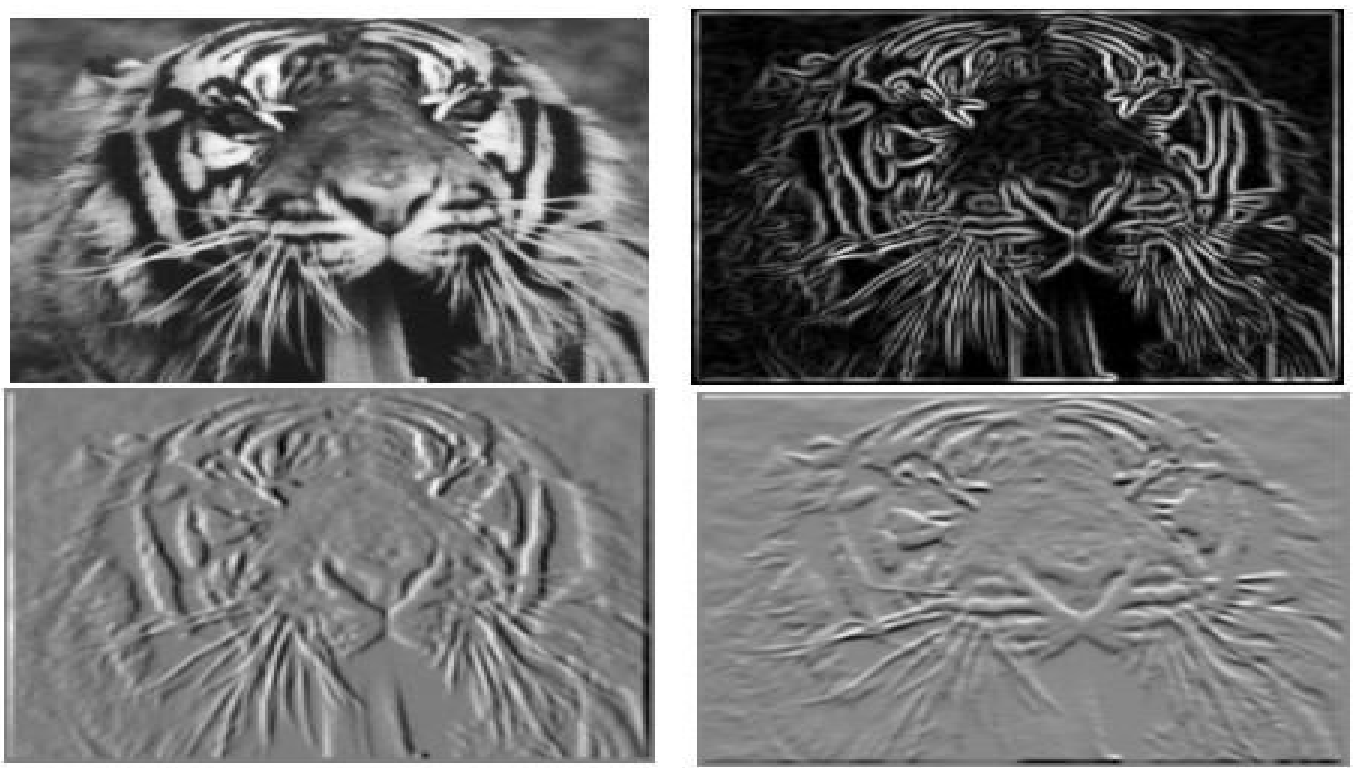
The gradient of an image border



Introduction to image processing

- Gradient filter

Results of gradient estimation



$$||\nabla \mathtt{I}(\mathbf{u})||$$

$$\frac{\partial \mathtt{I}(\mathbf{u})}{\partial x} \qquad \frac{\partial \mathtt{I}(\mathbf{u})}{\partial y}$$

Introduction to image processing

- Laplacian filter

The Laplacian of digital image $\mathtt{I}(x, y)$ is given by the scalar

$$\Delta \mathtt{I} = \frac{\partial^2 \mathtt{I}}{\partial x^2} + \frac{\partial^2 \mathtt{I}}{\partial y^2}$$

How do we compute it?

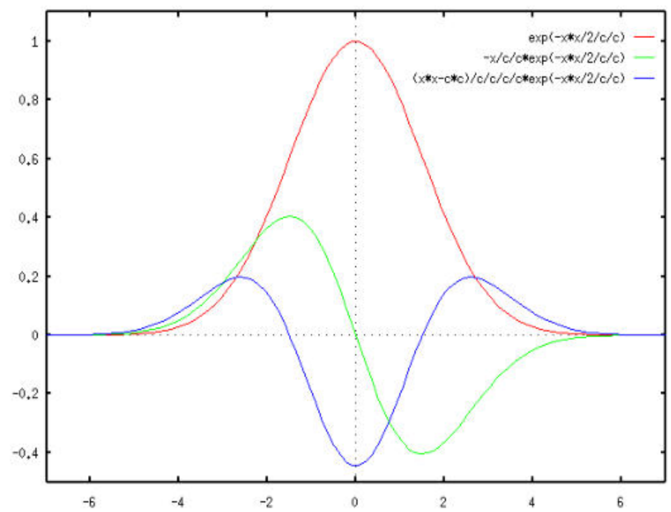
$$\mathtt{I}_x \approx \mathtt{I}(x+1, y) - \mathtt{I}(x, y) = [1 \;\; -1] * \mathtt{I}$$

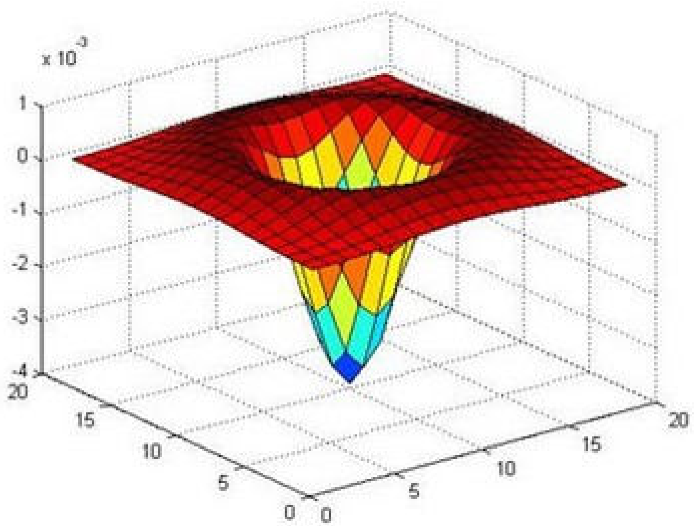
$$\mathtt{I}_{xx} \approx [1 \;\; -1] * [1 \;\; -1] * \mathtt{I} = [1 \;\; -2 \;\; 1] * \mathtt{I}$$

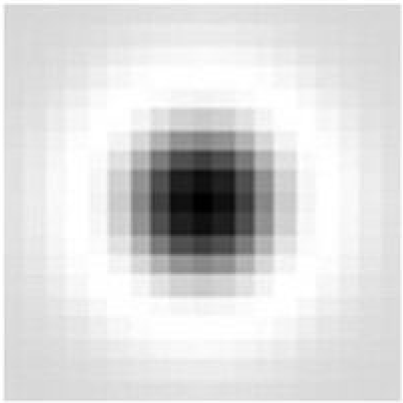
$$\Delta \mathtt{I} = \left(\begin{bmatrix} 1 & -2 & 1 \end{bmatrix} + \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} \right) * \mathtt{I} = \mathrm{h}_{l_a} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Introduction to image processing

- Laplacian filter. The Laplacian of Gaussian filter:



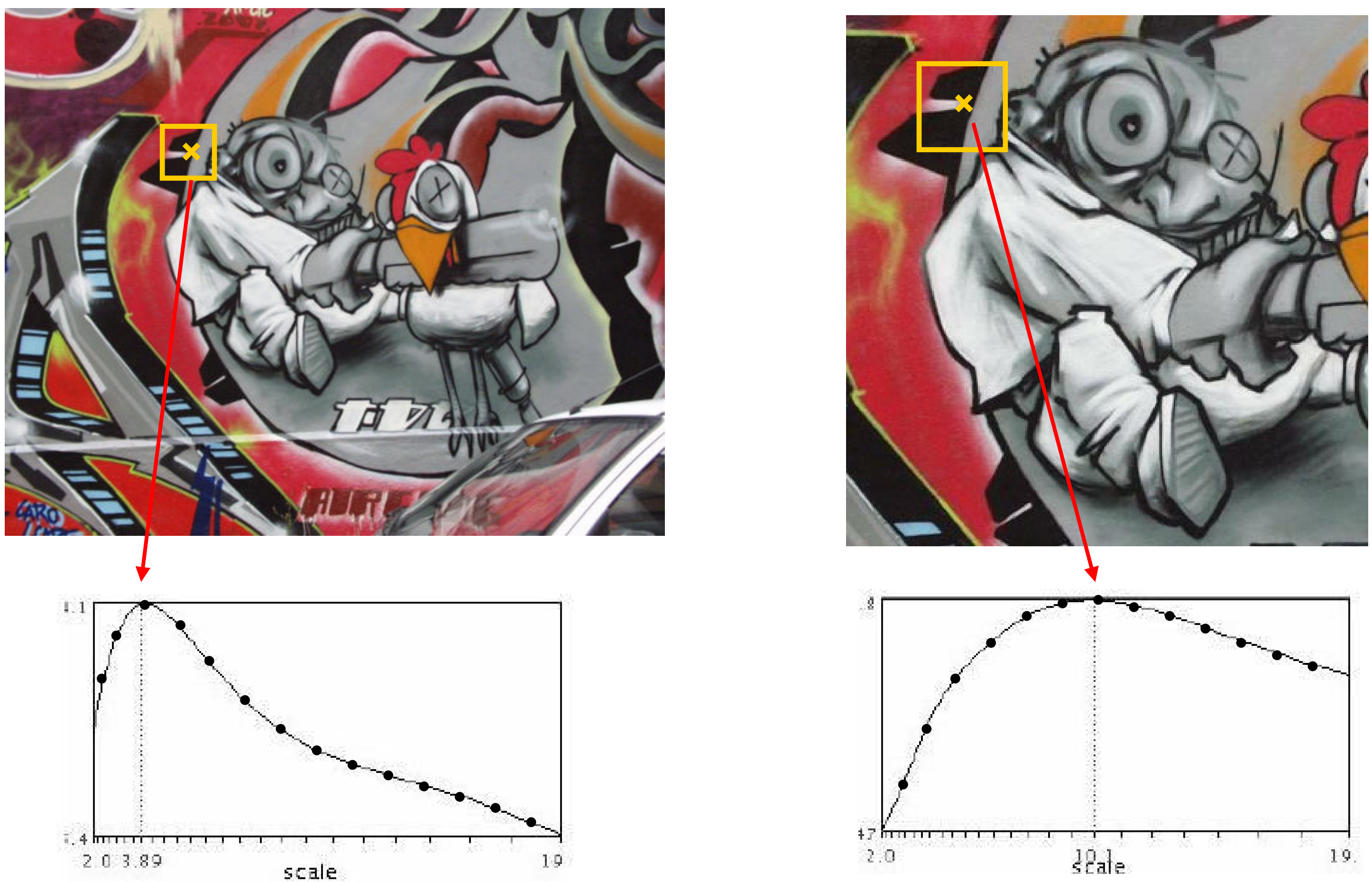




Introduction to image processing

- Laplacian filter. The Laplacian of Gaussian filter.

Response at different scales

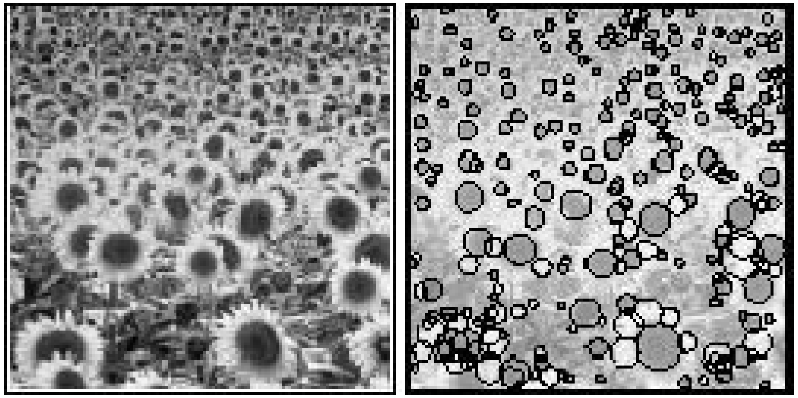


$f(I_{i_1 \dots i_m}(x, \sigma))$ $f(I_{i_1 \dots i_m}(x', \sigma'))$

Introduction to image processing

- Why are we interested in gradients and laplacians?
  - Gradients are useful for image description
    Image edges are invariant to illumination changes

  - Laplacians are useful for detecting image structures (blobs)
    - ➔ Local maxima mark blob centers
    - ➔ Variance represent blob size



Object recognition

- Problem statement

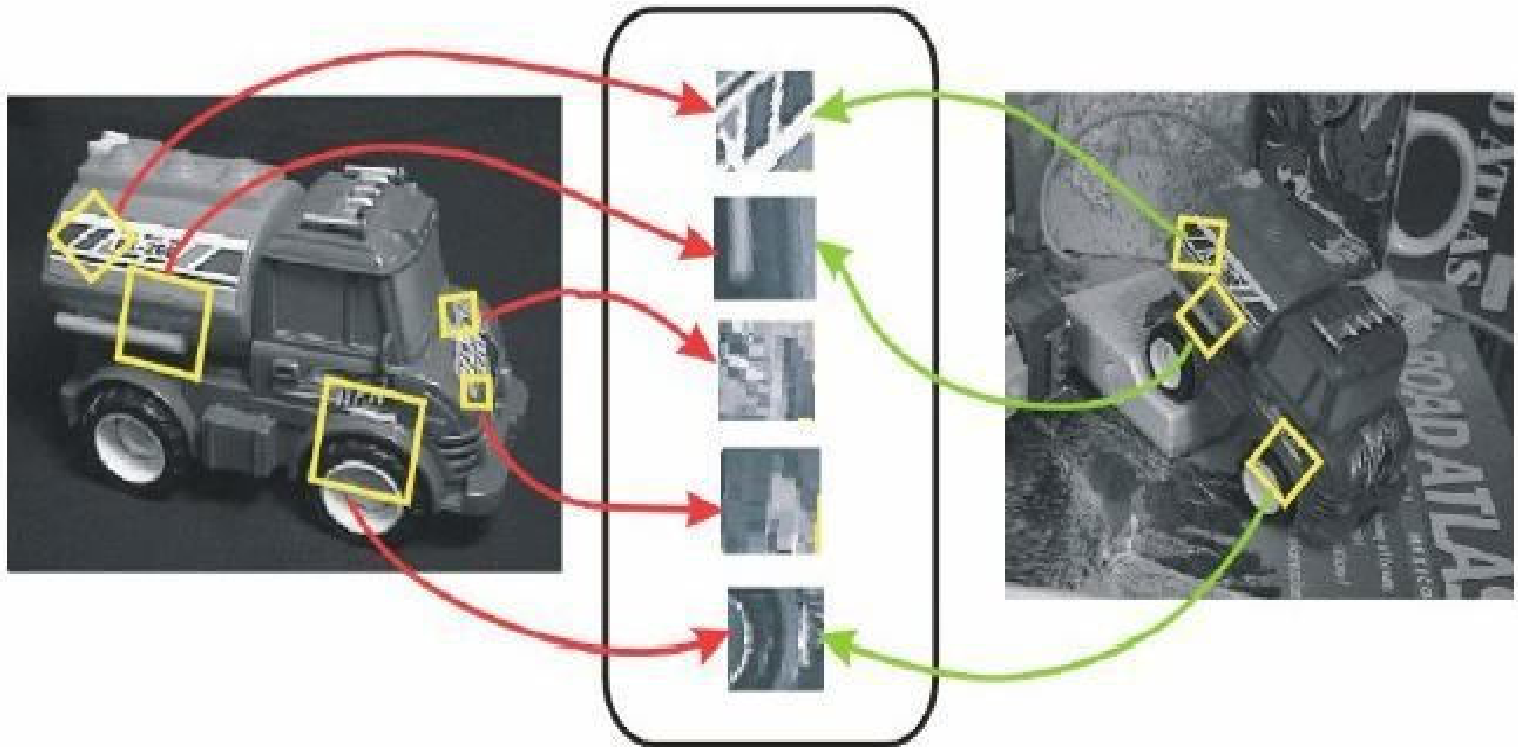
We want to recognize objects inspite of the large variability of object clases, changes in appearance caused by illumination, geometry, deformation, etc.

An appropriate image description invariant to most of those variations will be the key to success.

How to describe?

- General approach

Image content represented by local models of appearance



D. Lowe. Object Recognition from Local Scale-Invariant Features. ICCV 1999

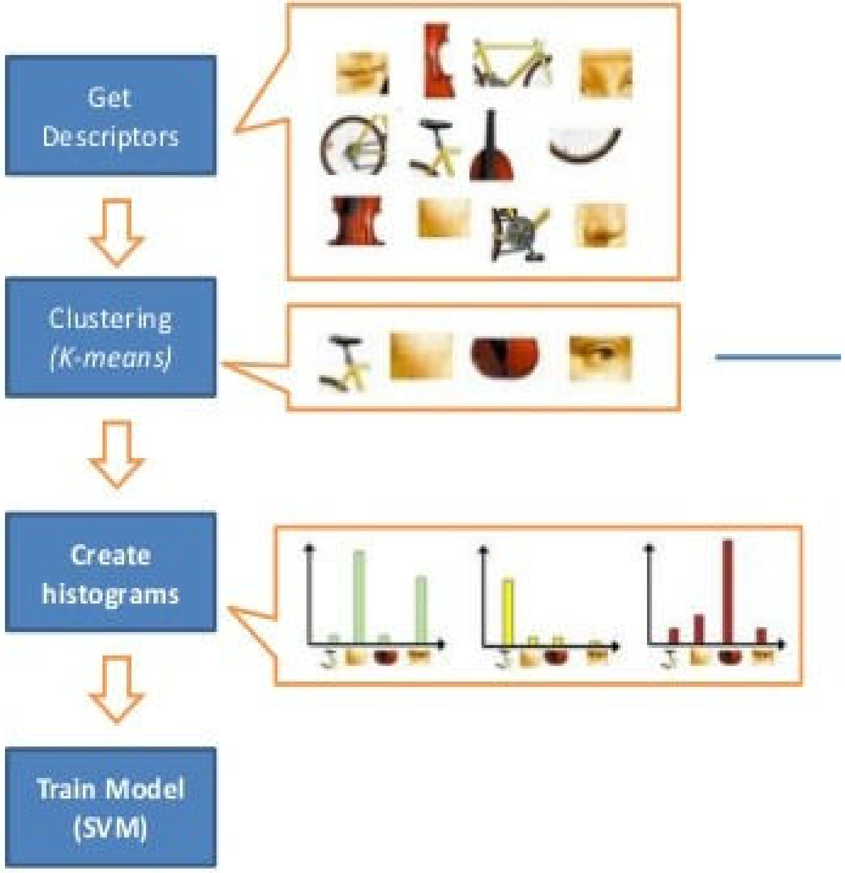
Shallow object recognition approach

1. Low-level features

2. Mid-level representation

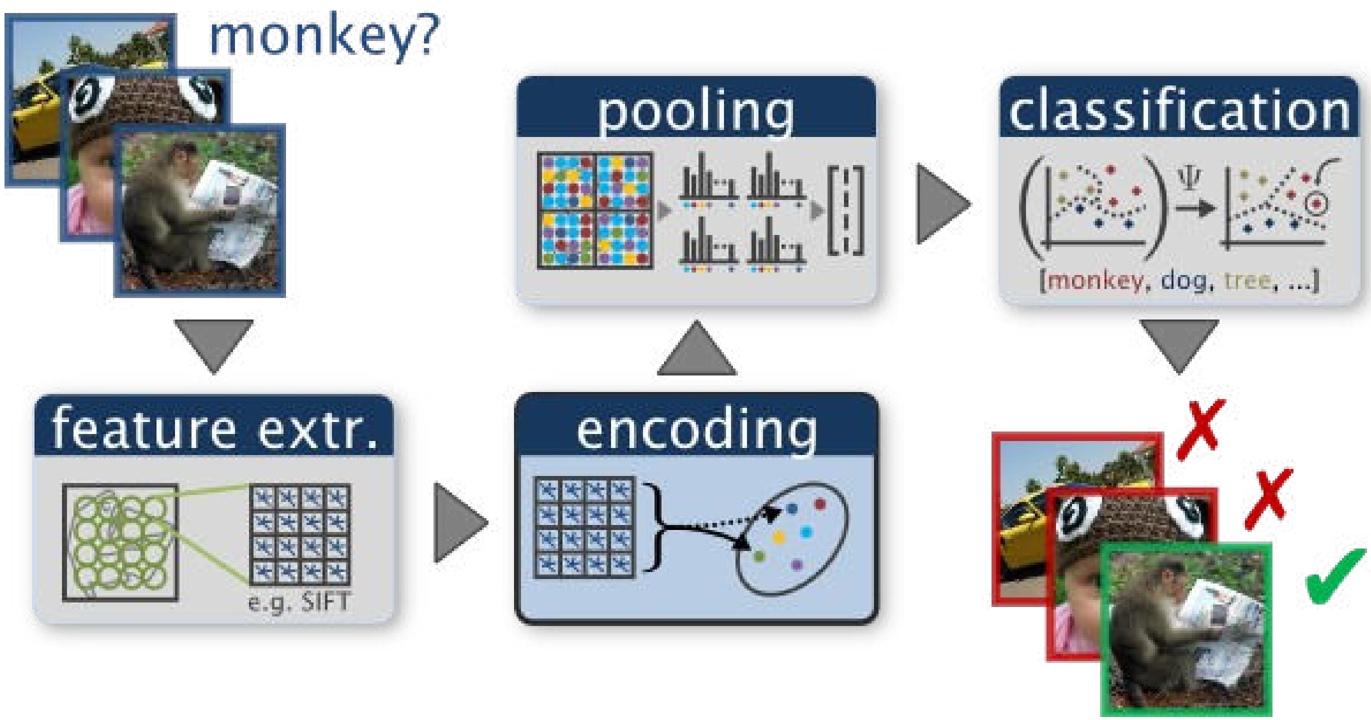
3. Pooling/aggregation

4. Classification



Shallow object recognition approach

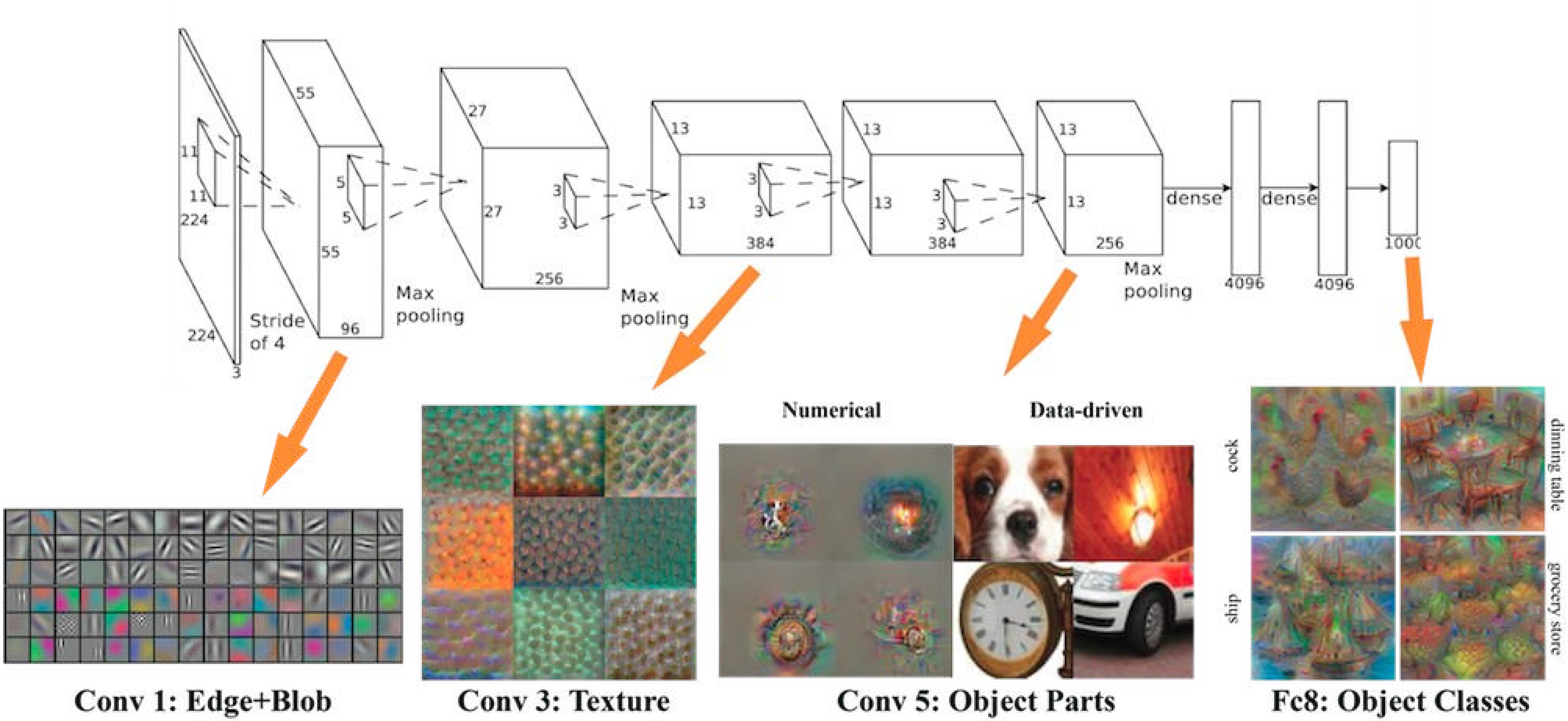
1. Extract low-level features
2. Compute mid-level representation (quantification)
3. Pool/aggregate spatial information
4. Classify



Deep object recognition approach

Trainable hierarchical representations (Krizhevsky, 2012)

Image content represented by the agreggation of local features into a hierarchy of representations AUTOMATICALLY trained.



http://vision03.csail.mit.edu/cnn_art/index.html

Object recognition challenges

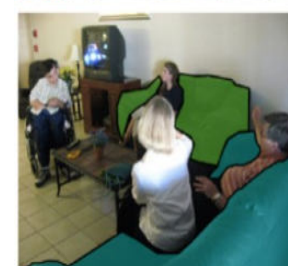
- Early challenges

  Caltech 101 (2004), Caltech 256 (2007), Pascal VOC (2006-2012) , .

- Image Large Scale Visual Recognition Challenge
  - 1000 categories
  - 1.4 M images
  - ~ 3 instances per image

IMAGENET
- 15.000 visual categories
- 10 M labeled images
- ~ 700 images/category

- Common Objects in Context (Microsoft)
  - 91 categories, 328 k images
  - 2.5 M instances (~ 7.7 per image)
  - Every instance fully segmented



Deep object recognition approach

Image Large Scale Visual Recognition Contest (ILSVRC)