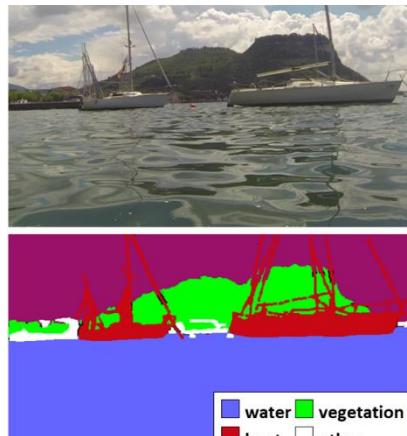
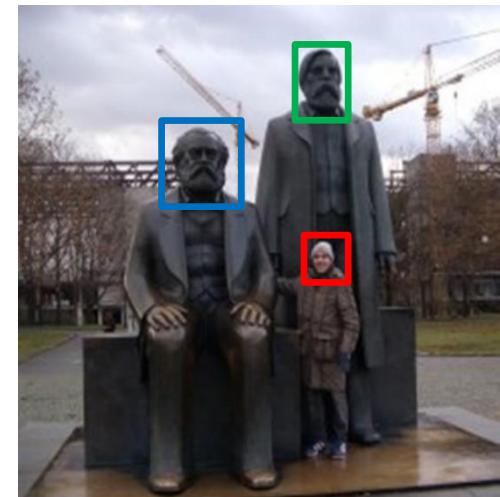
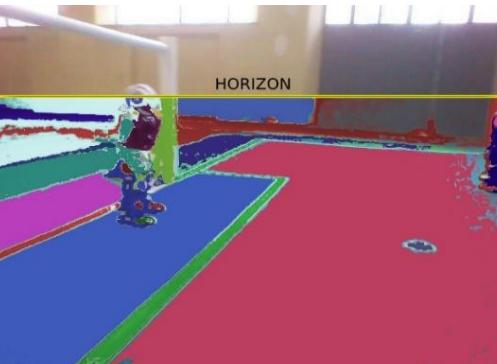




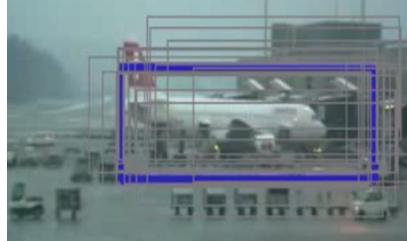
UNIVERSITÀ DEGLI STUDI  
DELLA BASILICATA

*Corso di Visione e Percezione*

# Percezione Visione



Docente  
Domenico D. Bloisi



# Domenico Daniele Bloisi

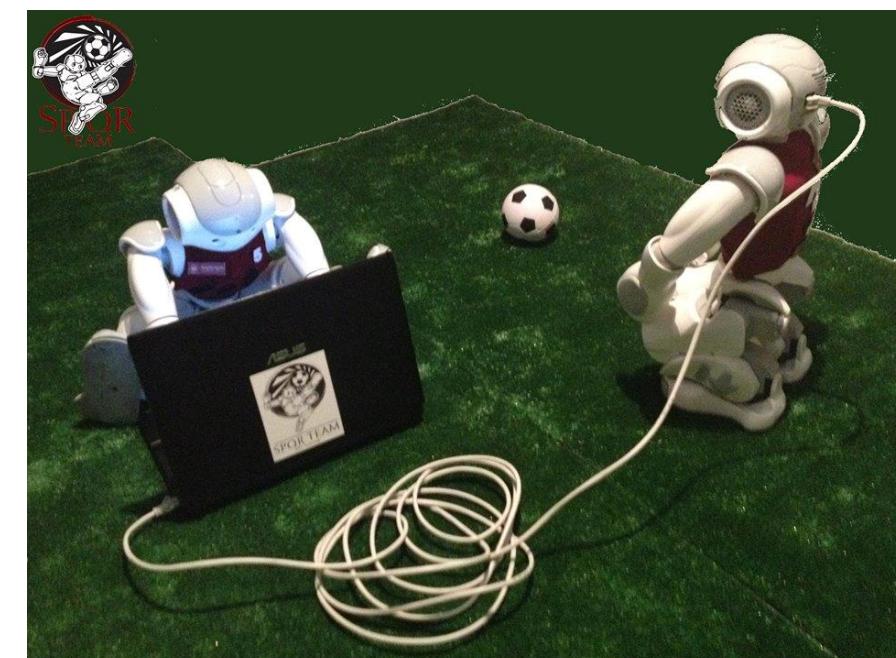
---

- Professore Associato  
Dipartimento di Matematica, Informatica  
ed Economia  
Università degli studi della Basilicata

<http://web.unibas.it/bloisi>

- SPQR Robot Soccer Team  
Dipartimento di Informatica, Automatica  
e Gestionale Università degli studi di  
Roma “La Sapienza”

<http://spqr.diag.uniroma1.it>



# UNIBAS Wolves <https://sites.google.com/unibas.it/wolves>

---



- UNIBAS WOLVES is the robot soccer team of the University of Basilicata. Established in 2019, it is focussed on developing software for NAO soccer robots participating in RoboCup competitions.
- UNIBAS WOLVES team is twinned with [SPQR Team](#) at Sapienza University of Rome.



# Informazioni sul corso

---

- Home page del corso:  
<https://web.unibas.it/bloisi/corsi/visione-e-percezione.html>
- Docente: Domenico Daniele Bloisi
- Periodo: Il semestre marzo 2022 – giugno 2022
  - Martedì dalle 15:00 alle 17:00 (Aula Copernico)
  - Mercoledì dalle 8:30 alle 10:30 (Aula Copernico)

# Ricevimento

---

- Durante il periodo delle lezioni:

Mercoledì dalle 11:00 alle 12:30 → Edificio 3D, II piano, stanza 15

Si invitano gli studenti a controllare regolarmente la bacheca degli avvisi per eventuali variazioni

- Al di fuori del periodo delle lezioni:

da concordare con il docente tramite email

Per prenotare un appuntamento inviare  
una email a

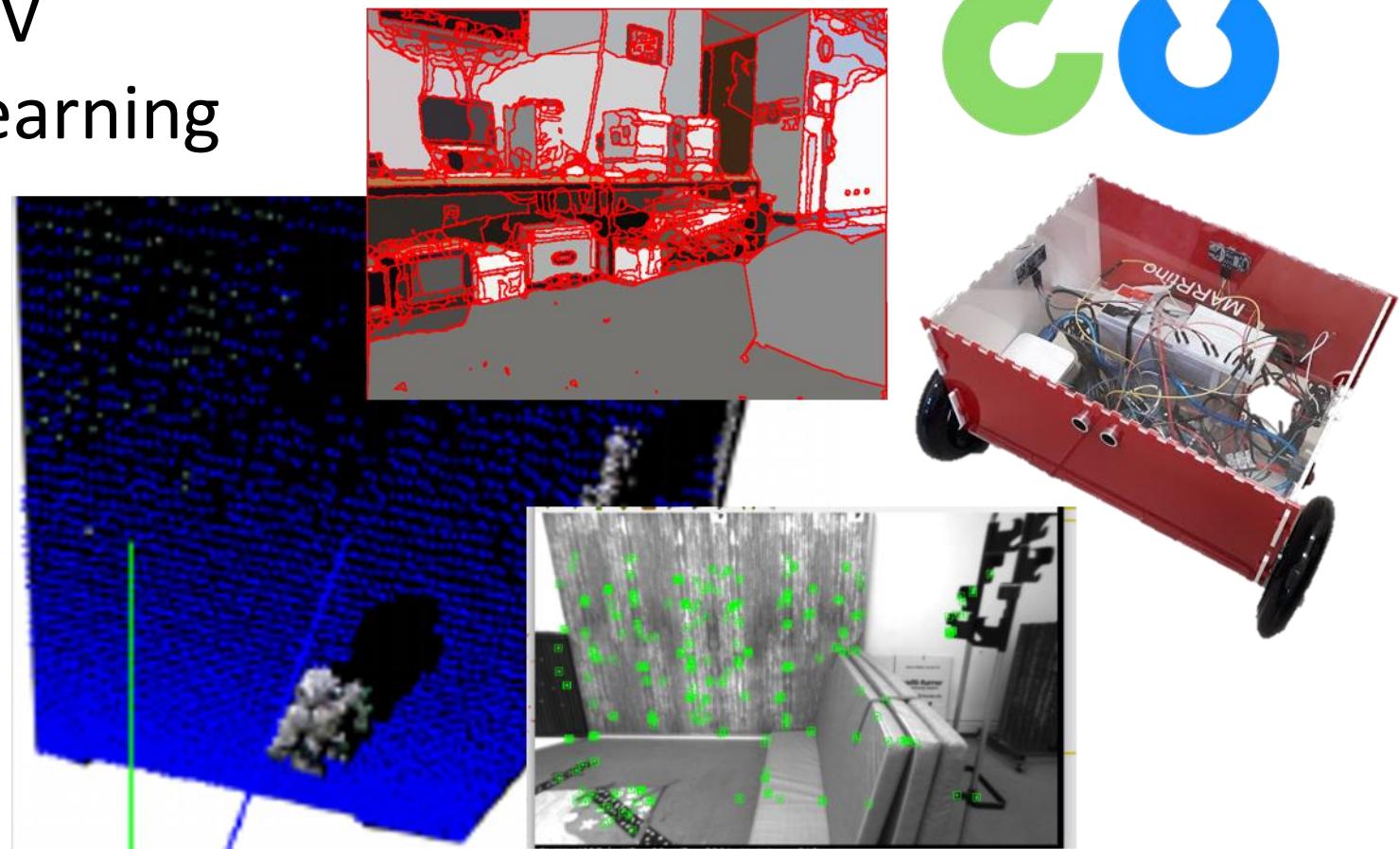
[domenico.bloisi@unibas.it](mailto:domenico.bloisi@unibas.it)



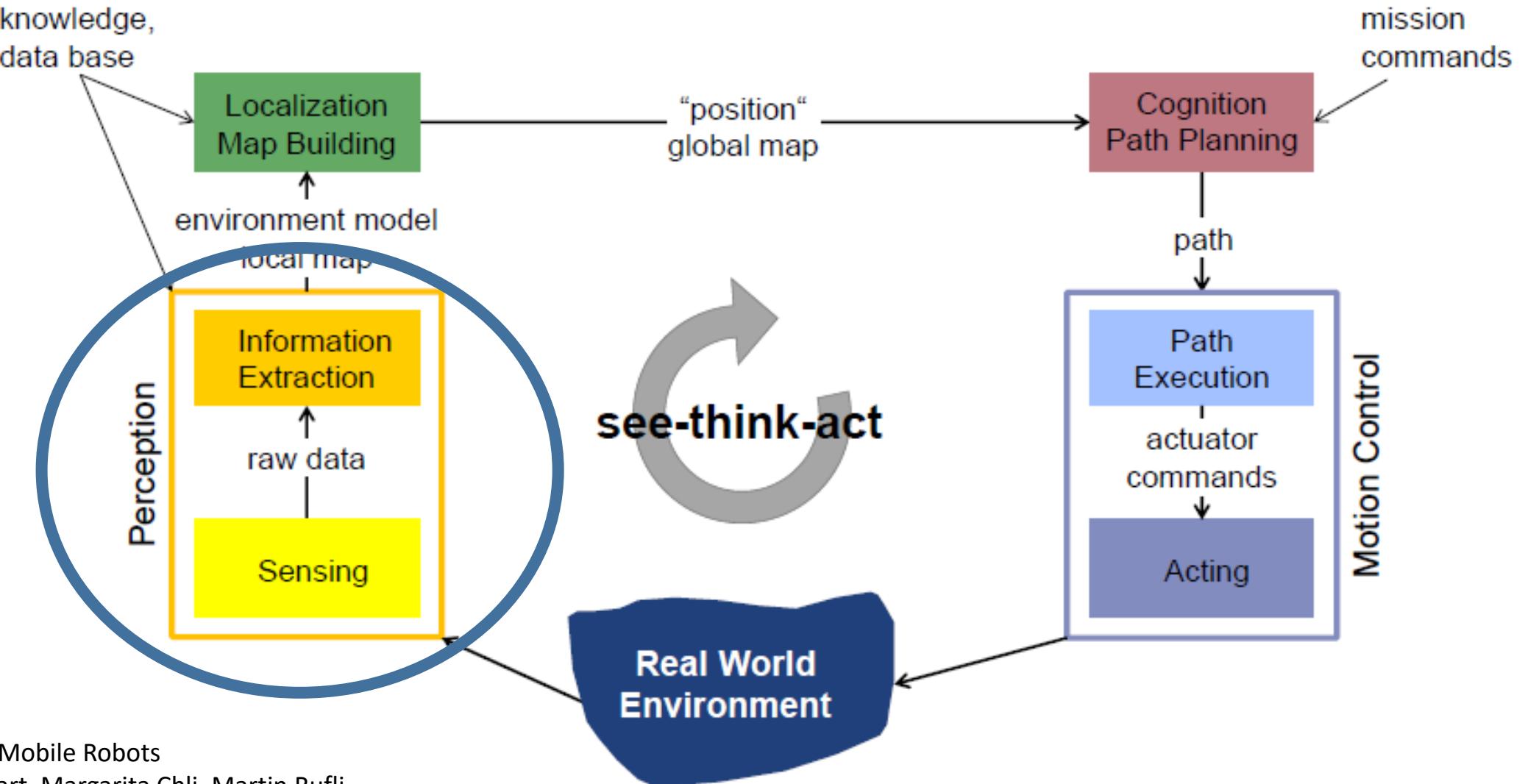
# Programma – Visione e Percezione

---

- Introduzione al linguaggio Python
- [Elaborazione delle immagini con Python](#)
- Percezione 2D – OpenCV
- Introduzione al Deep Learning
- ROS
- Il paradigma publisher and subscriber
- Simulatori
- Percezione 3D - PCL



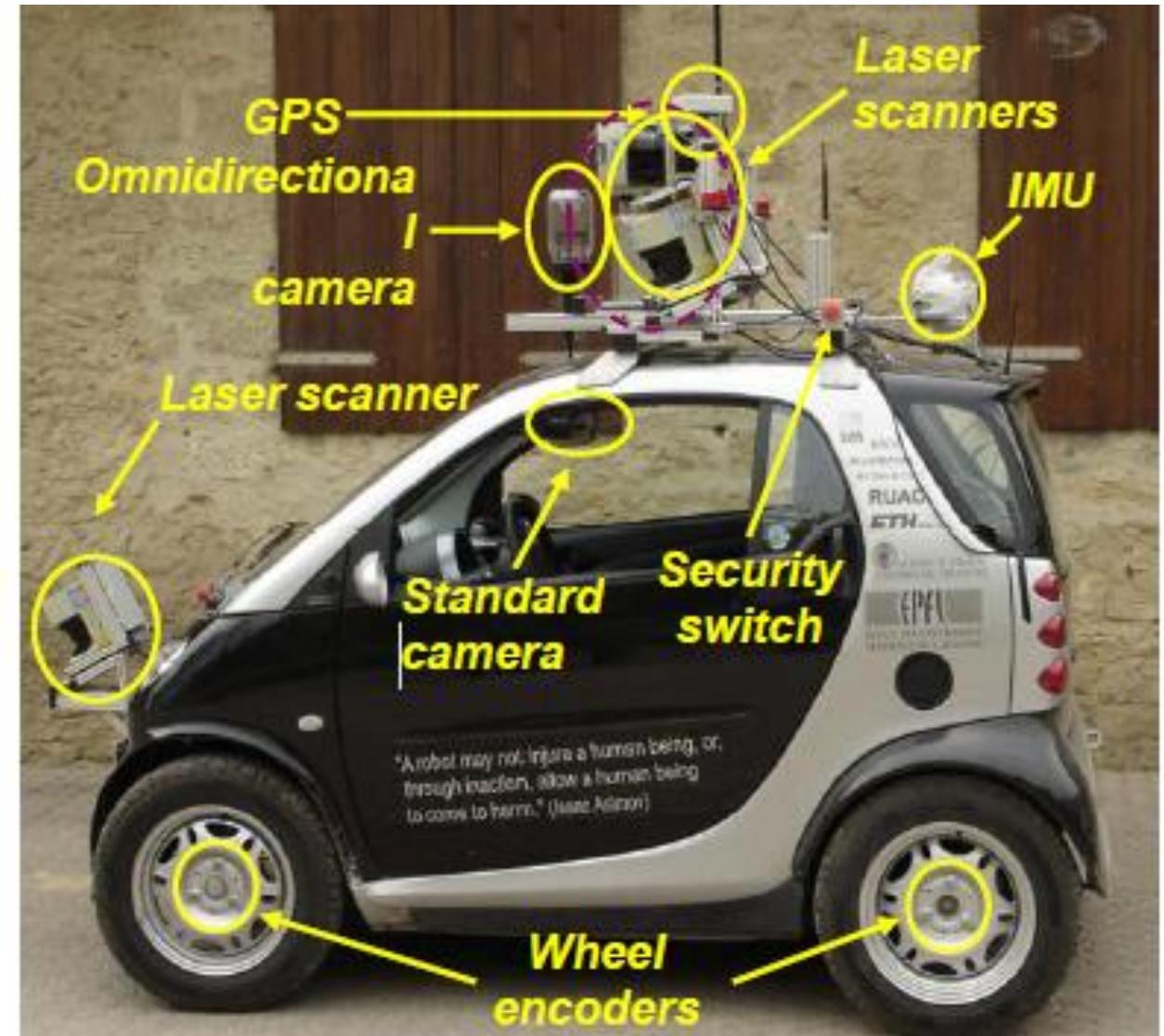
# See-Think-Act Cycle



# Percezione

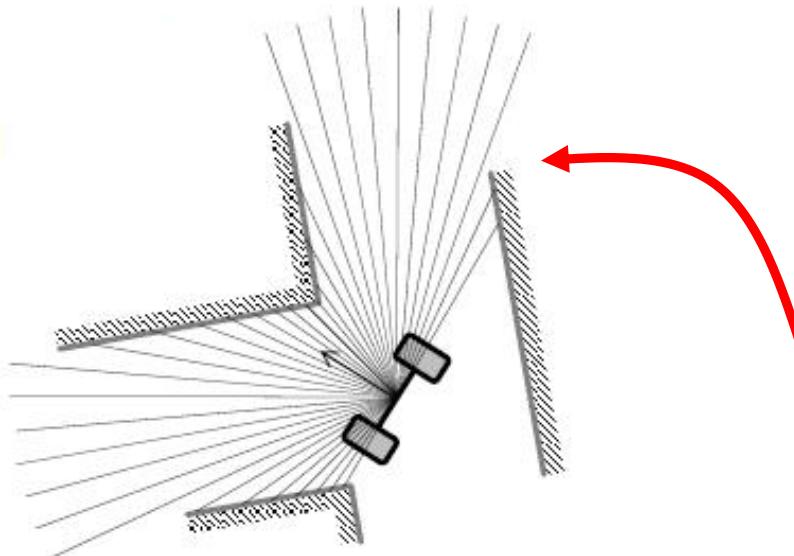


UC3M

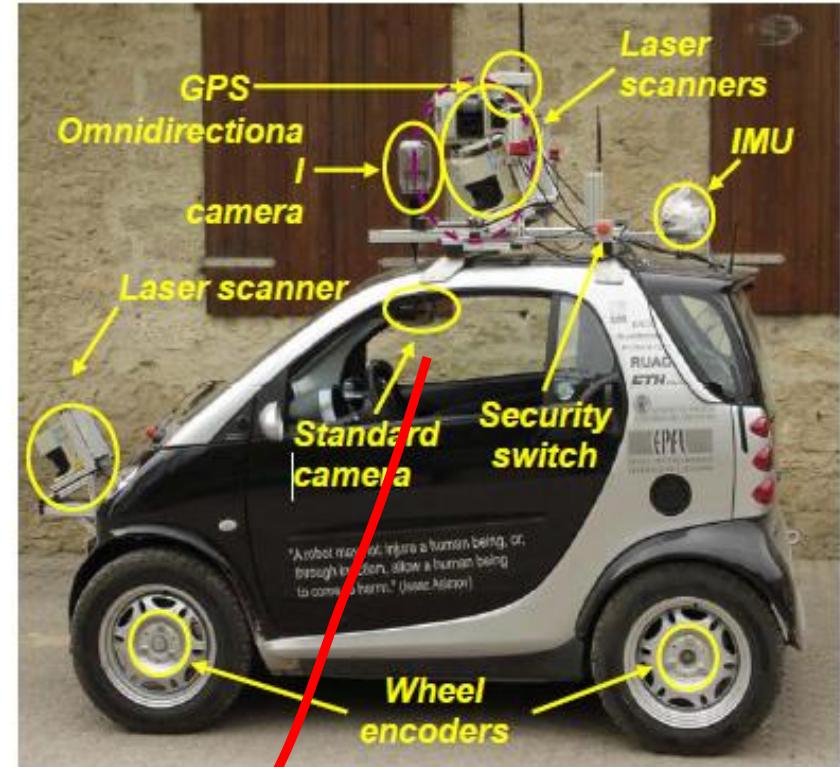
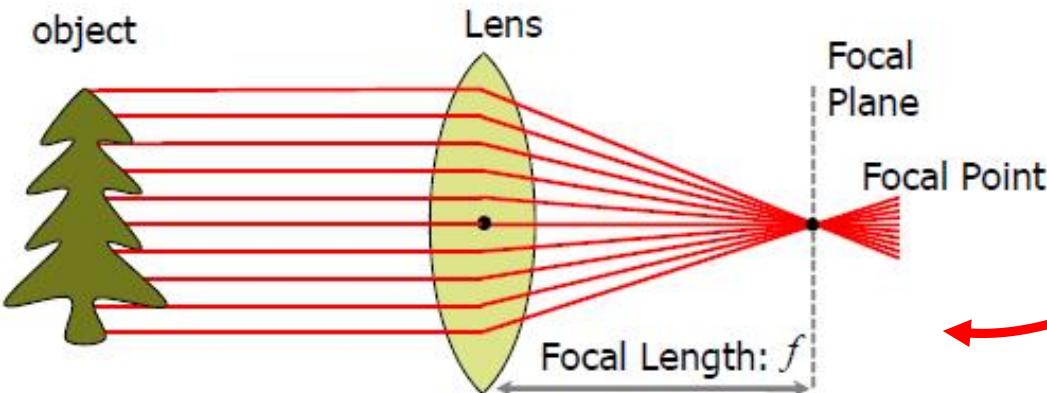


# Sensori per la percezione

- Laser scanner
  - time of flight

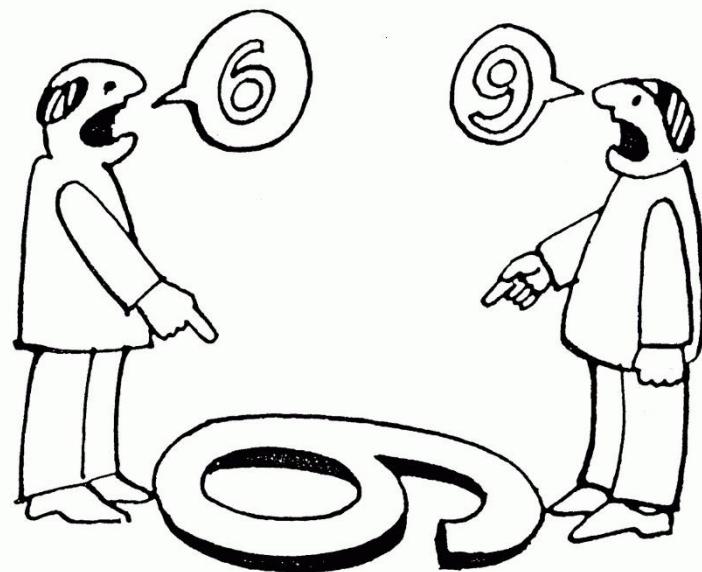


- Camera

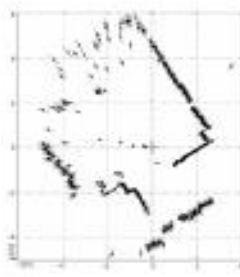
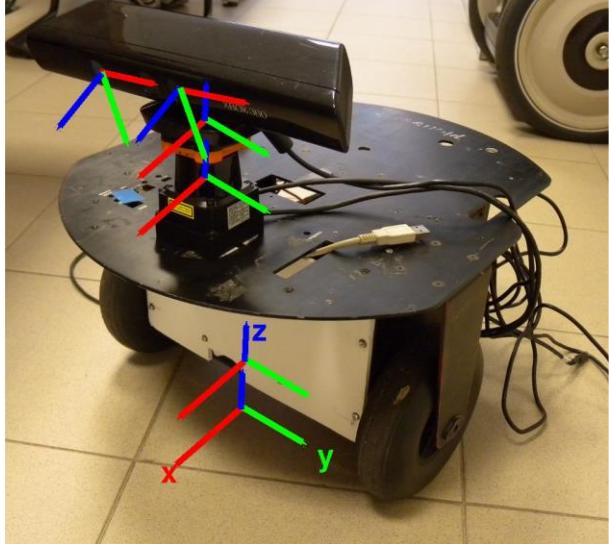


# Ambiguità nella percezione

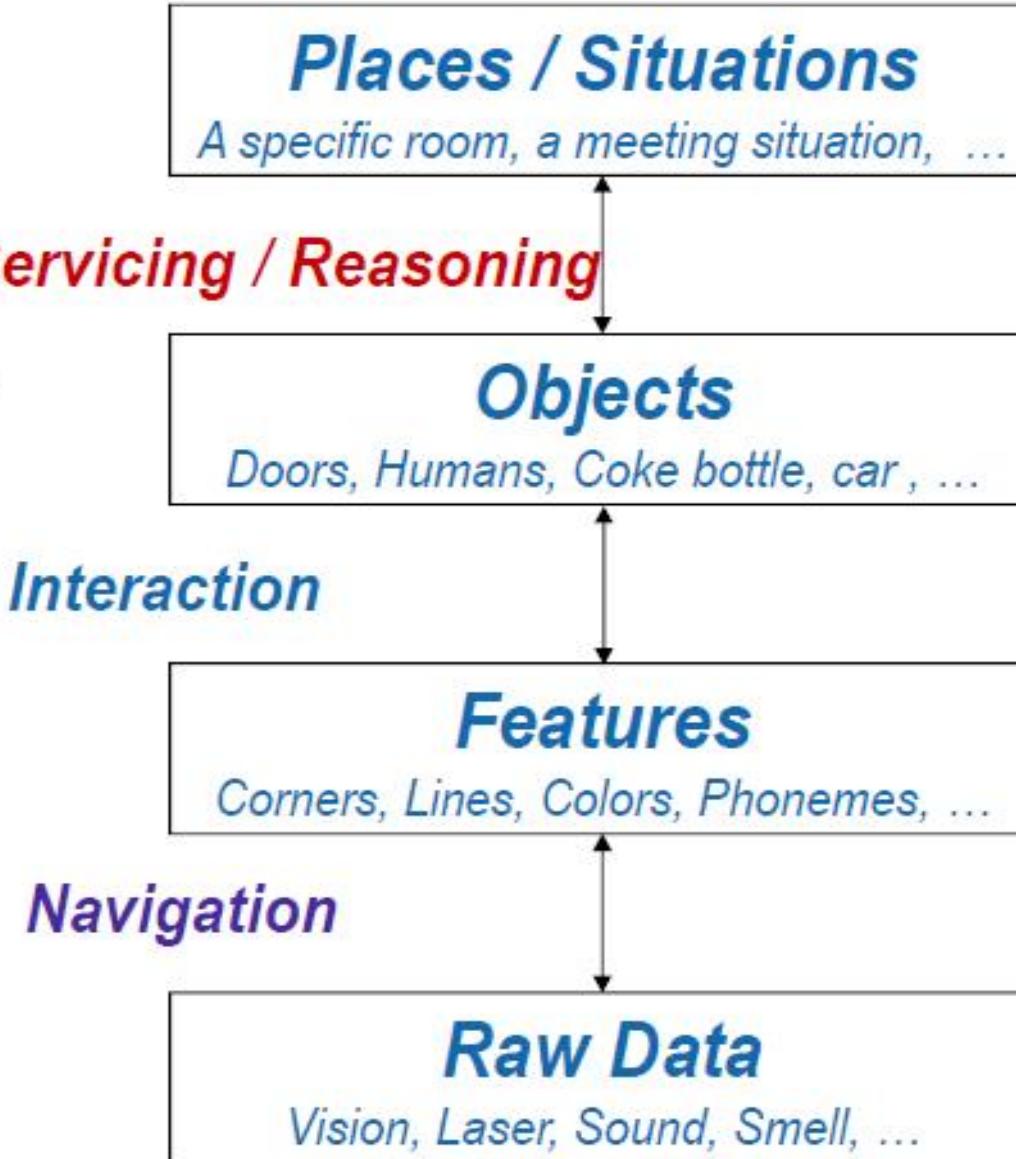
---



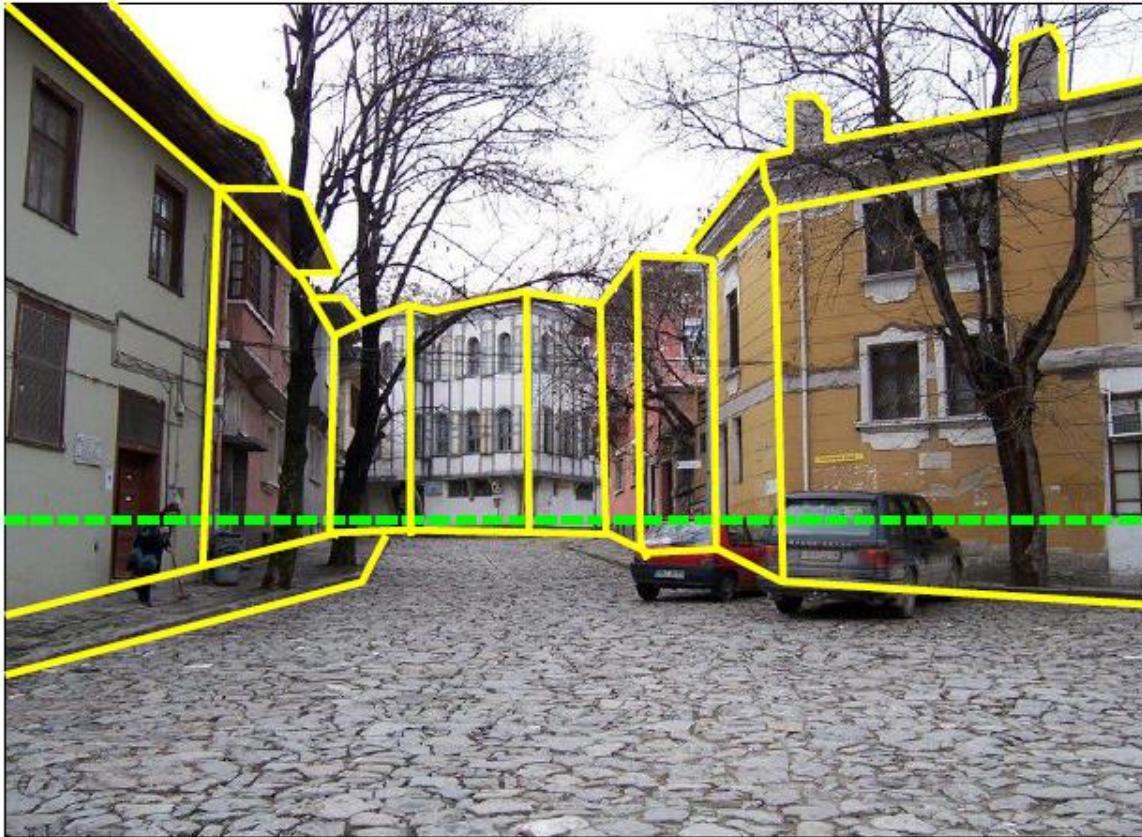
# Percezione per robot mobili



Compressing Information  
↑



# Informazioni geometriche e semantiche



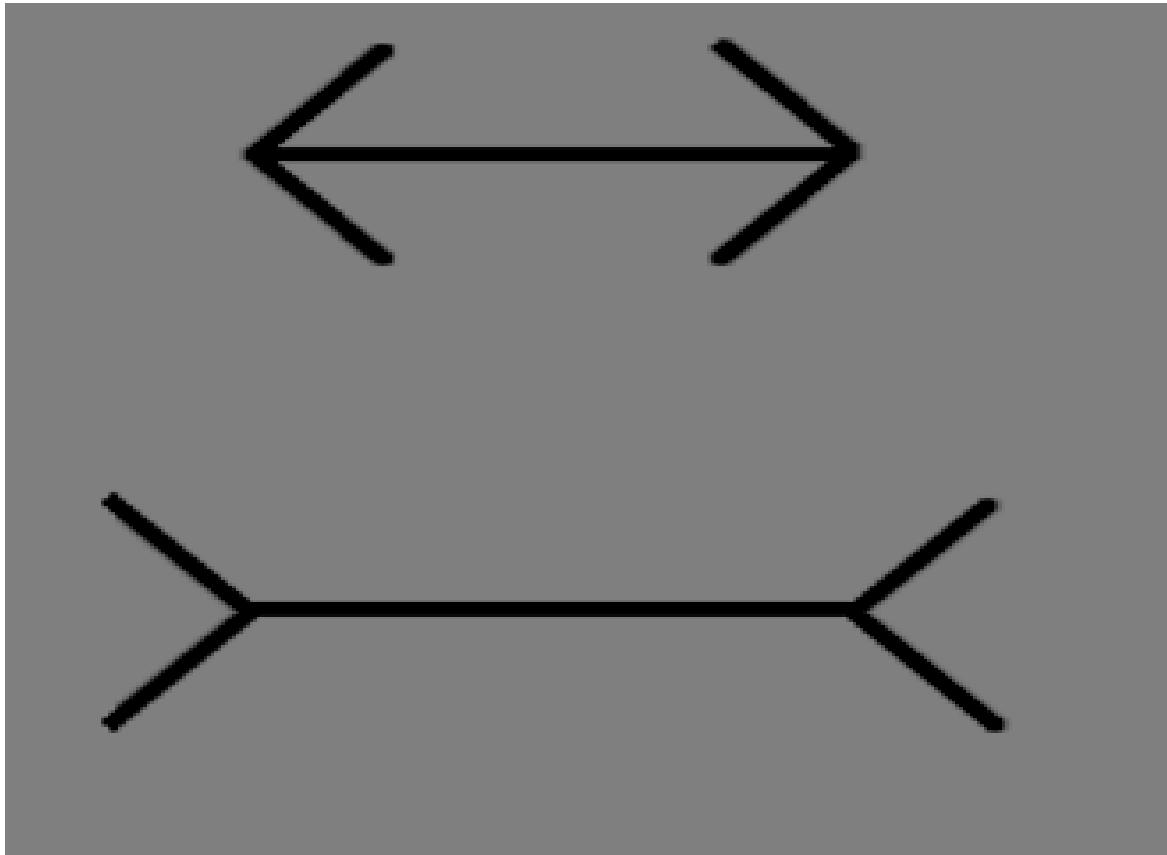
informazioni geometriche



informazioni semantiche

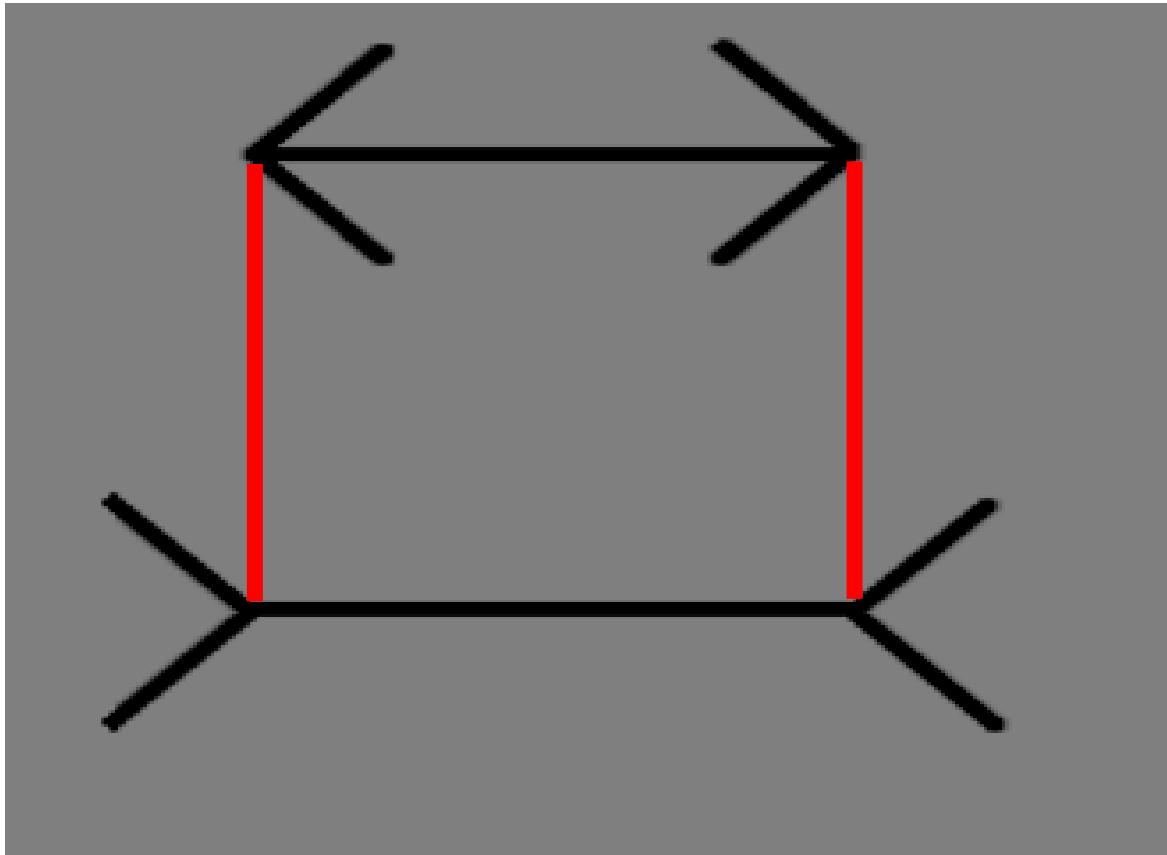
# Percezione visuale

---



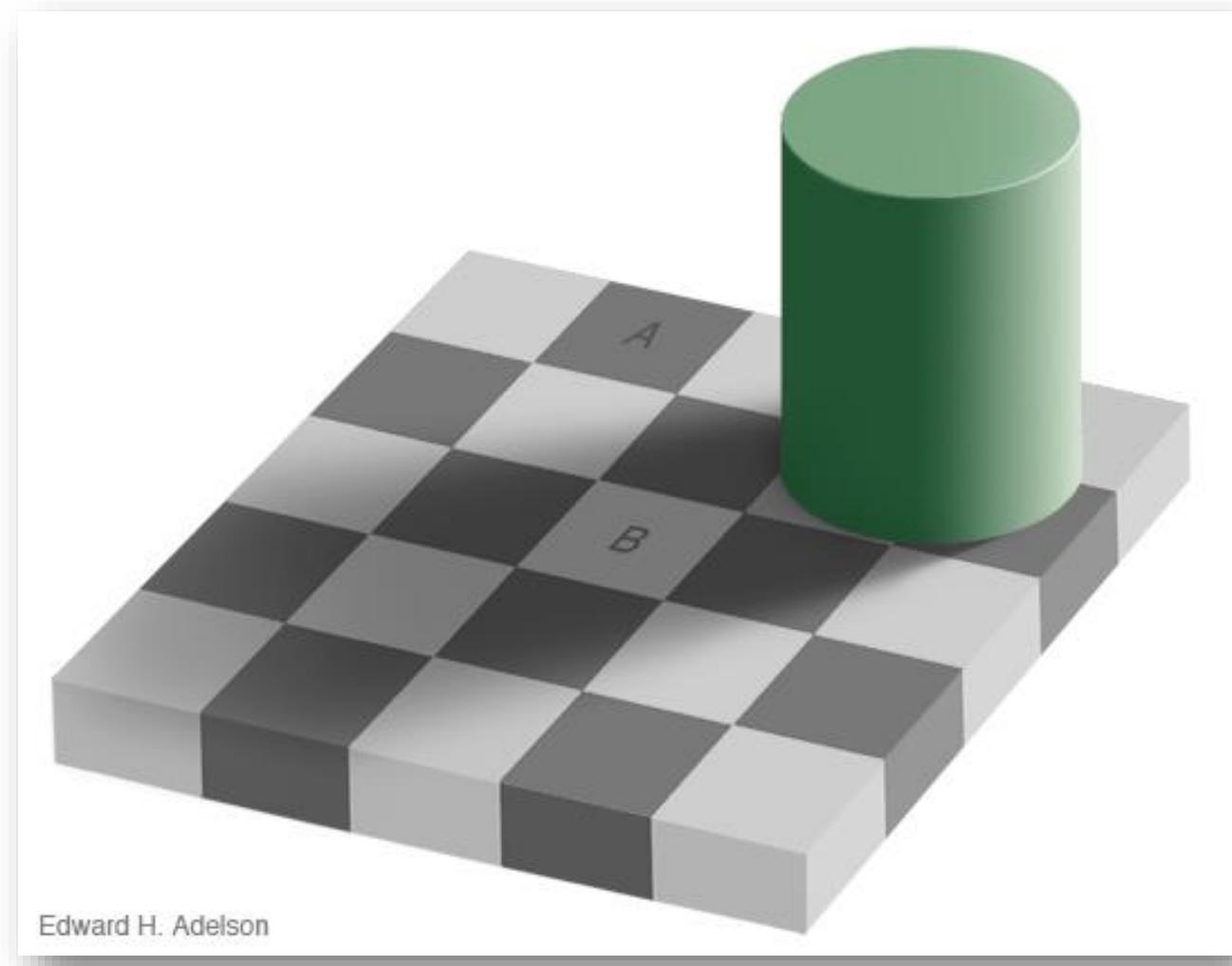
# Percezione visuale

---



# Illusioni ottiche

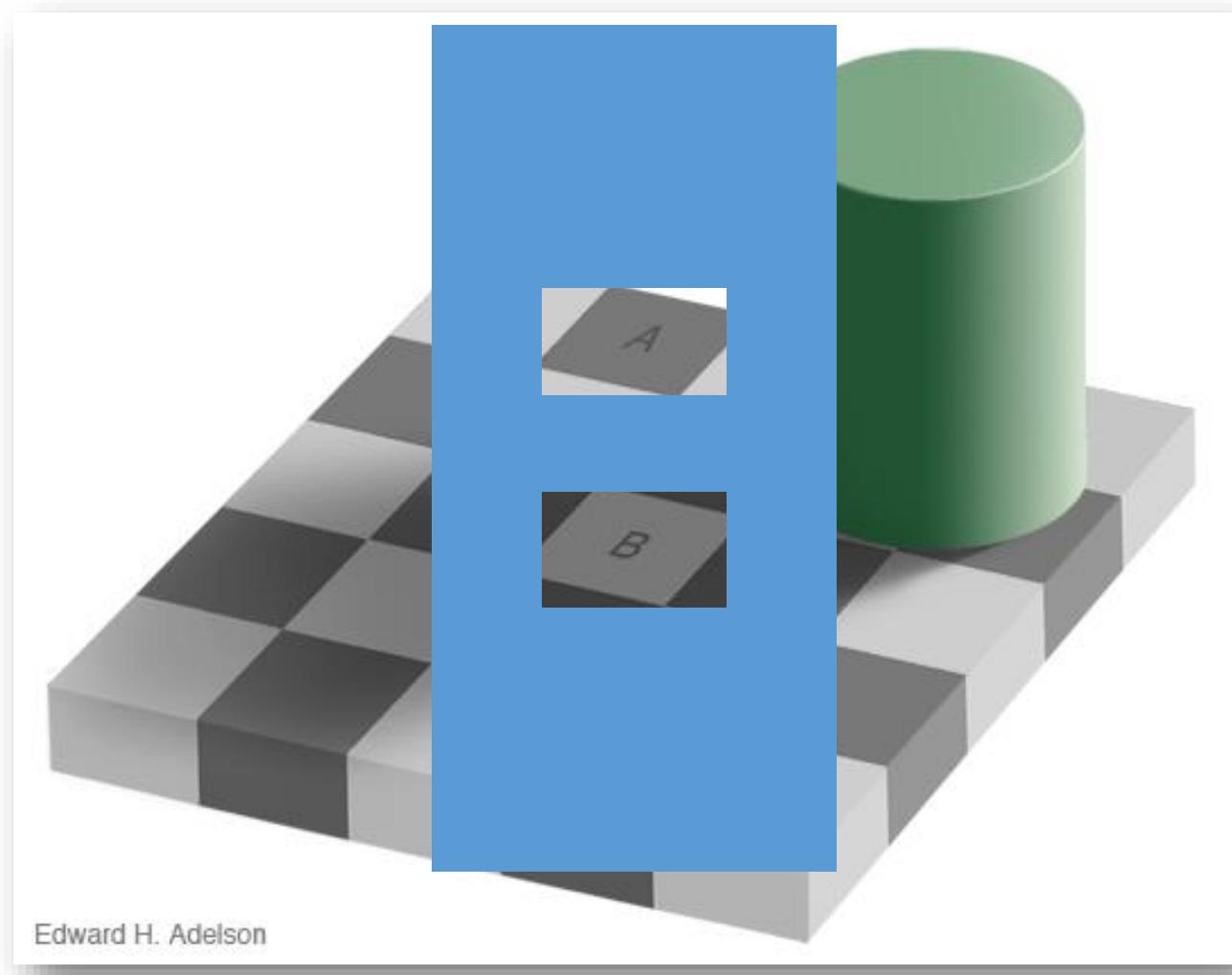
---



Edward H. Adelson

# Illusioni ottiche

---

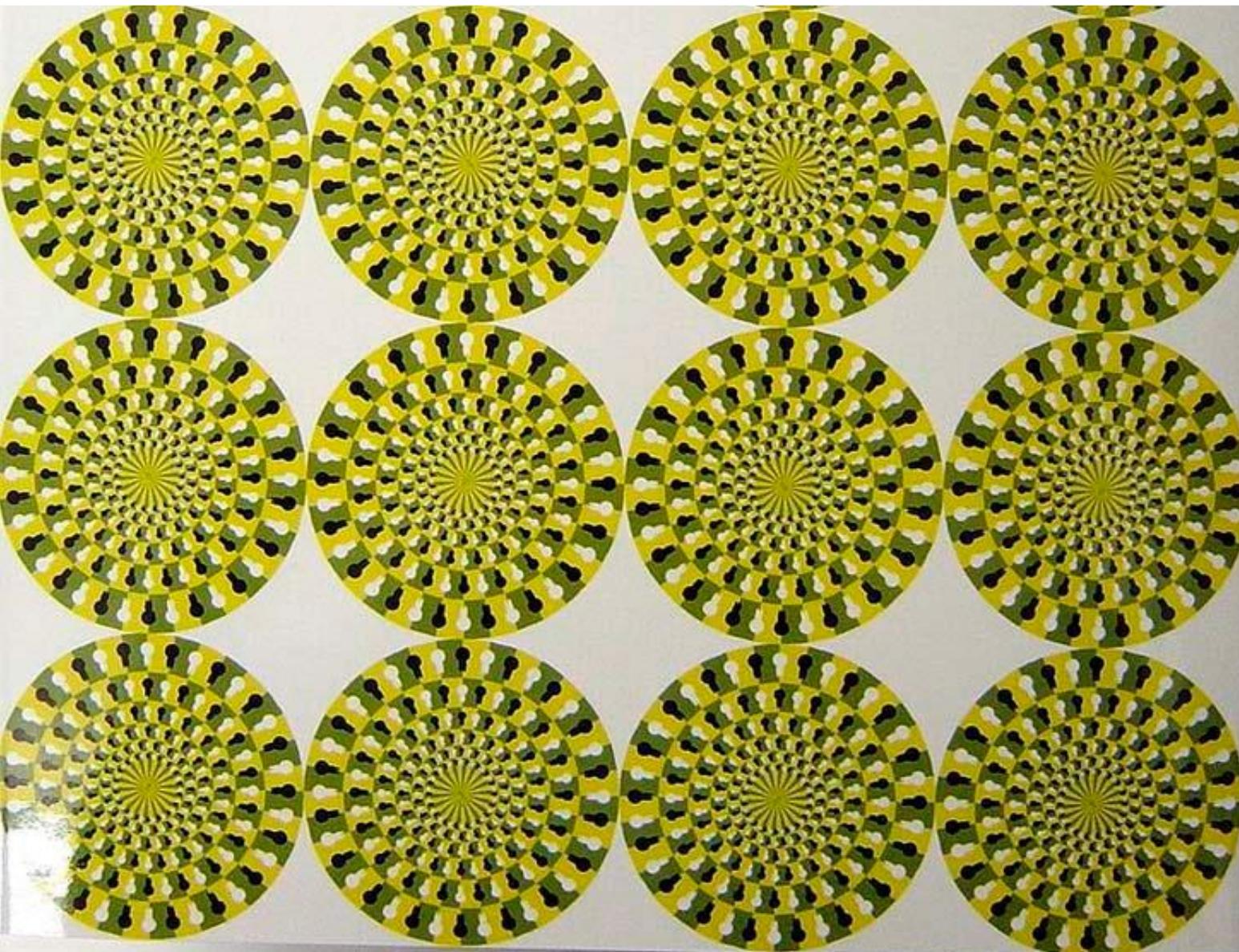


Edward H. Adelson

Image from Computer Vision: Algorithms and Applications  
© 2010 Richard Szeliski, Microsoft Research

# Illusioni ottiche

---



Autonomous Mobile Robots  
Margarita Chi, Martin Rufli, Roland Siegwart

[www.donparrish.com/FavoriteOpticalIllusion.html](http://www.donparrish.com/FavoriteOpticalIllusion.html)

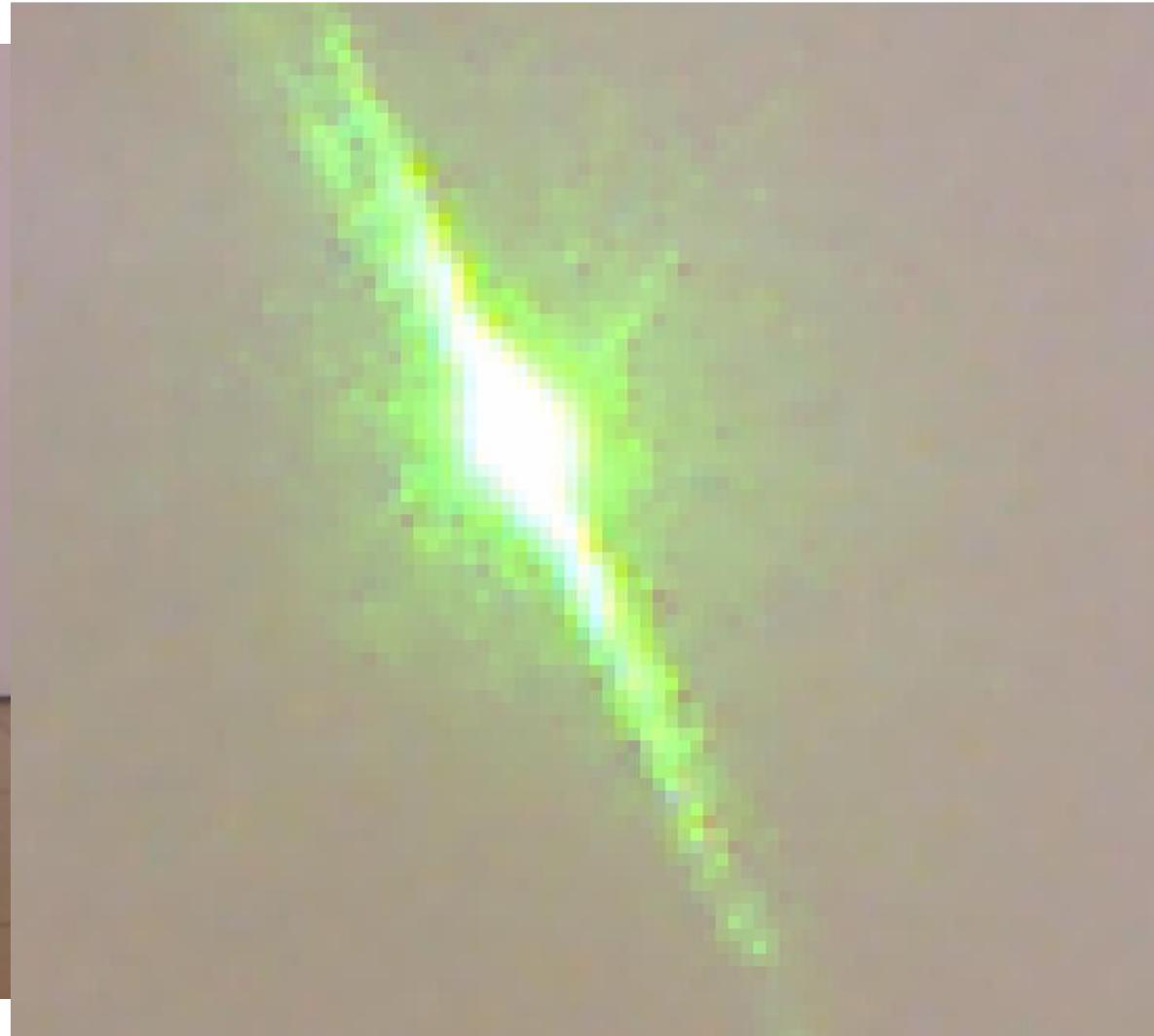
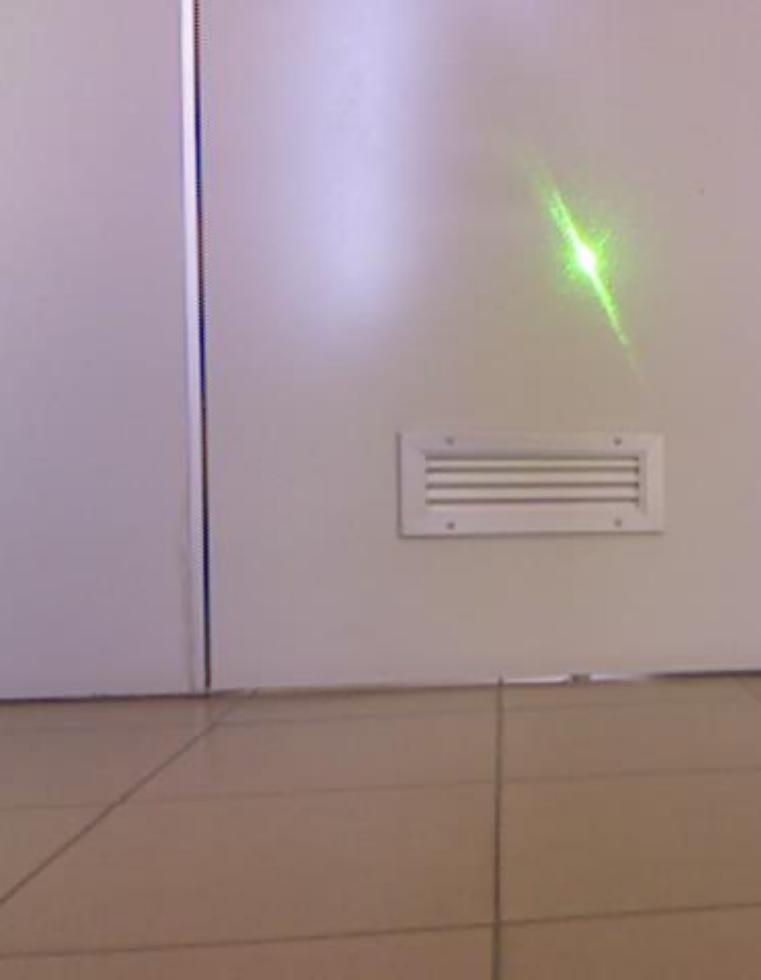
# Trova il punto verde

---



# È davvero verde?

---



# Immagine Digitale

---

- Una immagine digitale è una matrice di pixel
- Il termine pixel deriva da *picture element*
- Il pixel contiene l'informazione relativa alla rappresentazione della realtà che è stata catturata tramite uno scanner, una macchina fotografica o un frame grabber (per i video)



# Campionamento e Quantizzazione

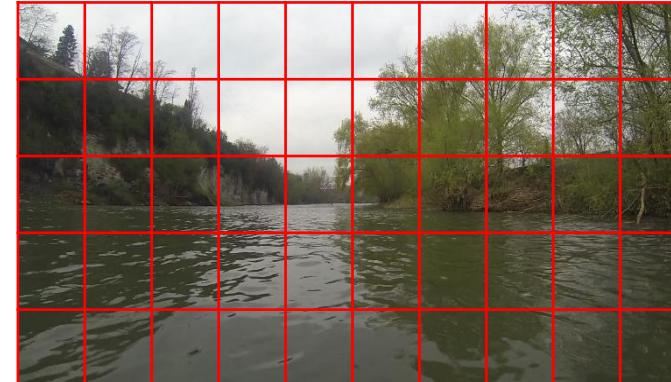
L'immagine è tradotta in un insieme di valori numerici attraverso due fasi:

1. Campionamento spaziale
  2. Quantizzazione numerica

## Mondo reale proiettato sul sensore di acquisizione



## Campionamento



Quantizzazione  
4 valori {0,1,2,3}

A 5x11 grid of numbers from 0 to 3, with a red border around the grid. The background shows a landscape with trees and water.

# Dimensioni

---

- La dimensione dell'immagine è rappresentata dal numero dei pixel che la compongono
- Per esprimere la dimensione si usa il formato:  
**WxH**  
dove W indica il numero di pixel orizzontali e H il numero di pixel verticali  
Esempio: 640x480 pixel



# Risoluzione

---

Con il termine risoluzione si indica la densità dei pixel in relazione alla dimensione del supporto di visualizzazione (per esempio un foglio di carta o uno schermo)

- Si esprime comunemente in pixel per inch (**ppi**) o dot per inch (**dpi**)
- Ad esempio, nel caso si voglia stampare una immagine potremo selezionare la risoluzione attraverso un valore del tipo *300 dpi*
- La risoluzione può essere vista come la capacità di dettaglio di una immagine. Più è grande la risoluzione migliore è la discriminazione dei dettagli.

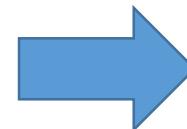
# Risoluzione: esempio pratico

---



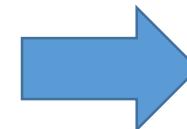
24 pollici  
(grandezza dello schermo)

Dimensioni  
 $1920 \times 1080$   
cioè Full HD



Risoluzione  
91,79 ppi

Dimensioni  
 $3840 \times 2160$   
cioè Ultra HD (4k)



Risoluzione  
183,58 ppi

# Profondità di colore

---

- Ogni pixel contiene un quantità di informazione che può essere espressa in **bit** (binary digit).
- il numero di bit riservati per ogni pixel viene denominato **profondità di colore**
- Data la profondità di colore N, il numero di possibili tonalità per una immagine digitale è  $2^N$

Esempi:

$N = 1 \Rightarrow 2$  tonalità

$N = 4 \Rightarrow 16$  tonalità

$N = 8 \Rightarrow 256$  tonalità



The F1 Story Of 2016: 8-Bit, Video-Game-Style!

[https://www.youtube.com/watch?v=U4E9Qx5\\_ITY](https://www.youtube.com/watch?v=U4E9Qx5_ITY)

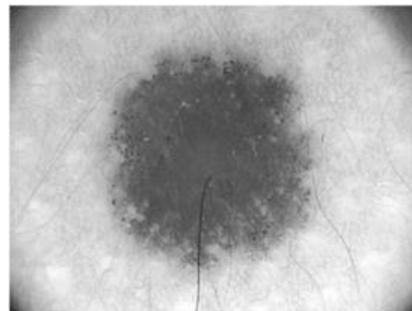
# Immagini e colori

---

- Immagine binaria  
(due valori, per es. 0 e 255)



- Immagine in scala di grigi  
(valori nel range 0÷255)

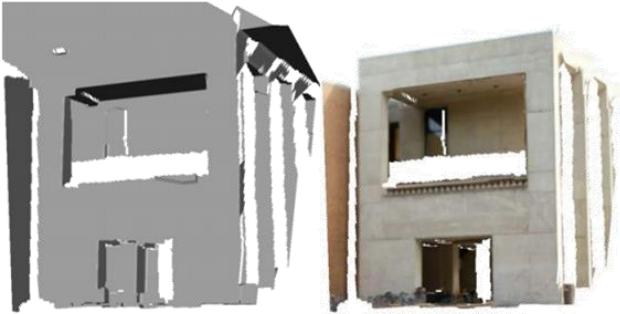
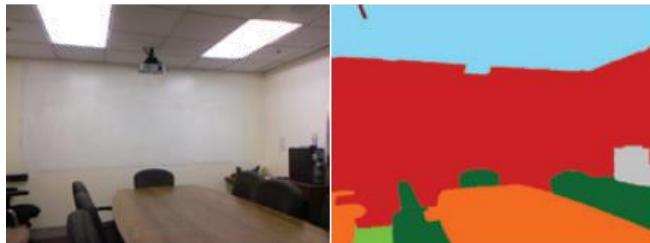
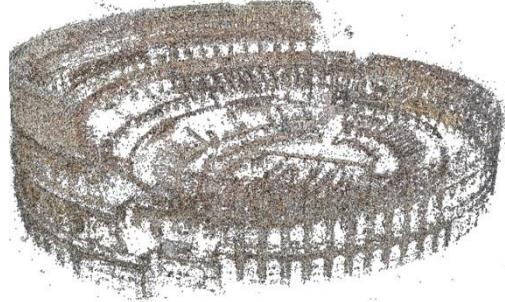


- Immagine a colori  
(3 canali R, G e B)



# Ricostruzione 3D da immagini

---



Point Cloud (PC):  
Depth information  
Image Saliency

Multi View (MV):  
Multiple Image  
Saliency

Single View (SV):  
Image Saliency

[Silberman et al. 2011]

[Fukurawa et al. 2009]

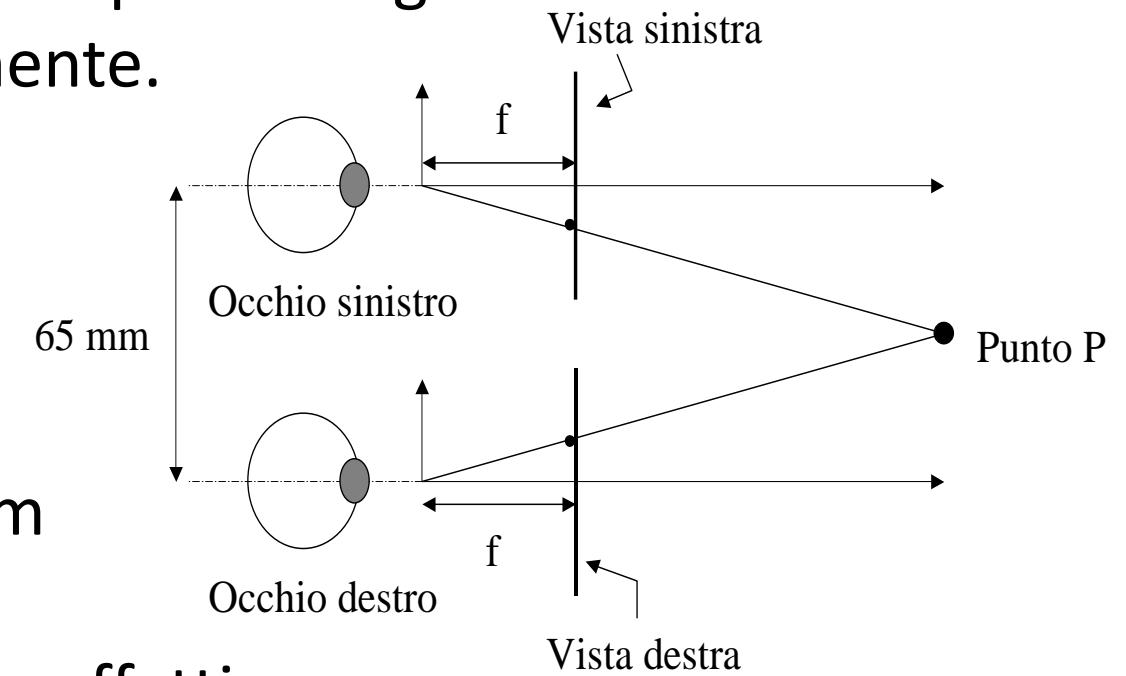
[Lee et al. 2009]

# Stereo Visione

---

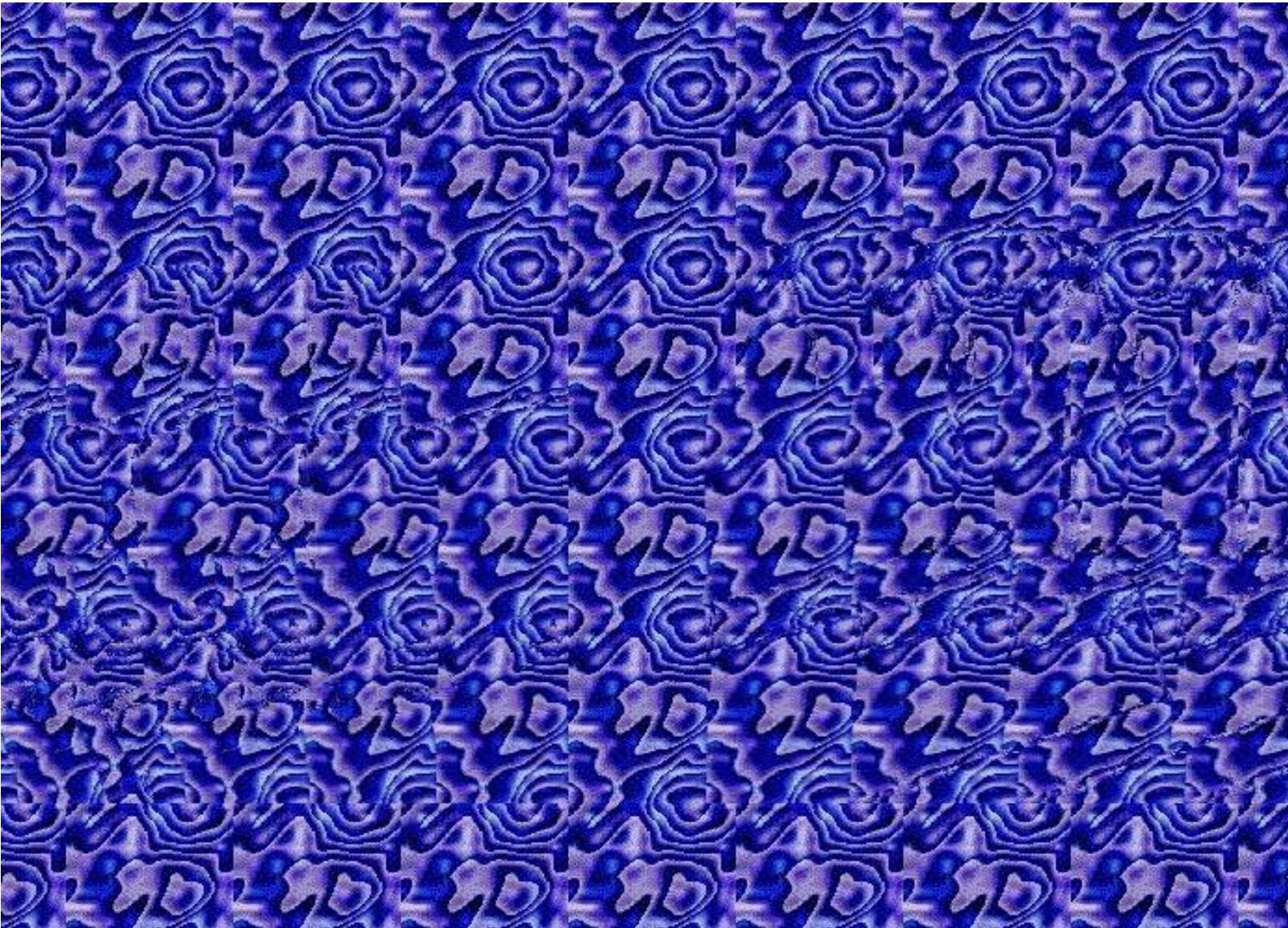
L'analisi stereo è il processo di misurazione della distanza da un oggetto, basato sul confronto di due o più immagini dell'oggetto stesso ottenute simultaneamente.

La percezione della terza dimensione, che avvertiamo attraverso i nostri occhi, deriva dal fatto che i due bulbi oculari hanno i loro assi ottici distanti circa 65 mm e forniscono due immagini leggermente diverse degli oggetti che, sommando i loro effetti, procurano il senso della profondità



# Stereogramma

---

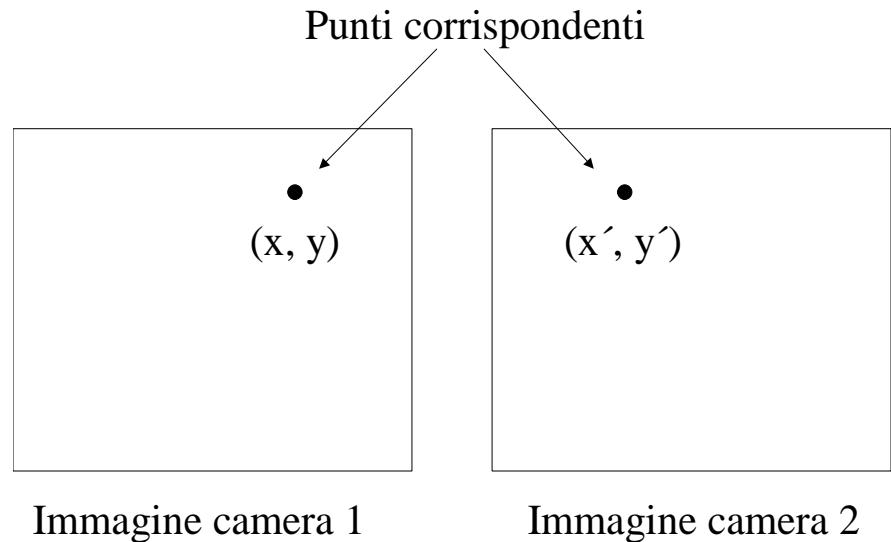


# Correlazione tra punti

---

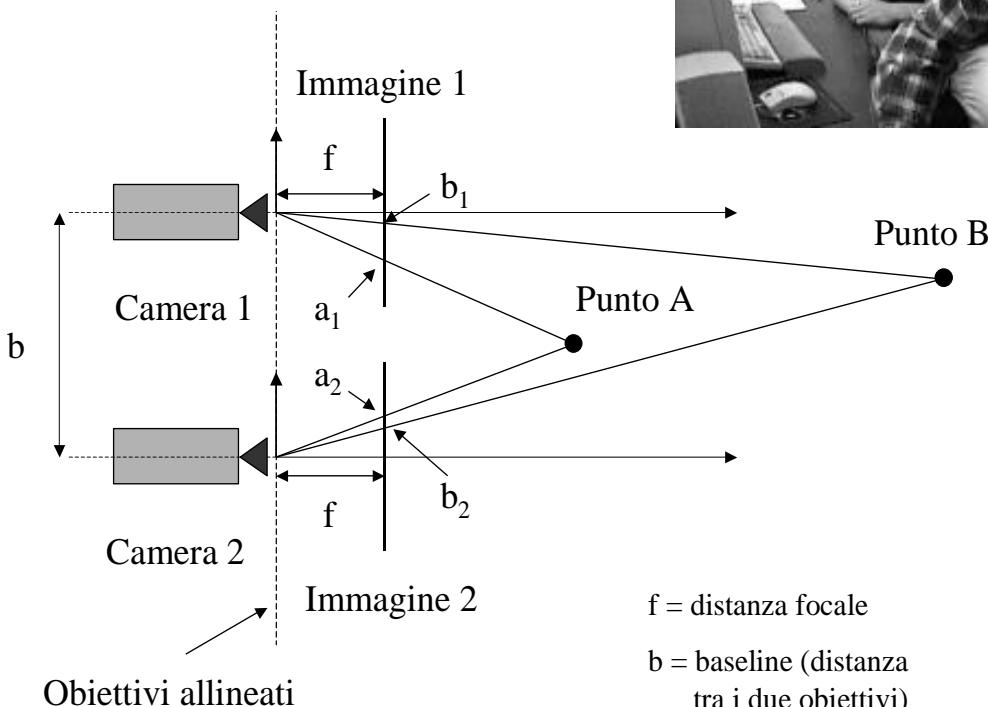
Il problema fondamentale nell'analisi stereo è quello di trovare la corrispondenza tra gli elementi delle varie immagini disponibili.

Una volta che tale corrispondenza è stata scoperta, la distanza dall'oggetto può essere ottenuta tramite l'ottica geometrica.



La coppia di locazioni  $(x, y)$  e  $(x', y')$  è unica. Proprio perché tale coppia è unica, se si riescono a trovare le due locazioni che corrispondono allo stesso identico punto nello spazio, allora è possibile risalire alle coordinate tridimensionali di detto punto.

# Correlazione area-based



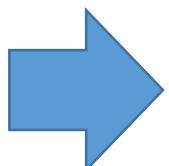
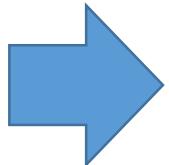
disparità per il punto A:  $d(A) = a_1 - a_2$

disparità per il punto B:  $d(B) = b_1 - b_2$

Troviamo  $d(A) > d(B)$ , poiché  $a_1 > b_1$  e  $a_2 < b_2$

# Rettificazione

---

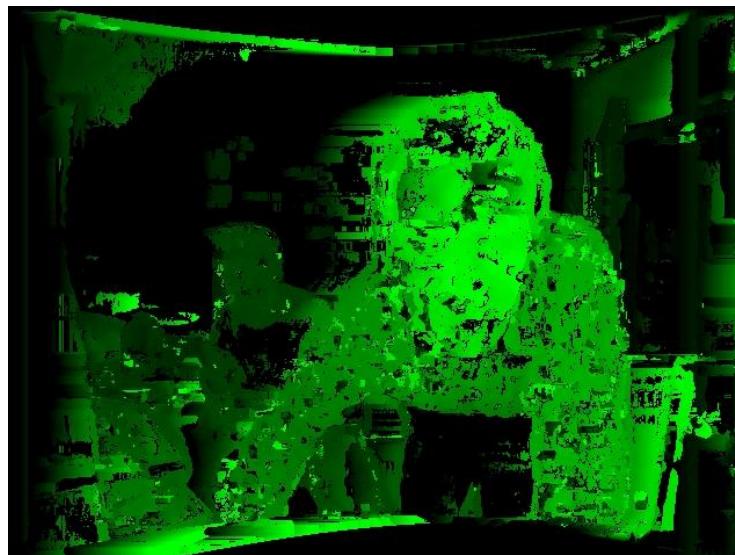


# Mappa di disparità

---



Coppia stereo  
rettificata



Mappa di disparità

# Ricostruzione 3D con stereo visione

---



# Microsoft Kinect

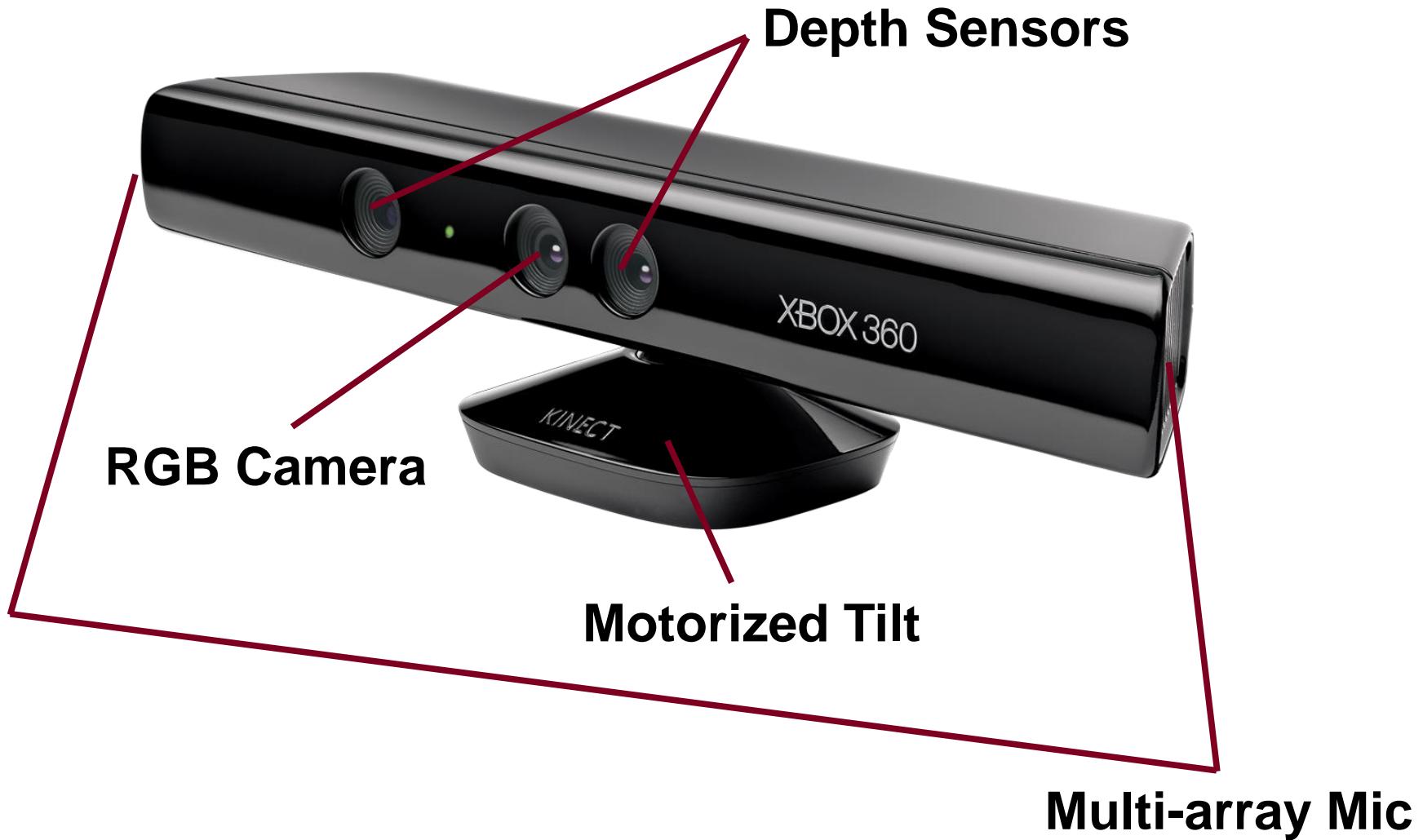
---

- Il sensore Kinect è stato lanciato in Nord America il 4 Novembre 2010
- Il Kinet è un sensore di movimento in grado di fornire informazioni 3D



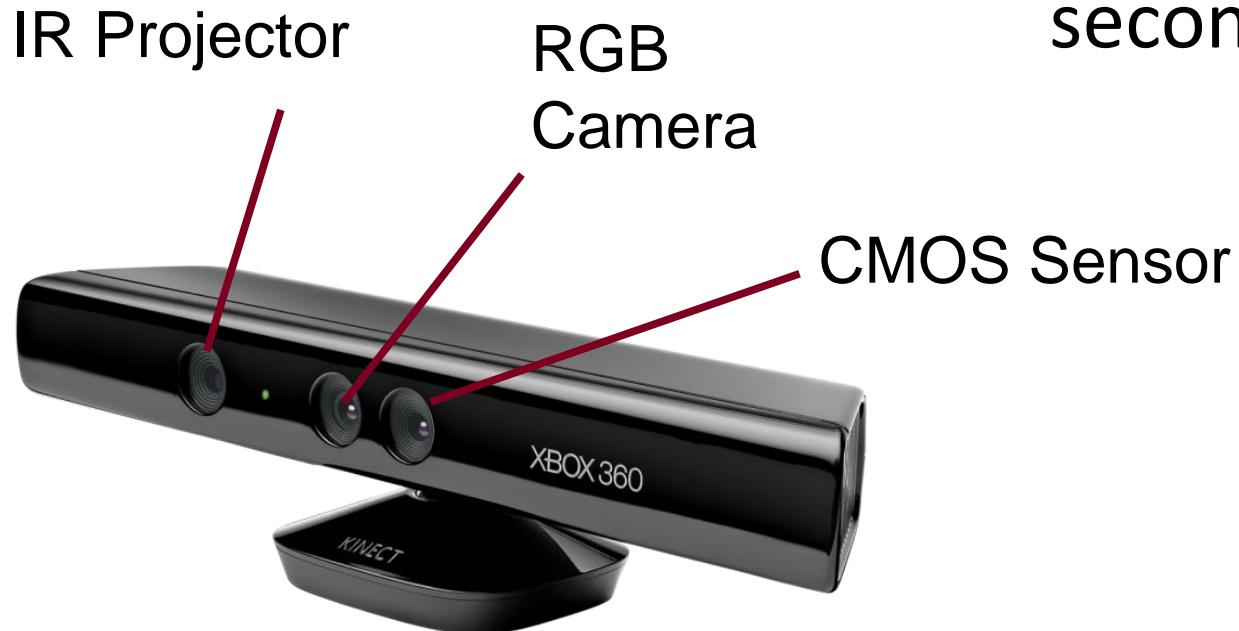
# Sensori del Kinect

---



# Sensori di Disparità

- Consiste in un proiettore di raggi infrarosso (IR) e un sensore CMOS
- L'IR beam riflette sul soggetto e viene catturato dal sensore CMOS
  - 640x480 pixel a 30 frame al secondo (fps)



# Stima della disparità



Il tempo di ritorno viene usato per misurare la distanza degli oggetti dal sensore



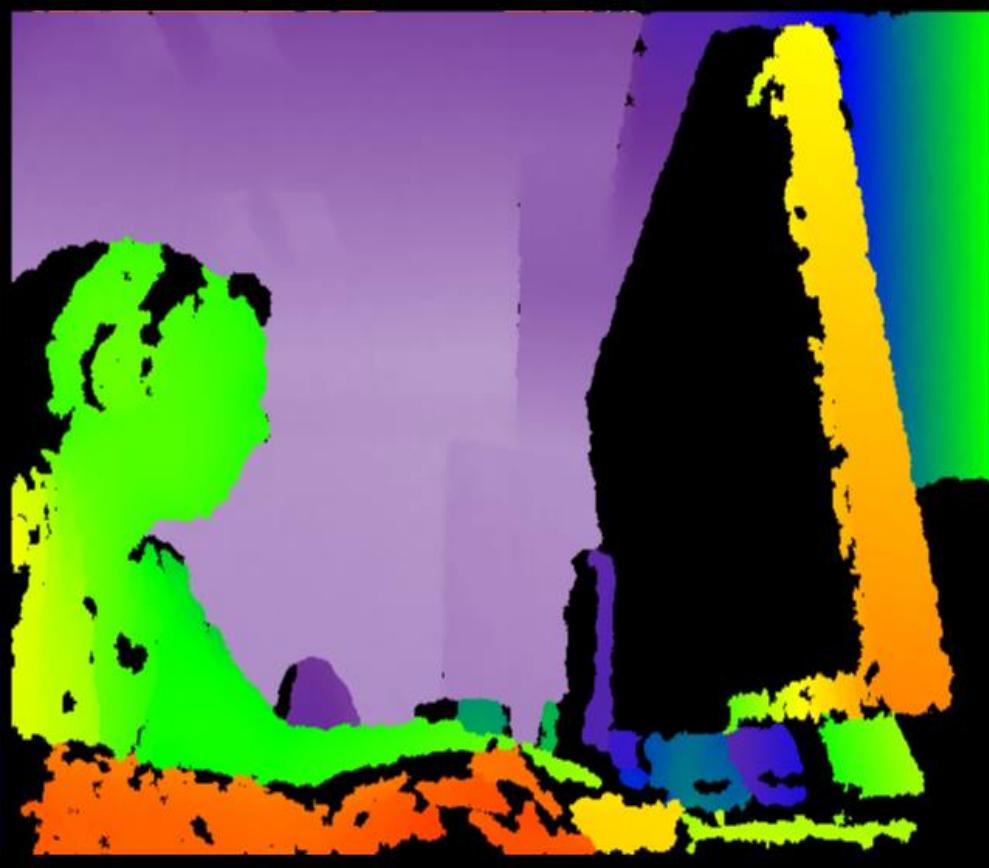
# Immagine a colori + immagine di disparità

---

24-bit RGB data

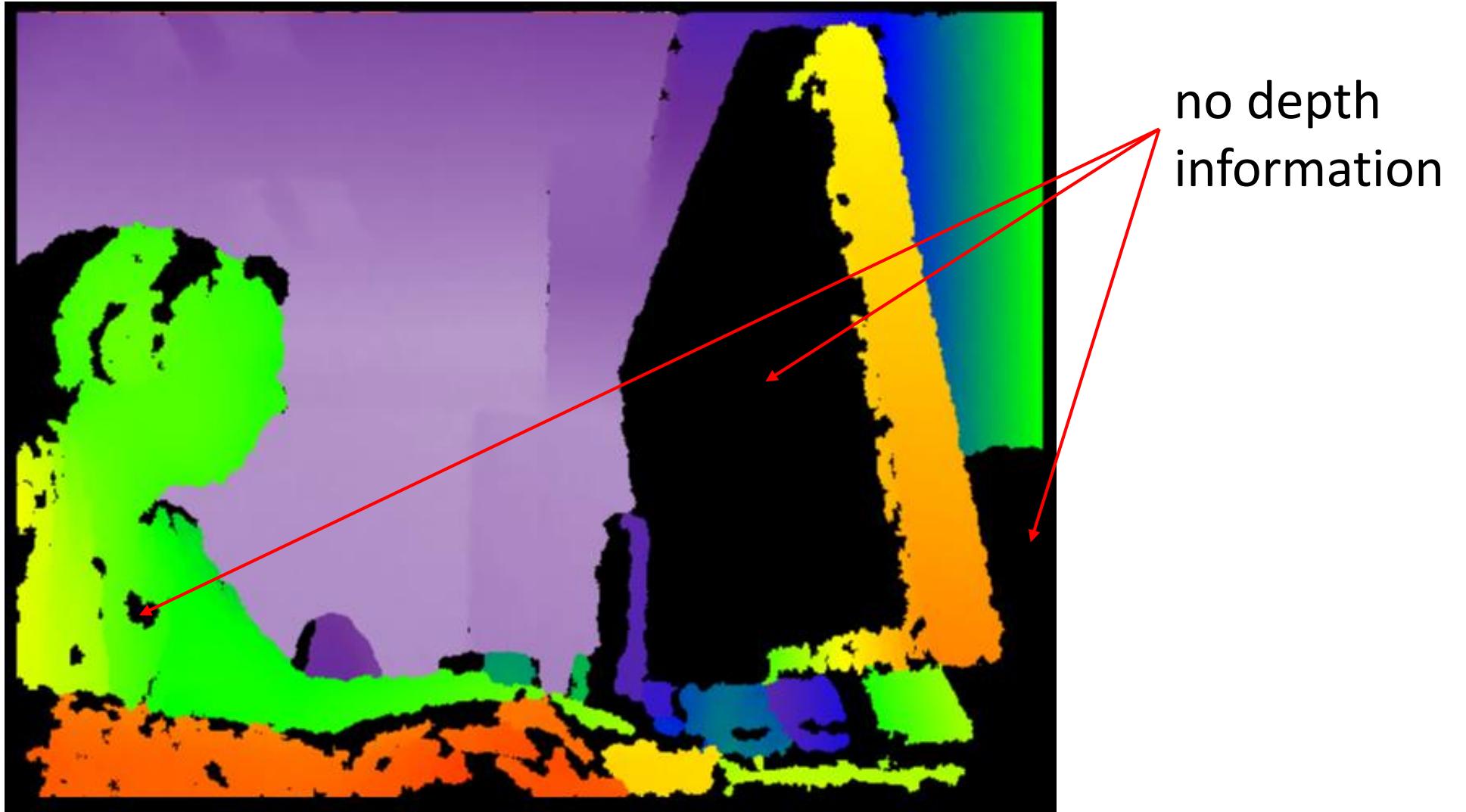


11-bit depth data



# Mappa di disparità densa

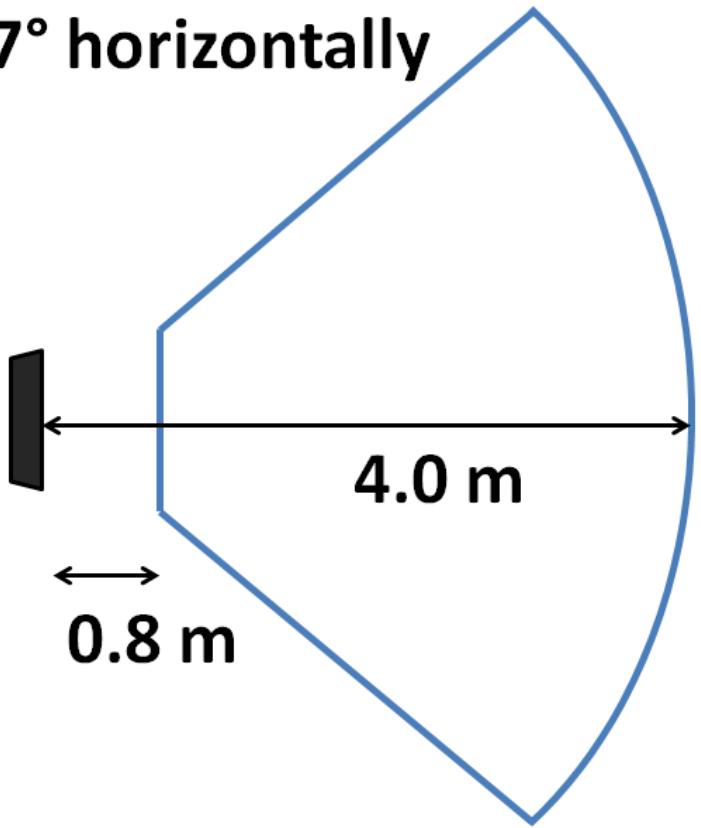
---



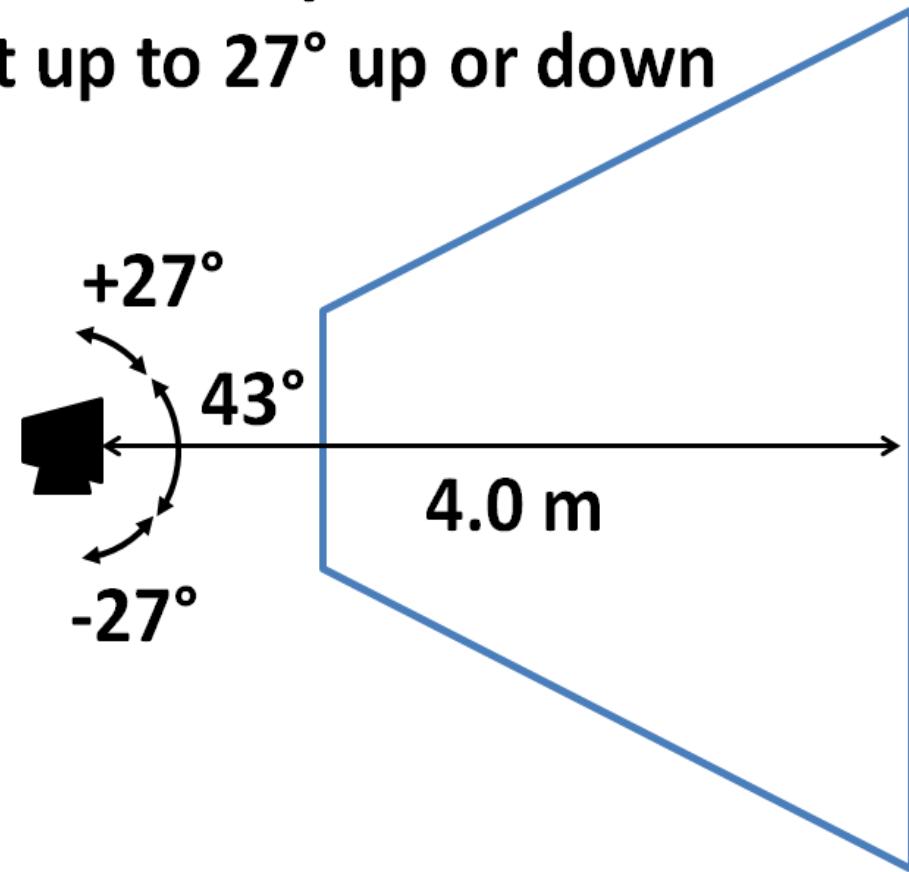
# Limiti fisici del Microsoft Kinect

---

Angular field of view:  
57° horizontally



Angular field of view:  
43° vertically  
Tilt up to 27° up or down



# Ricostruzione 3D con Microsoft Kinect

---

<http://graphics.stanford.edu/~mdfisher/Kinect.html>



point  
cloud

# Esempio Kinect

---

Online Simultaneous Localization and Mapping  
with  
RTAB-Map and TORO on ROS

**Mathieu Labbé**

*[www.rtabmap.googlecode.com](http://www.rtabmap.googlecode.com)*



January 2013

Link al video: <https://www.youtube.com/watch?v=AMLwjo80WzI>

# Kinect 2.0 & Intel RealSense

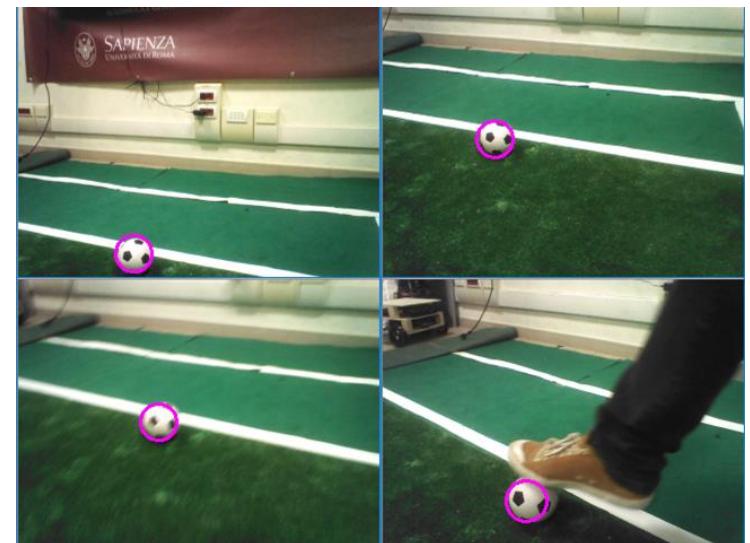
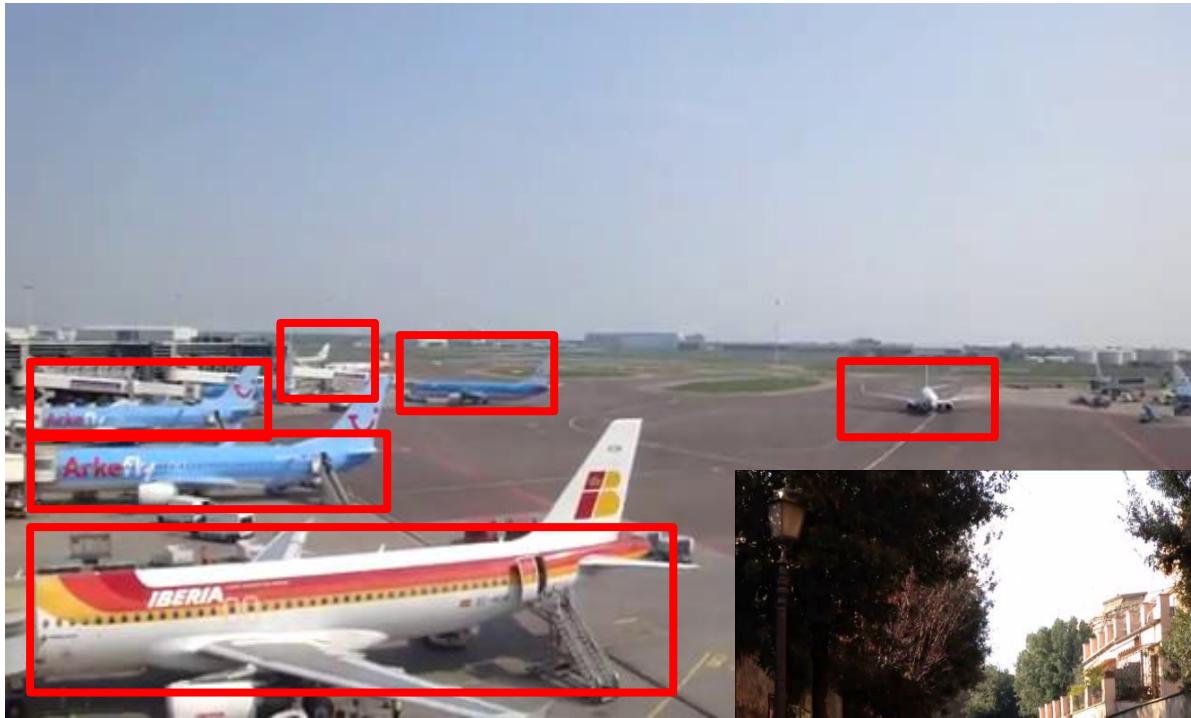
---

- Typical characteristic
  - Resolution 1920x1080 pixels
  - Field of view: 70 deg (H), 60 deg (V)
  - Claimed accuracy: 1 mm
  - Claimed max range: 6 meters



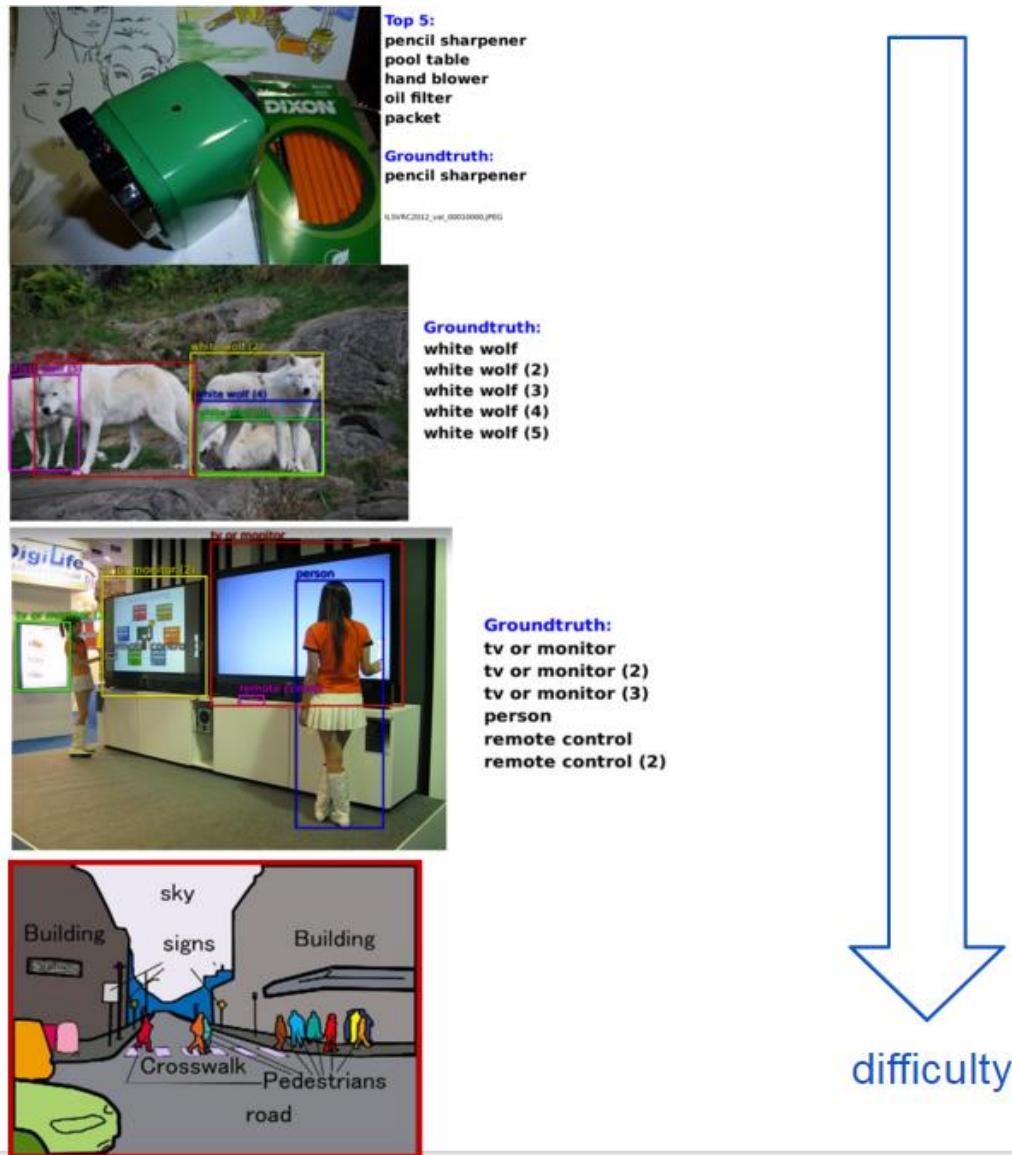
<https://www.youtube.com/watch?v=yvgPrZNp4So>

# Oggetti nelle immagini



# Classificazione, localizzazione, detection e segmentation

- classification
- localization
- detection
- segmentation



# Classificazione

---



Barca  
Lago  
Acqua  
Nuvole

# Localizzazione

---

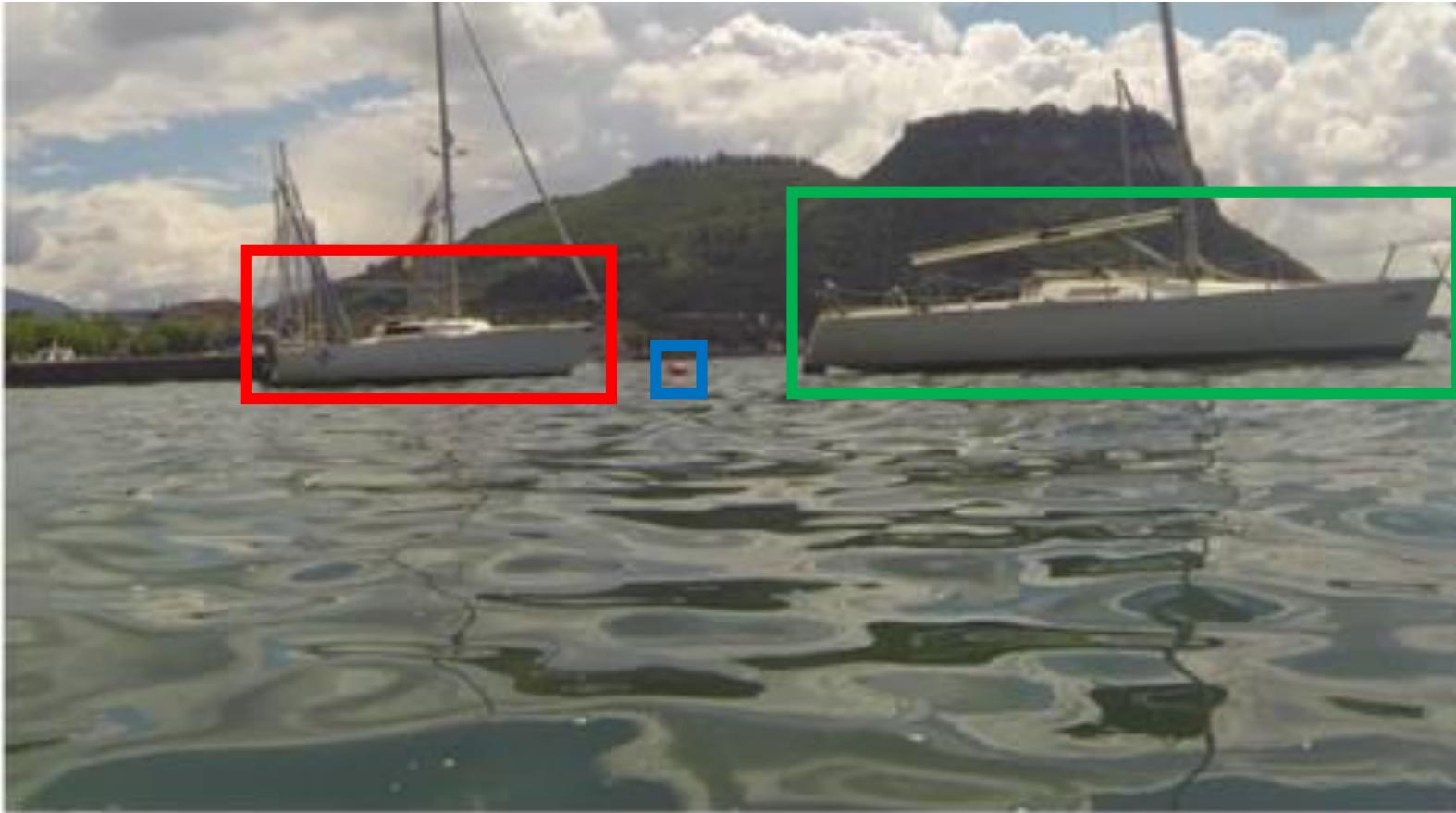


Barca 1

Barca 2

# Detection

---



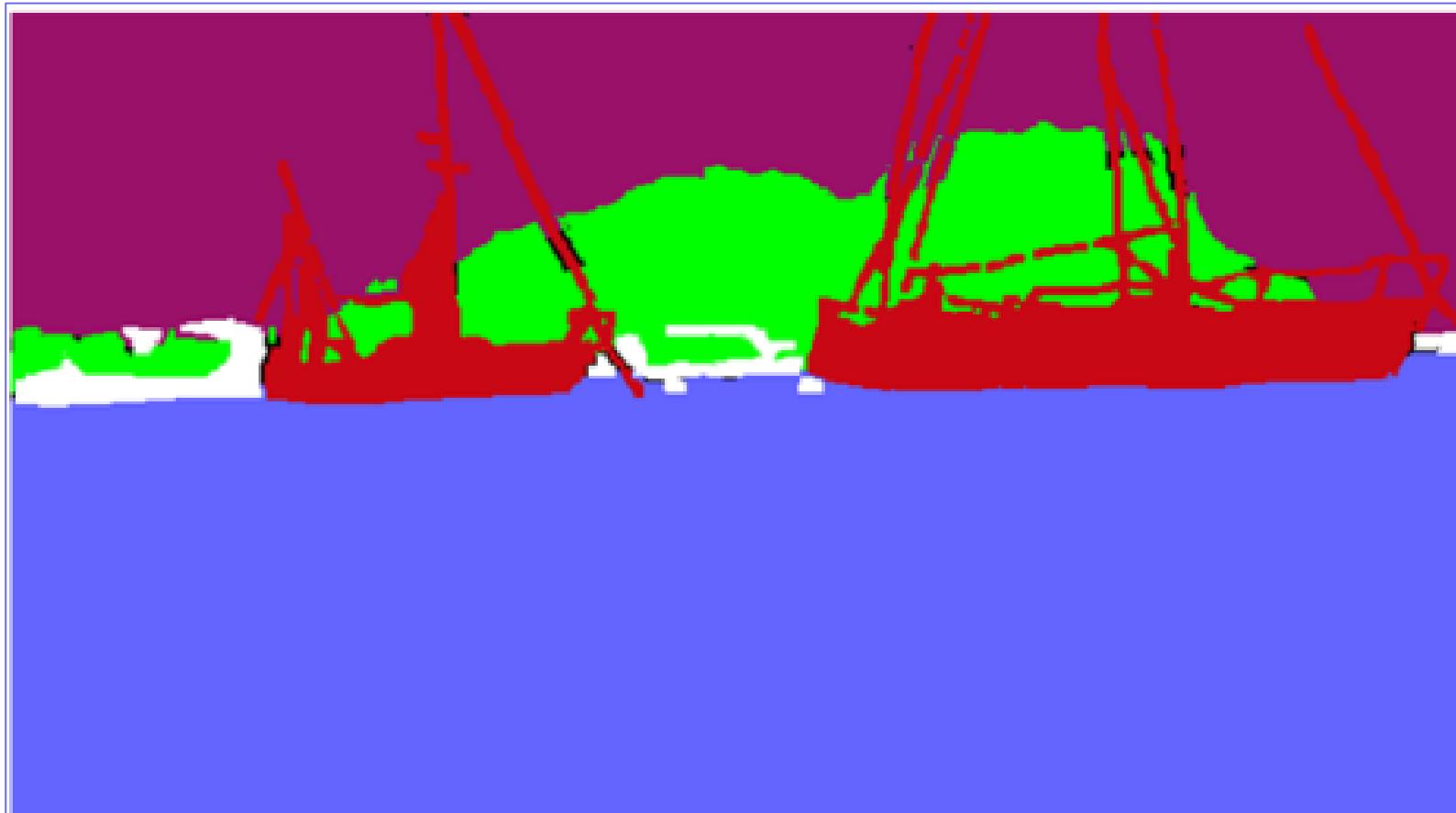
Barca 1

Barca 2

Boa 1

# Segmentation

---



- █ water
- █ boat
- █ vegetation
- █ other

# Face Detection Problem

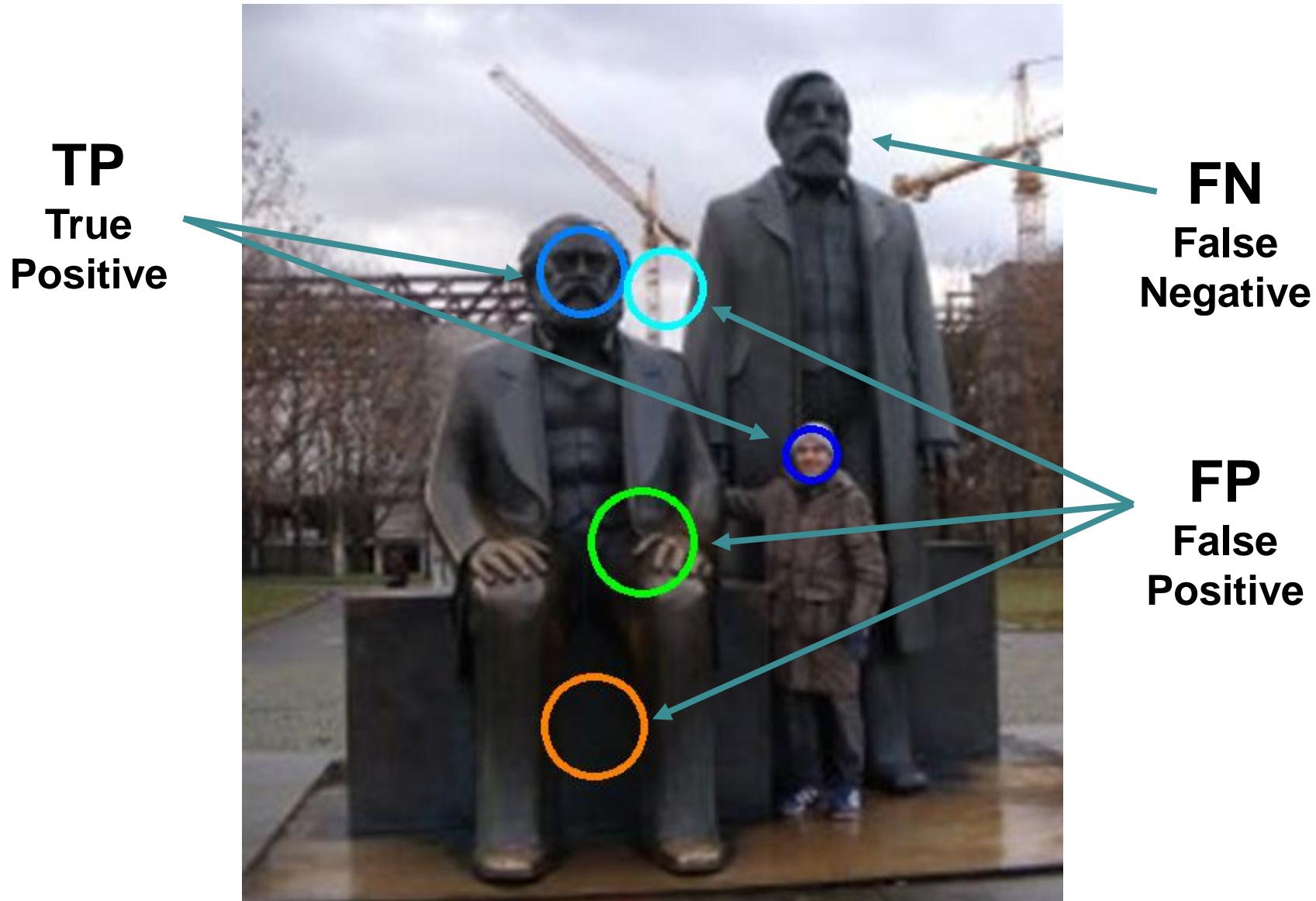
---

Trovare le regioni  
dell'immagine  
contenenti una  
istanza della classe  
“faccia”



# Problematiche nel Face Detection

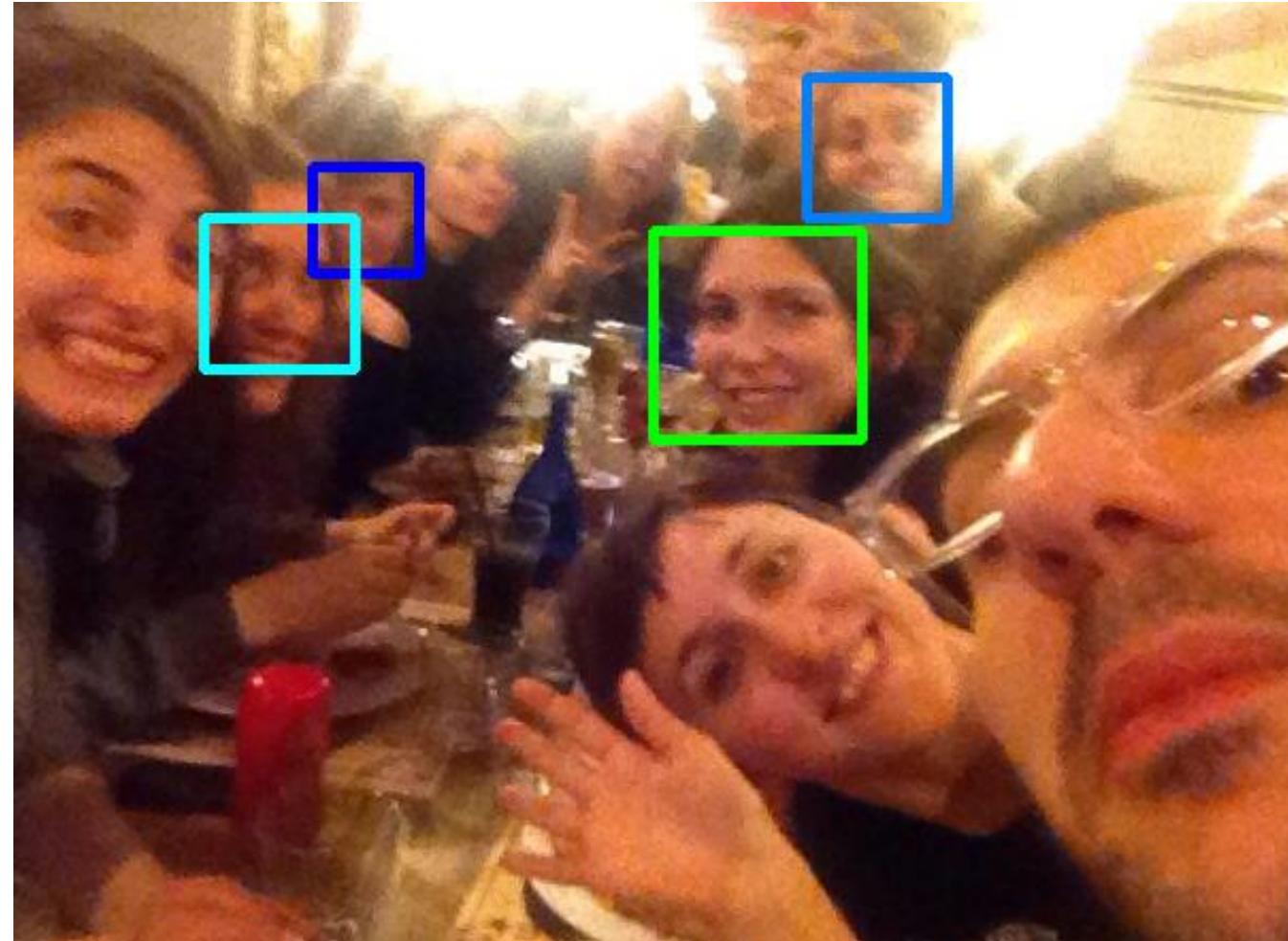
---



# Problematiche aggiuntive

---

- Rotazione
- Blurring
- Illuminazione
- Occlusioni
- Occhiali
- Barba
- ...



# Detection vs Identification

---



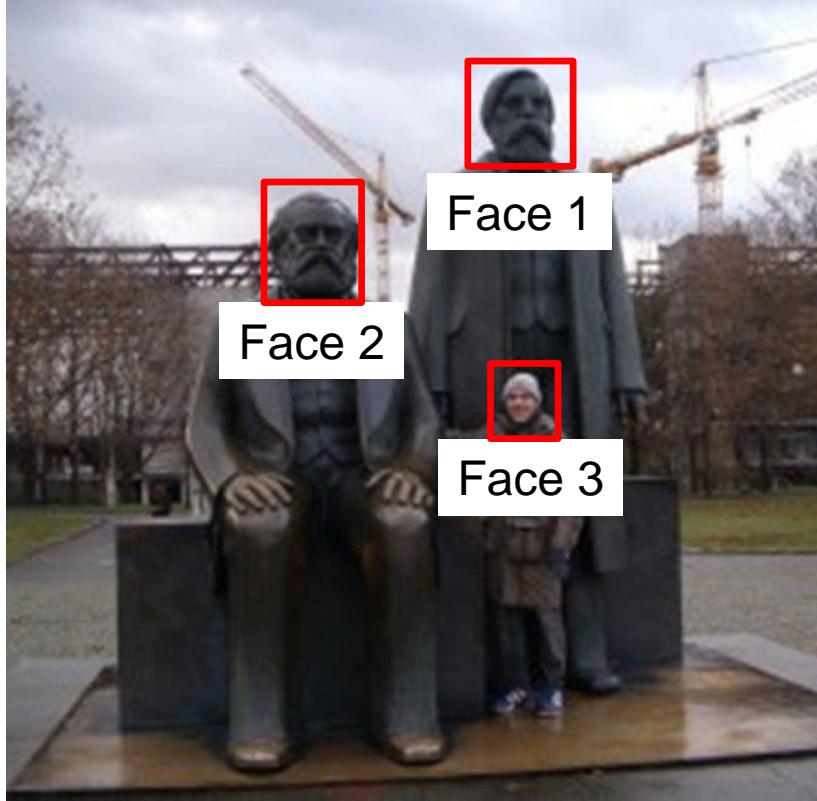
detection



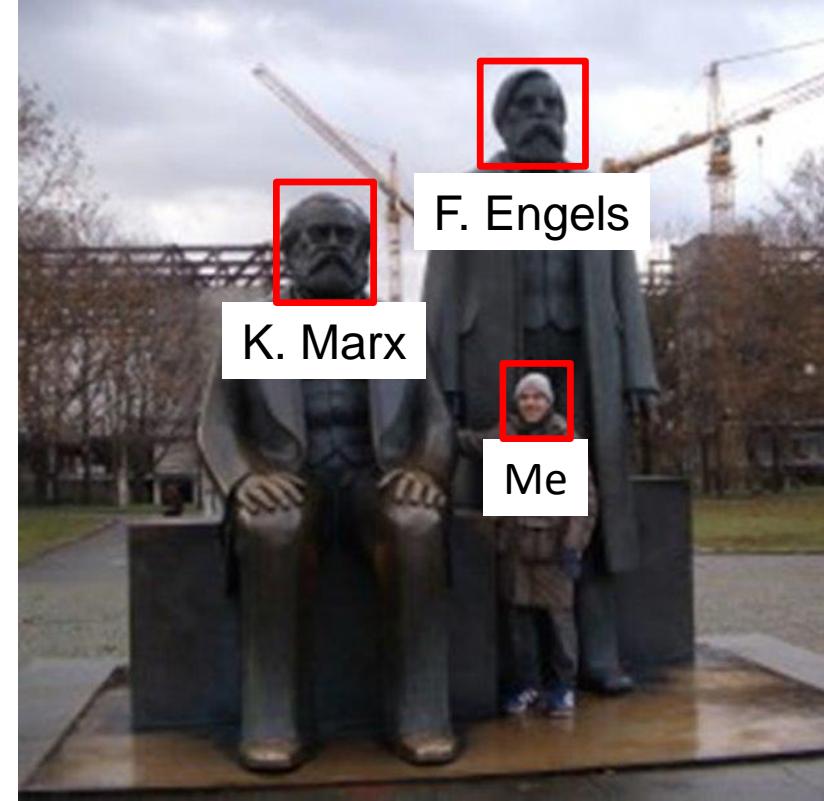
identification

# Detection vs Recognition

---

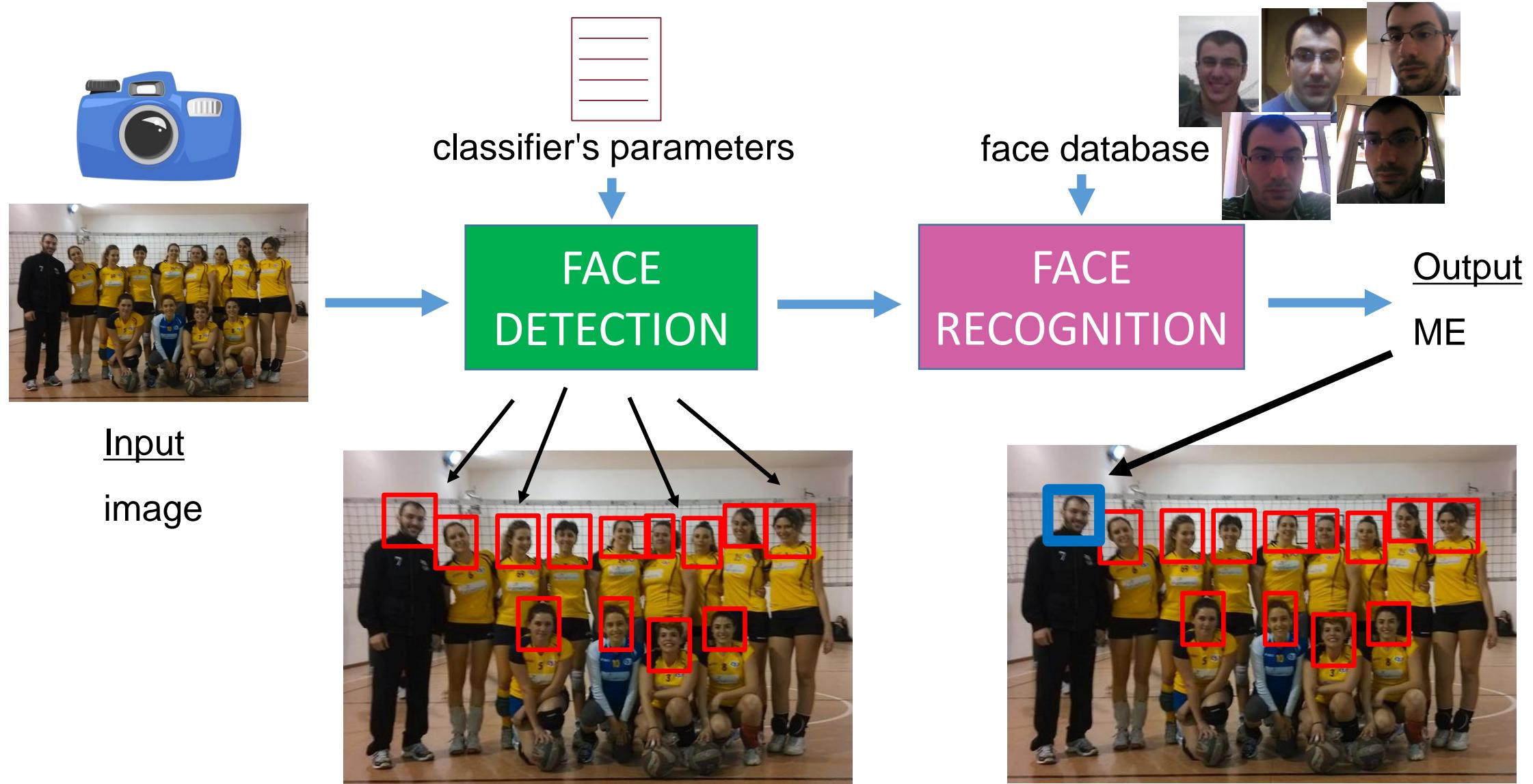


**detection**



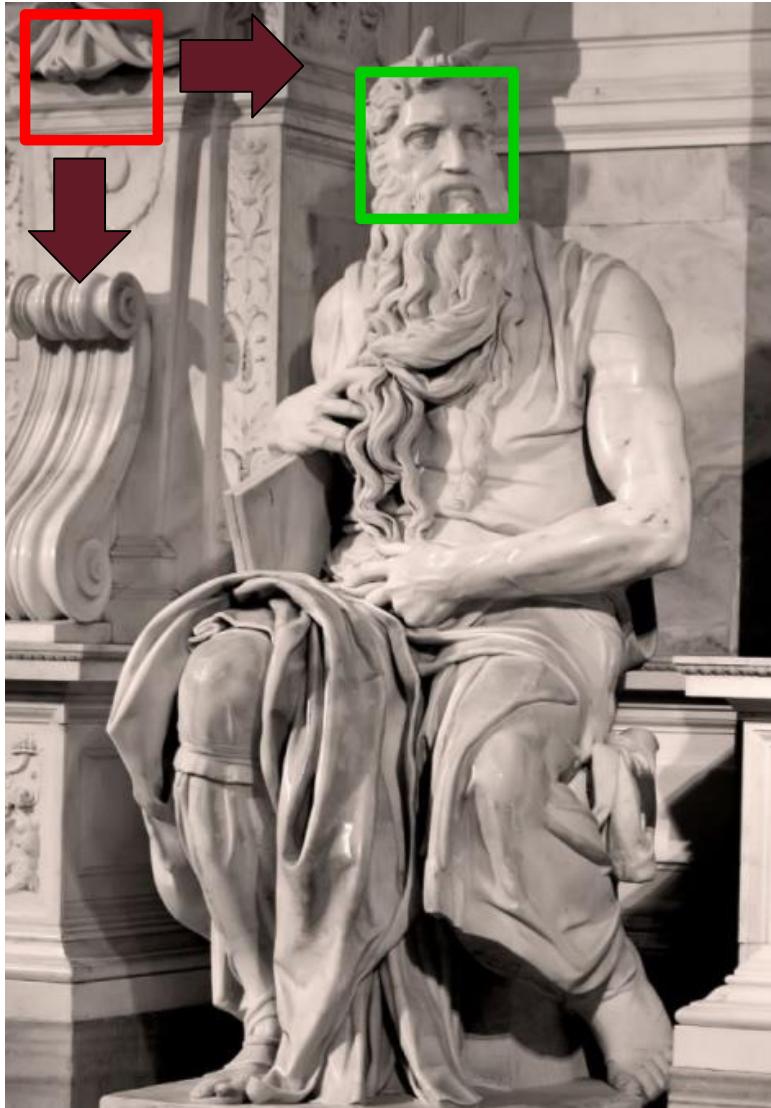
**recognition**

# Detection & Recognition



# Ricerca con finestra mobile

---



- Si fa scorrere una finestra (per esempio 30x30) sull'immagine e si valuta se la porzione di immagine nella finestra sia corrispondente al modello dell'oggetto che si sta cercando
- L'operazione deve essere ripetuta in posizioni situate lungo tutta l'immagine
- Si assume che il numero di possibili oggetti nell'immagine sia limitato rispetto al numero delle posizioni visitate

# Ricerca multiscala

---

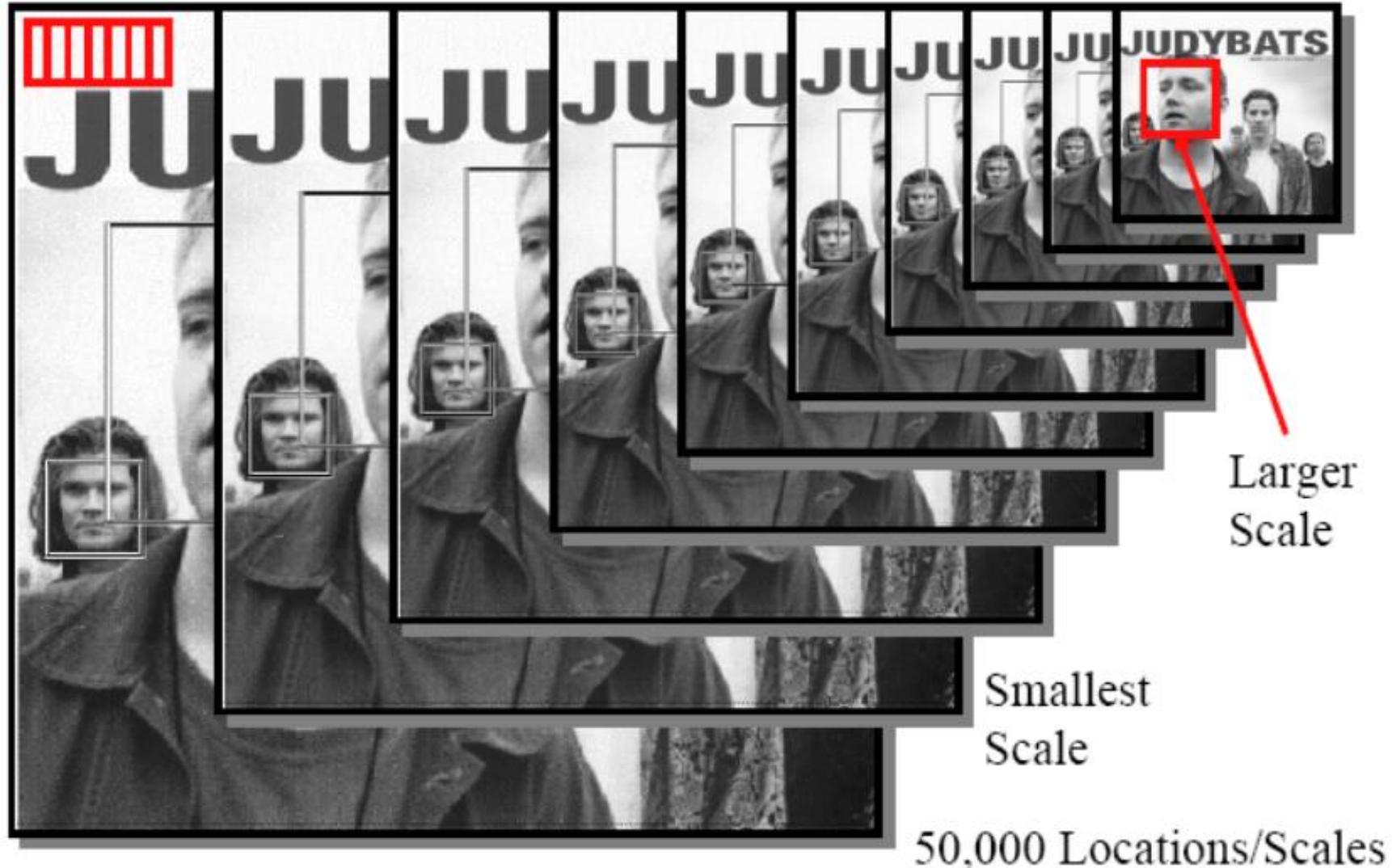


# Ricerca multiscala: resize dell'input

---



# Piramide di immagini



# Passi per la object detection

---

## 1. Feature Computation

What features?  
How can they be computed as quickly as possible?

## 2. Feature Selection

What are the most discriminating features?

## 3. Detection (in real time)

Must focus on potentially positive areas

# Algoritmo di Viola and Jones

---

- **Very popular method**
  - **Recognition is very fast  
(e.g., real-time for digital cameras)**
  - **Key contributions**
    1. **Integral image for fast feature extraction**
    2. **Boosting (Ada-Boost) for face detection**
    3. **Attentional cascade for fast rejection of non-face sub-windows**
- 
- A yellow triangular warning sign with a black border and a black exclamation mark in the center, indicating a potential pitfall or important note.
- Training may take a long time**

# Passi nell'algoritmo di Viola and Jones

---

## 1. Feature Computation

### Quick Feature Computation

Rectangle features

Integral image representation

## 2. Feature Selection

### Simple and Efficient Classification

Ada-Boost training algorithm

## 3. Detection (in real time)

### Real-timeliness

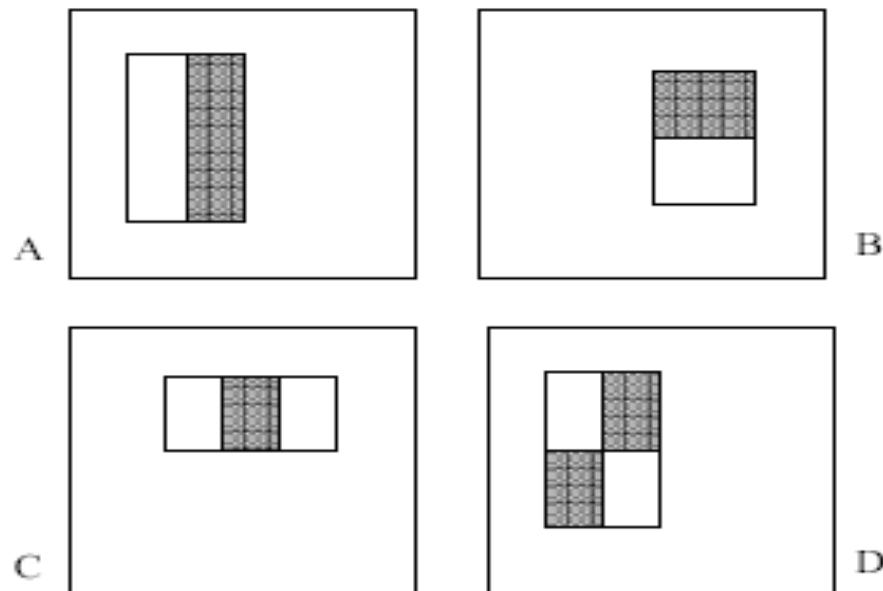
A cascade of classifiers

# Features

---

Four basic types

- Easy to calculate
- White areas are subtracted from the black ones
- Integral image representation makes feature extraction faster



# Features rettangolari

---

La principale motivazione dietro l'uso di features rettangolari, rispetto a filtri più espressivi, è data dalla grande efficienza computazionale che si può raggiungere usando immagini integrali



# Selezione delle Features

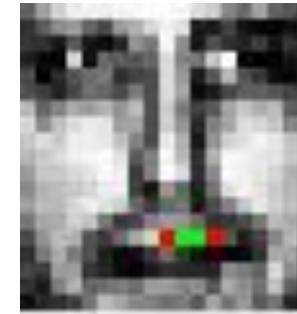
---

At test time, it is impractical to evaluate the entire feature set

We want a subset of relevant features, which are informative to model a face



Relevant feature

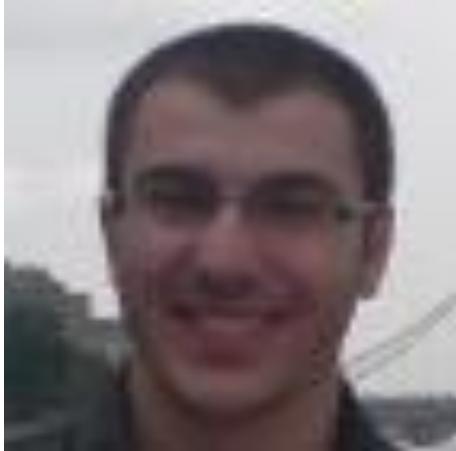


Irrelevant feature

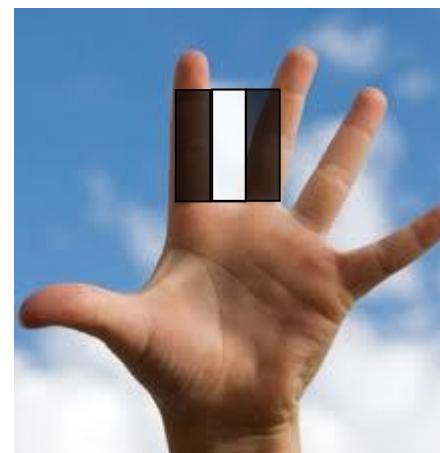
- Can we create a good classifier using just a small subset of all possible features?
- How to select such a subset?

# Features Rettangolari

---

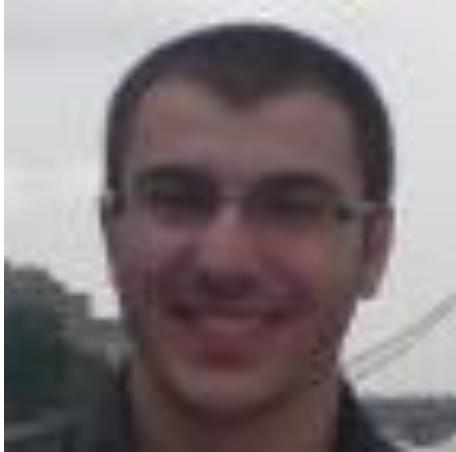


$\text{Value} = \sum (\text{pixels in white area}) - \sum (\text{pixels in black area})$

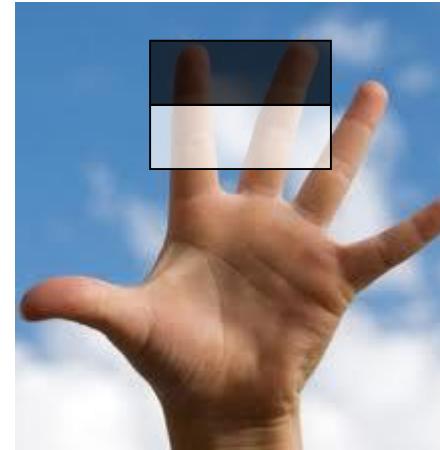


# Features Rettangolari

---

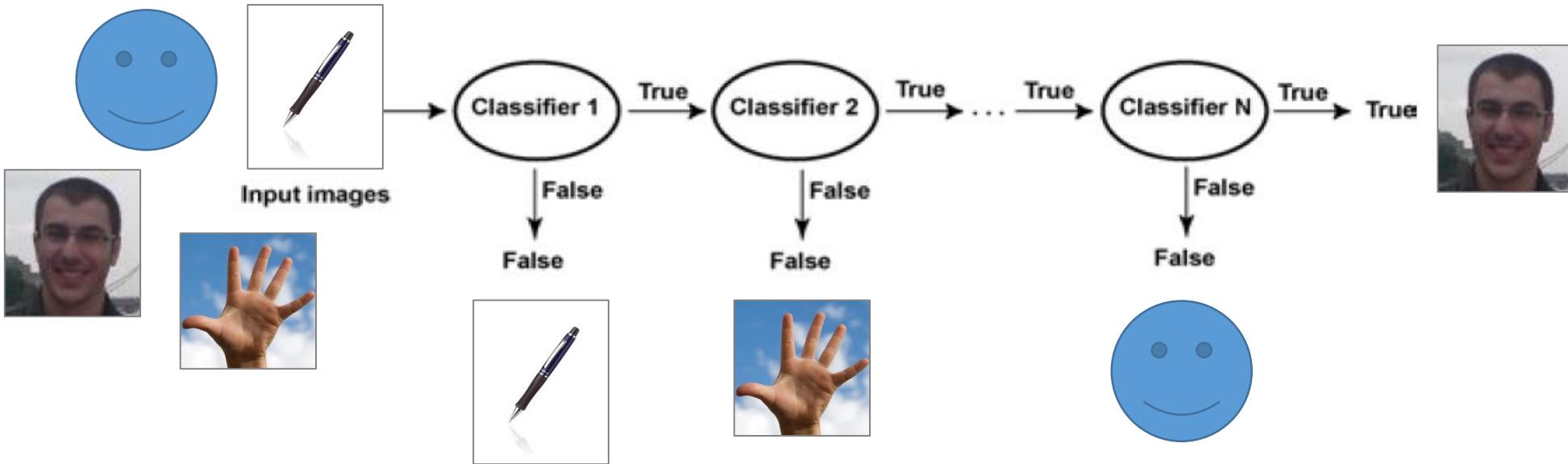


$\text{Value} = \sum (\text{pixels in white area}) - \sum (\text{pixels in black area})$



# Cascata di classificatori

---



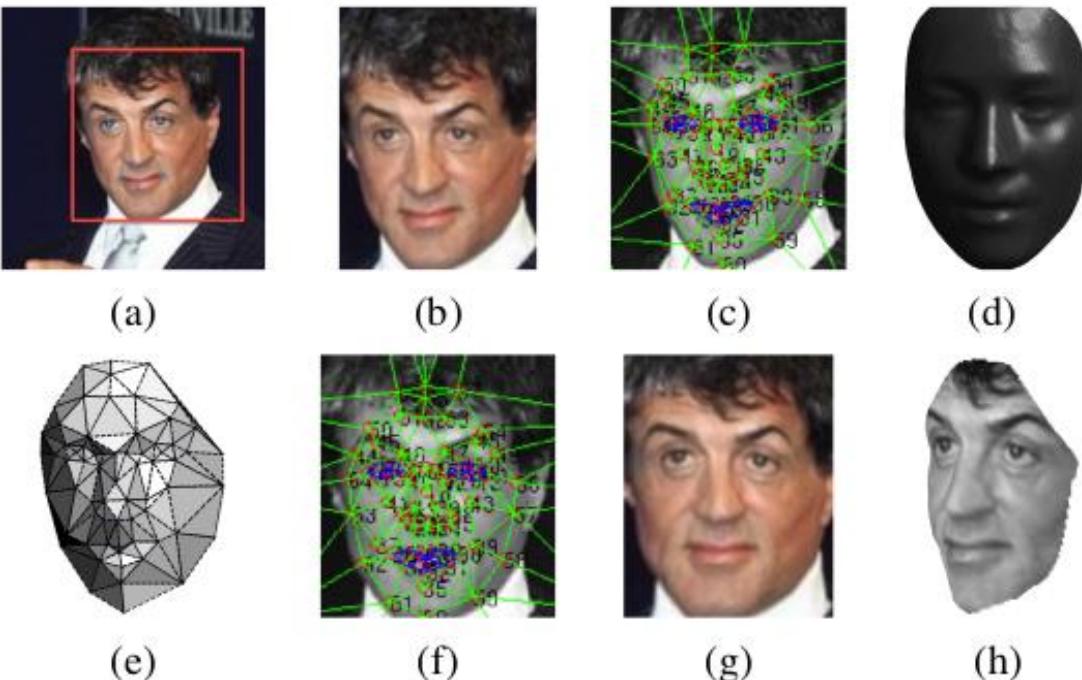
- A chain of classifiers that each reject some fraction of the negative training samples while keeping almost all positive ones
- Each classifier is an AdaBoost ensemble of rectangular Haar-like features sampled from a large pool

# Face detection in fb

---



# Face Alignment



**Figure 1. Alignment pipeline.** (a) The detected face, with 6 initial fiducial points. (b) The induced 2D-aligned crop. (c) 67 fiducial points on the 2D-aligned crop with their corresponding Delaunay triangulation, we added triangles on the contour to avoid discontinuities. (d) The reference 3D shape transformed to the 2D-aligned crop image-plane. (e) Triangle visibility w.r.t. to the fitted 3D-2D camera; darker triangles are less visible. (f) The 67 fiducial points induced by the 3D model that are used to direct the piece-wise affine warpping. (g) The final frontalized crop. (h) A new view generated by the 3D model (not used in this paper).

Image from  
*DeepFace: Closing the Gap to Human-Level Performance in Face Verification*

# Deep Face

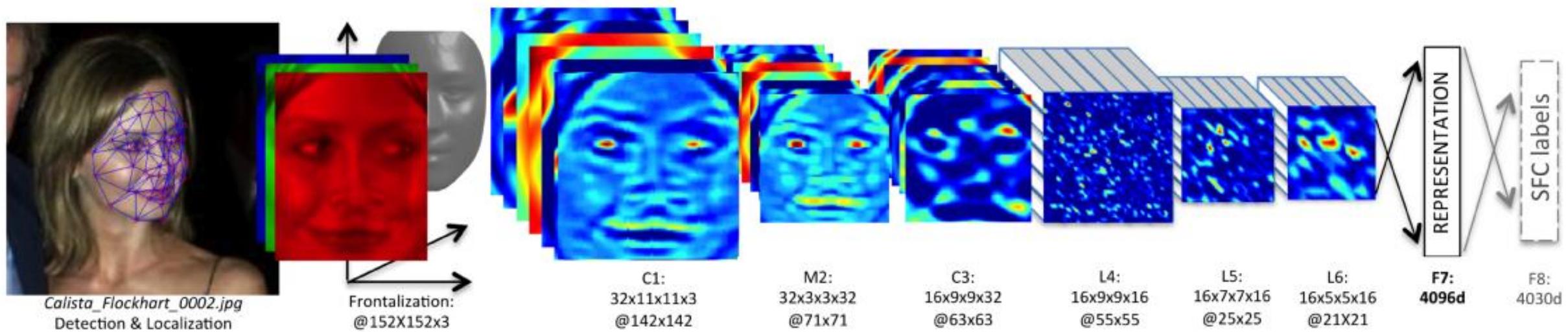


Figure 2. **Outline of the DeepFace architecture.** A front-end of a single convolution-pooling-convolution filtering on the rectified input, followed by three locally-connected layers and two fully-connected layers. Colors illustrate feature maps produced at each layer. The net includes more than 120 million parameters, where more than 95% come from the local and fully connected layers.

Lettura consigliata

Y. Taigman, M. Yang, M. Ranzato, L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," in IEEE Conference on Computer Vision and Pattern Recognition, pp. 1701-1708, 2014

# Il Cybertruck di Tesla e i frigoriferi



<https://analyticsindiamag.com/deeplearning-model-thinks-tesla-cybertruck-is-a-refrigerator-elon-musk/>

# References and Credits

---

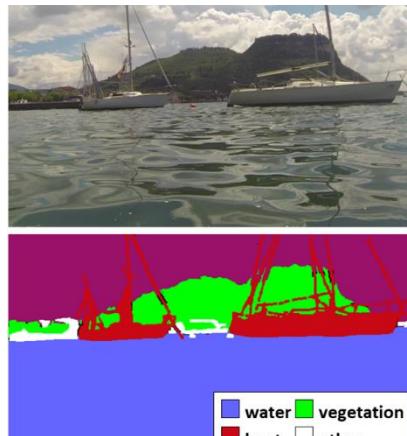
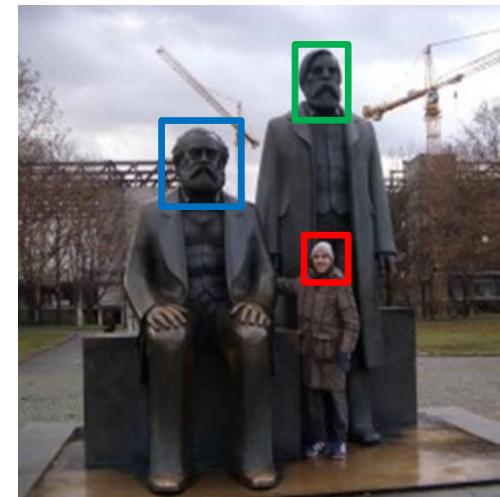
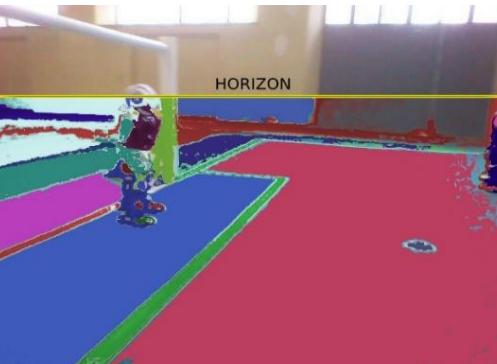
- P. Sermanet, “Object Detection with Deep Learning”
- K.H. Wong. “Ch. 6: Face detection”
- P. Viola and T.-W. Yue. “Adaboost for Face Detection”
- D. Miller. “Face Detection & Synthesis using 3D Models & OpenCV”
- S. Lazebnik. “Face detection”
- C. Schmid. “Category-level localization”
- C. Huang and F. Vahid. “Scalable Object Detection Accelerators on FPGAs Using Custom Design Space Exploration”
- P. Smyth. “Face Detection using the Viola-Jones Method”
- K. Palla and A. Kalaitzis. “Robust Real-time Face Detection”



UNIVERSITÀ DEGLI STUDI  
DELLA BASILICATA

*Corso di Visione e Percezione*

# Percezione Visione



Docente

Domenico D. Bloisi