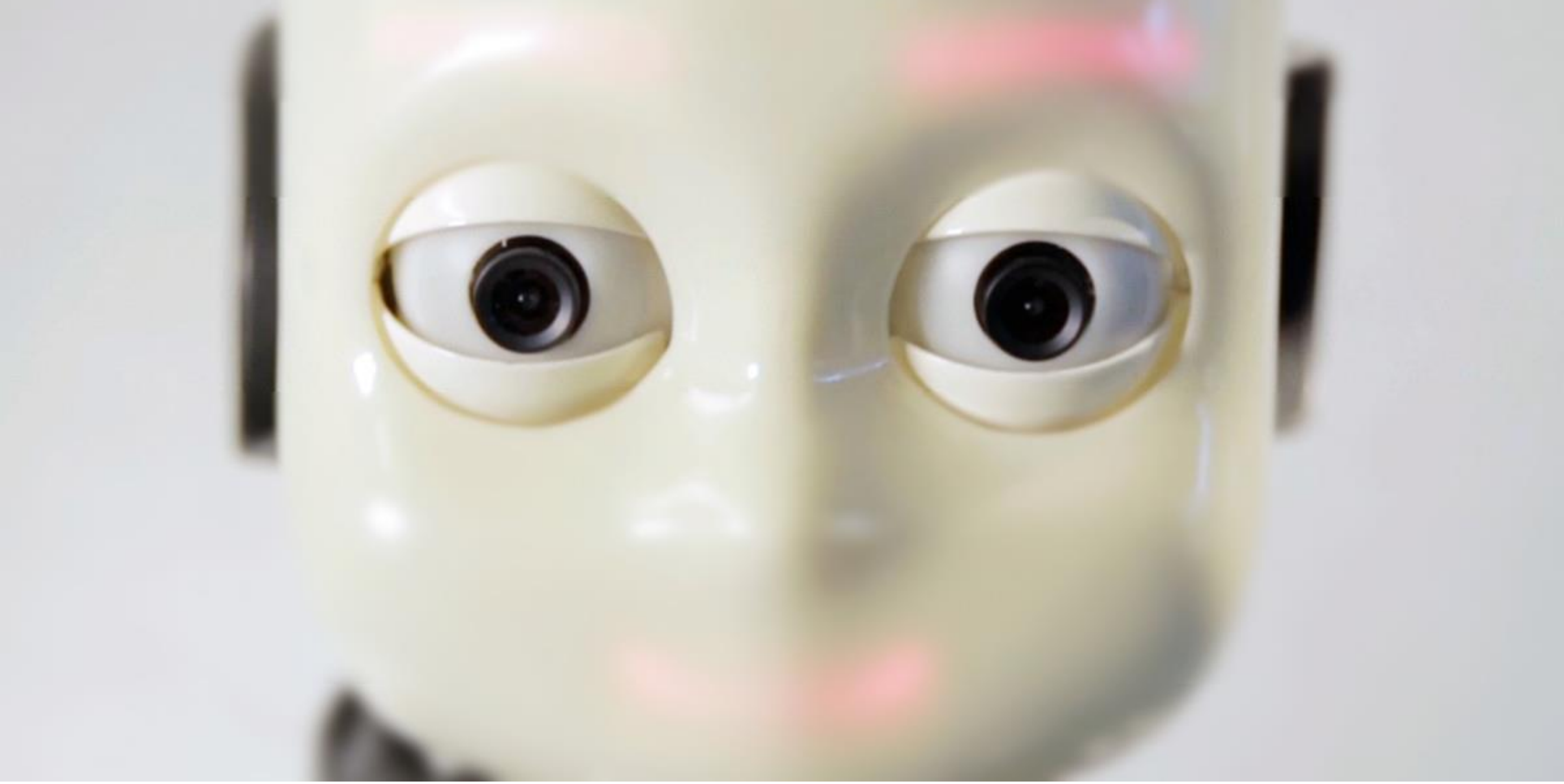


iCUB

a shared platform for research in robotics & AI

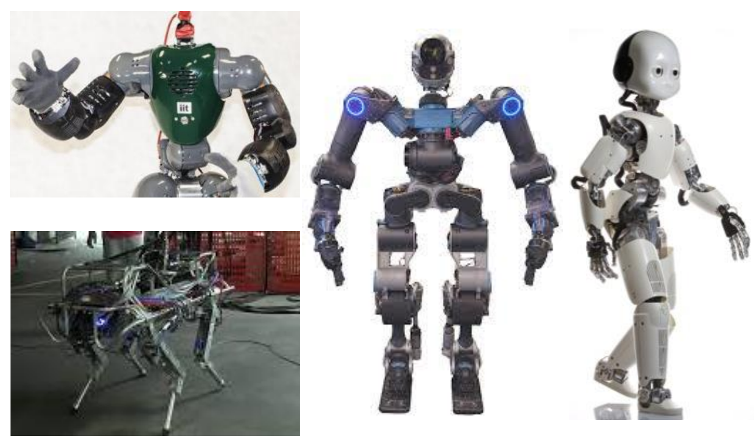
Alessandro Roncone
Postdoctoral Associate
Social Robotics Lab
http://alecive.github.io

IIT - Italian Institute of Technology

Italy

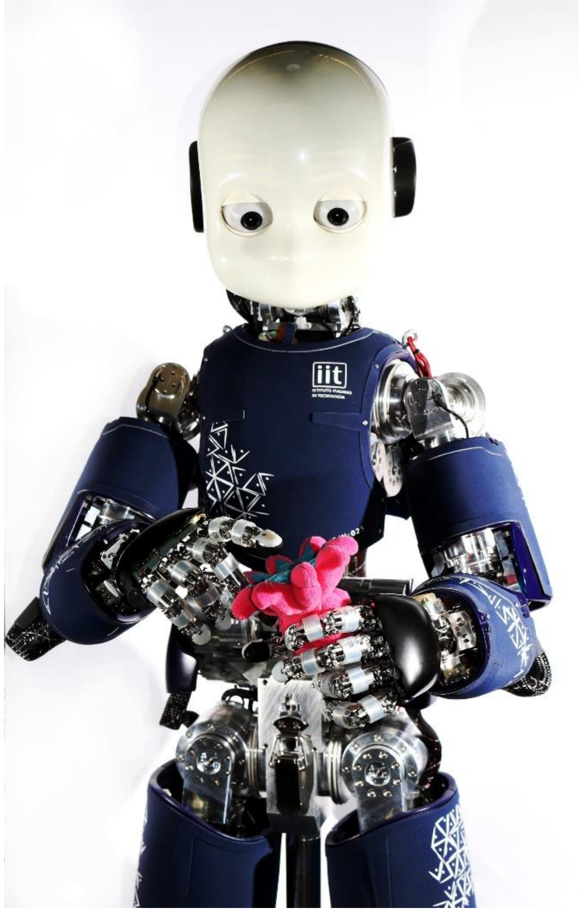
Genoa

IIT

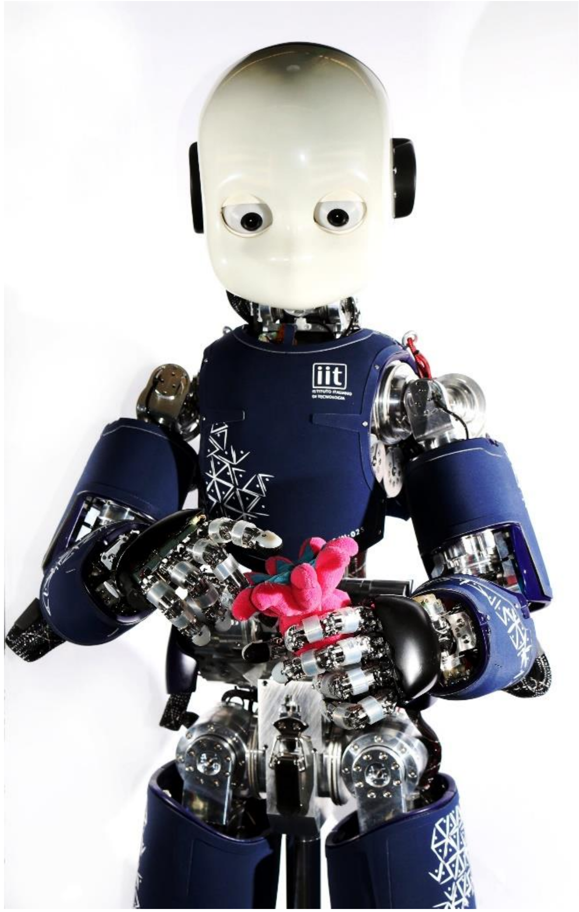


Robotics @ IIT

IIT - iCub Facility



The iCub



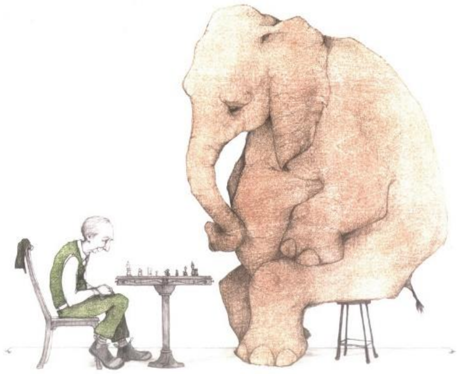
1. price: 250K€
2. born in 2004
3. 30 iCub distributed since 2008

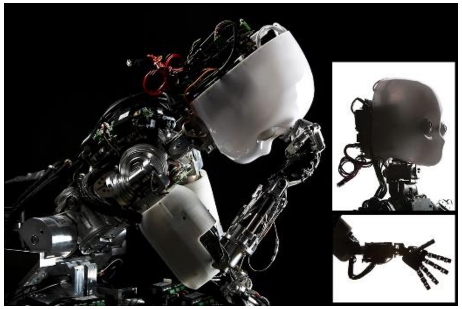
Why is the iCub so special?

- Full humanoid robot (104cm, 25 kg)
- 53 degrees of freedom (DoFs)
- Hands: 5 fingers, 9 degrees of freedom, 19 joints
- Human-like sensors: cameras, microphones, joint encoders, IMUs (accelerometer/gyroscope), force/torque sensors
- Artificial skin
- Large software repository (~2M lines of code)
- Open source HW & SW

Why humanoids?

- Scientific reasons (elephants don't play chess)
- Natural human-robot interaction
- Challenging mechatronics





Why open source?

- Repeatable experiments
- Benchmarking
- Quality of the HW & SW
- This resonates with industry-grade R&D in robotics

Why open source?

Outline

Hardware

Software

Force/Torque Sensor

YARP

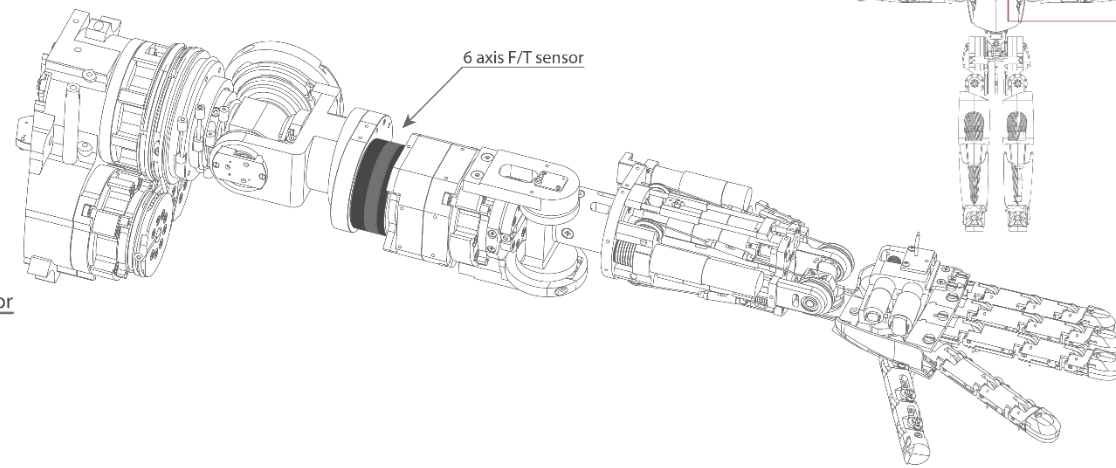
Artificial Skin

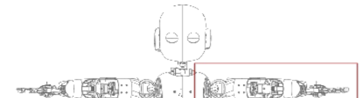
Kinematics/Dynamics

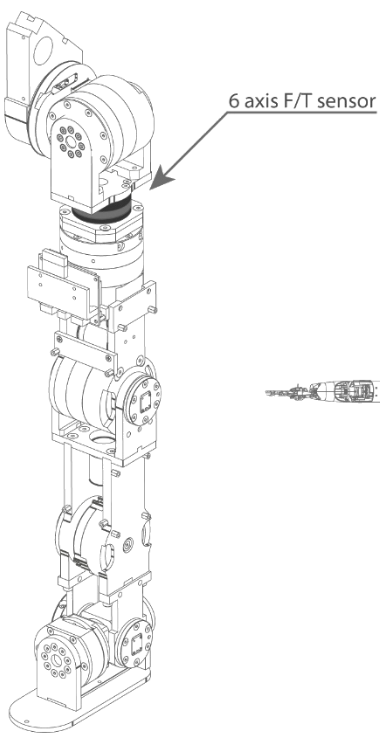
Inertial Sensor

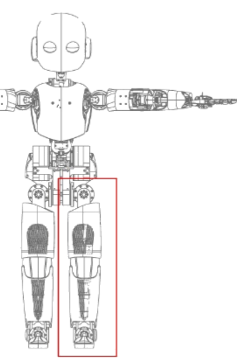
Computer Vision & Machine Learning

HW1 - Force / Torque sensors









- placed on the proximal part of the limb
- able to sense force up to the end-effector
- critical for many applications: safety, dynamics/control, HRI, ...

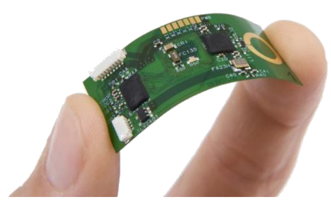
HW1 -Force / Torque sensors - Teaching Actions

HW2 - Artificial Skin

capacitor

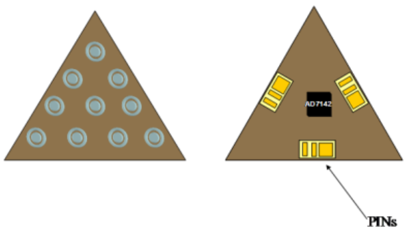
- ground plane: conductive fabric

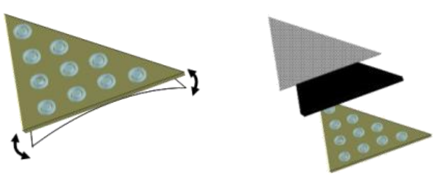
- soft material: e.g. silicone

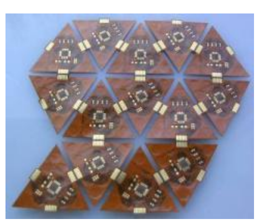


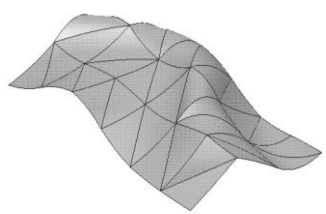
- electrodes: flexible PCB

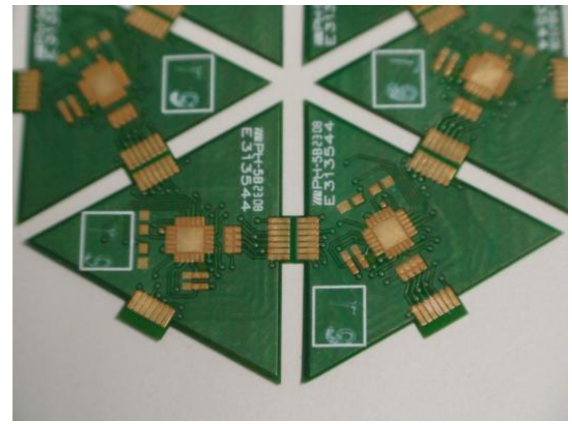
HW2 - Artificial Skin

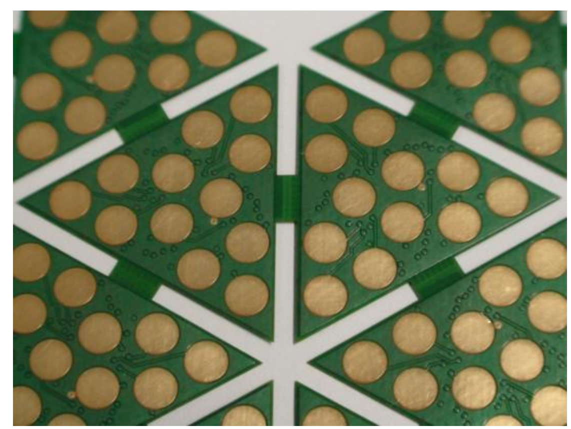




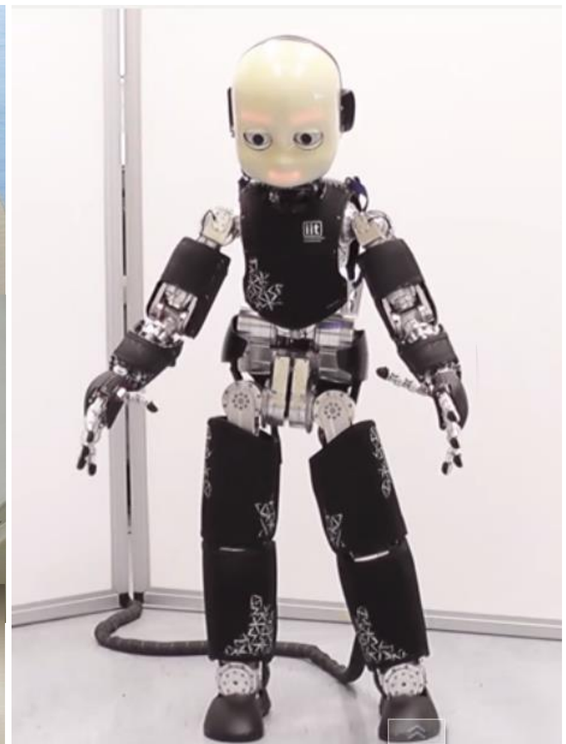








HW2 - Artificial Skin



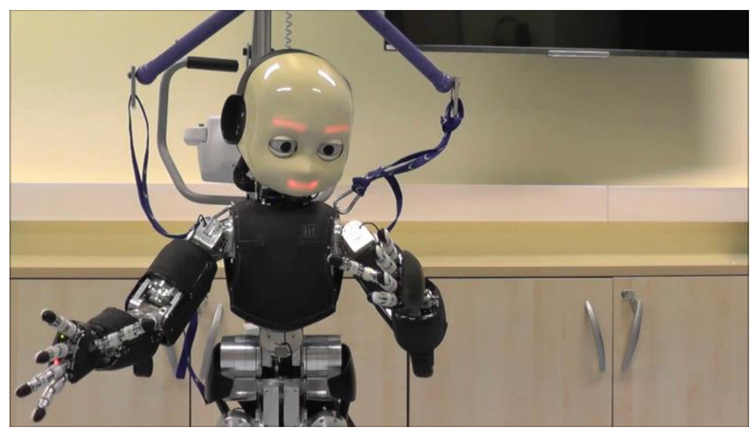
upper body: 1868

legs and feet: 1310x2

total: 4488 taxels!!

HW2 - Artificial Skin for grasping

Without tactile feedback



With tactile feedback

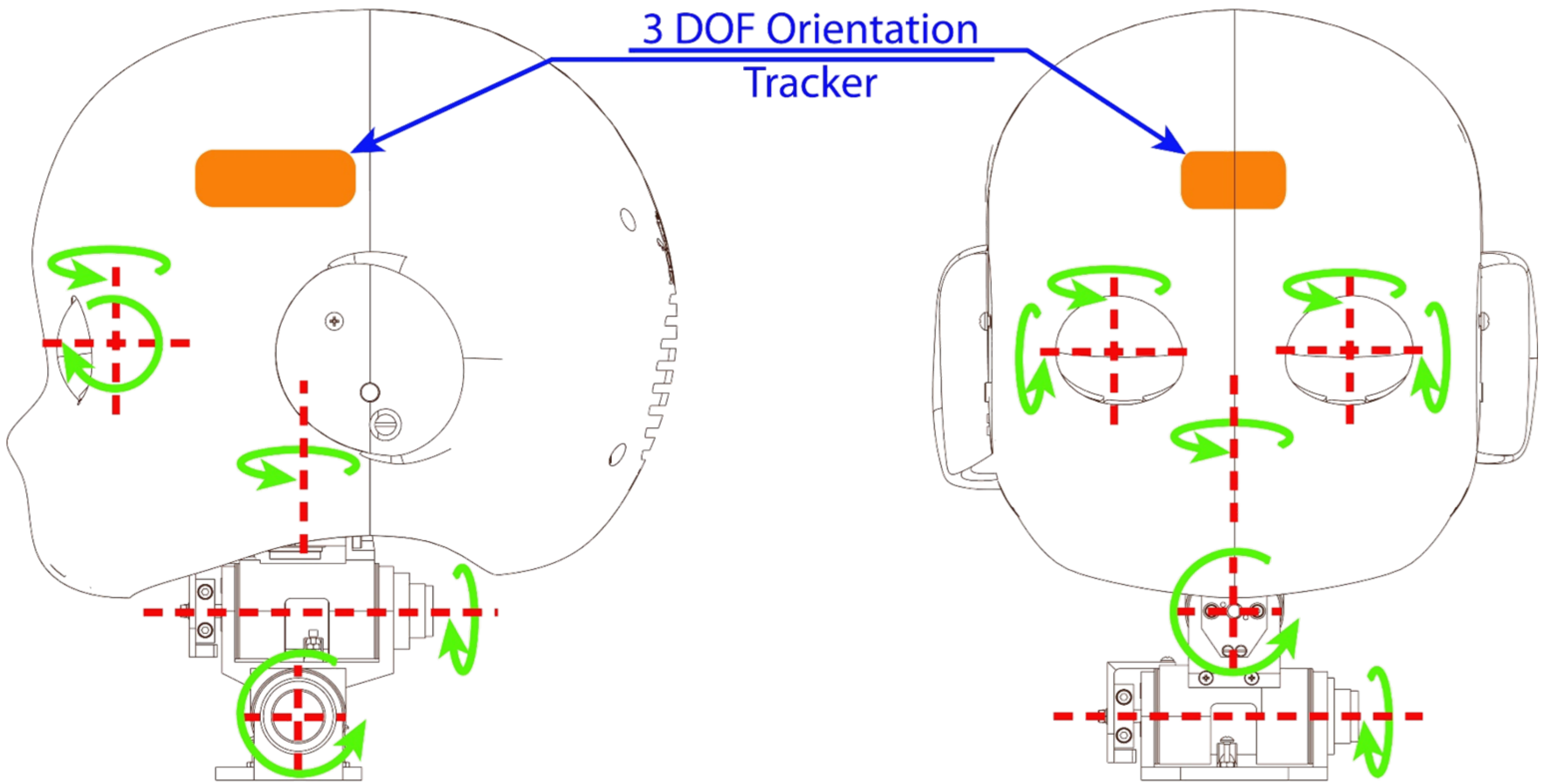
HW2 - Artificial Skin - Self Calibration

Automatic kinematic chain calibration using artificial skin: self-touch in the iCub humanoid robot

A. Roncone, M. Hoffmann,
U. Pattacini, and G. Metta

ISTITUTO ITALIANO DI TECNOLOGIA
iCUB FACILITY

HW3 - Inertial Sensor



HW3 - Inertial Sensor - Gaze Stabilization

Gaze Stabilization for Humanoid Robots: a Comprehensive Framework

A. Roncone, U. Pattacini,
G. Metta and L. Natale

iit iCub
ISTITUTO ITALIANO DI TECNOLOGIA
iCUB FACILITY

Outline

Hardware

Force/Torque Sensor

Artificial Skin

Inertial Sensor

Software

YARP

Kinematics & Dynamics

Computer Vision & Machine Learning

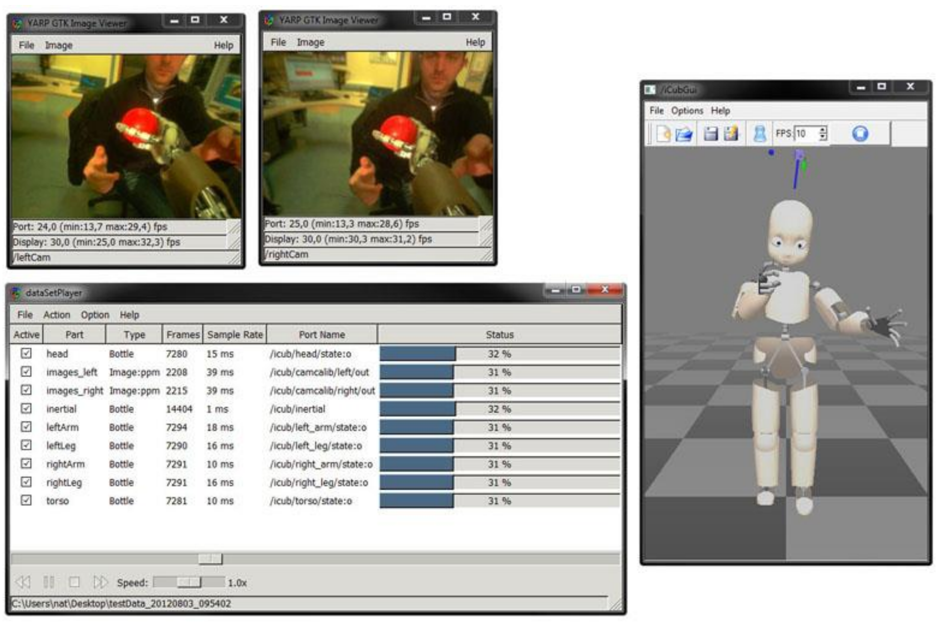
SW1 - YARP

YARP ⟶ Yet Another Robot Platform

- Peer-to-peer, loosely coupled, communication
- Very stable code base ~15 years old (older than ROS)
- Flexibility and minimal dependencies, fits well with other systems
- Easy install with binaries on many OSes/distributions (Ubuntu, Debian, Windows, MacOs)
- Many protocols:
    - Built-in: tcp/udp/mcast
    - Plug-ins: ROS tcp, xml rpc, mjpeg etc..

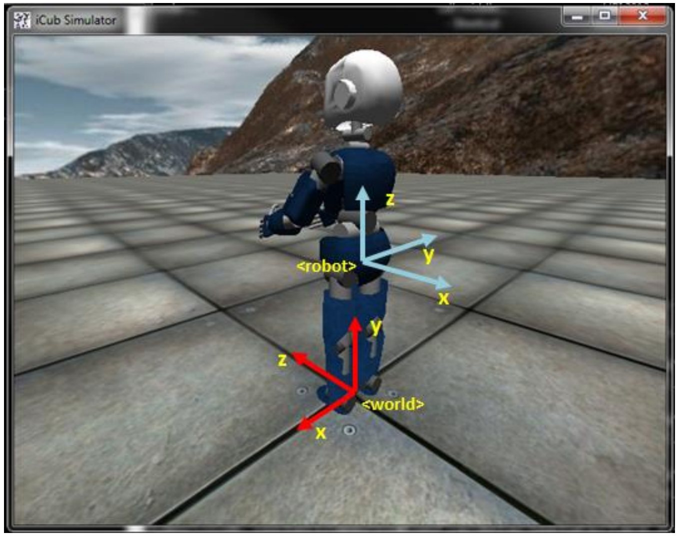
SW1 - YARP without hardware

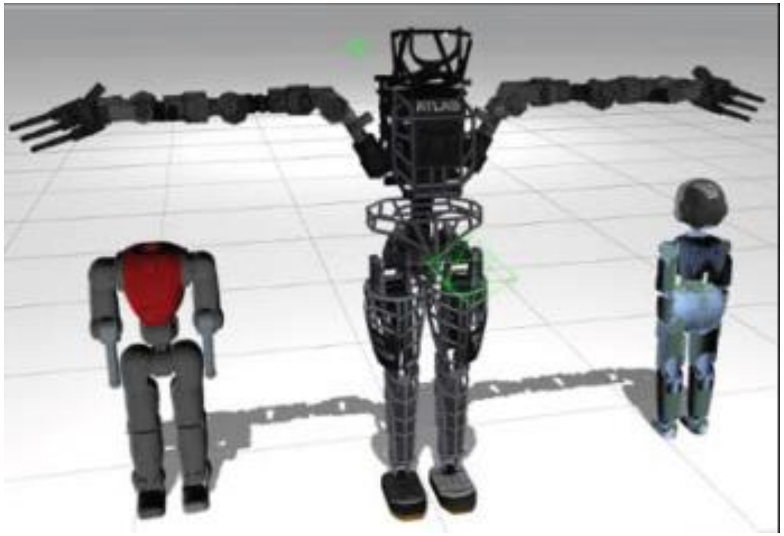
- Using YARP without hardware: dataset player
- Available in binary releases for Linux and Windows



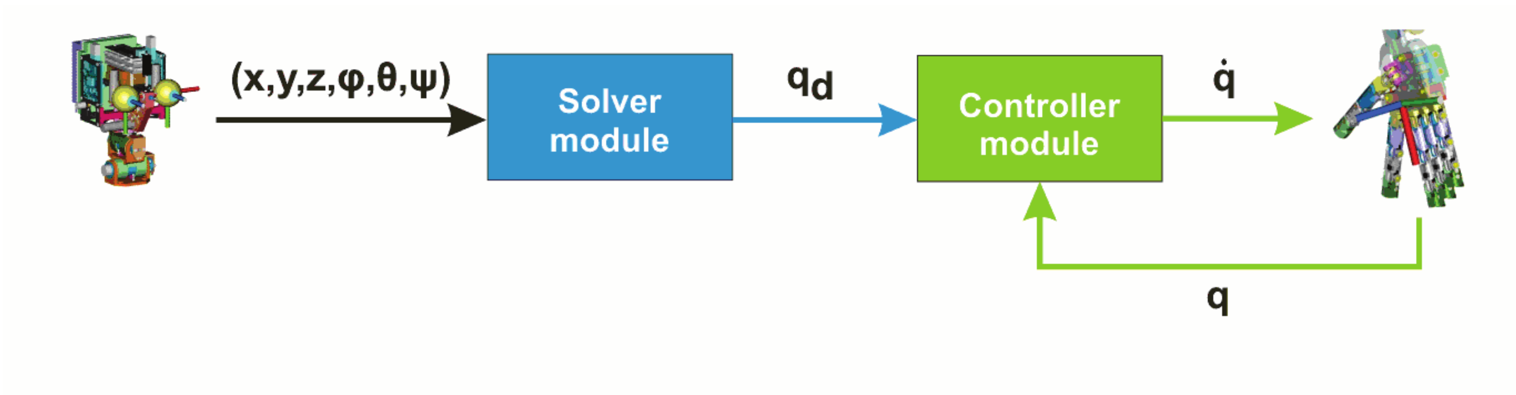
SW1 - YARP without hardware

- Using YARP without hardware: simulators
  - iCub_SIM, and ODE-based simulator
  - Gazebo, the VRC/DRC simulator





SW2 - Inverse Kinematics and Cartesian Control



- Inverse Kinematics Solver + Controller
- IK Solver → Non linear constrained optimization
- Controller → Able to generate smooth, human-like velocity profiles at the end-effector given the desired joint configuration

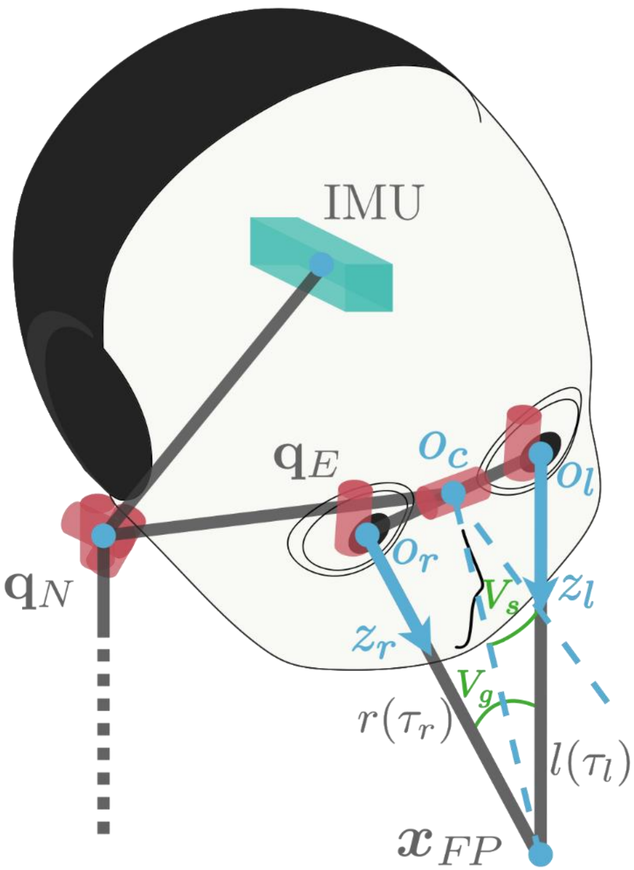
SW2 - Inverse Kinematics - IPOPT

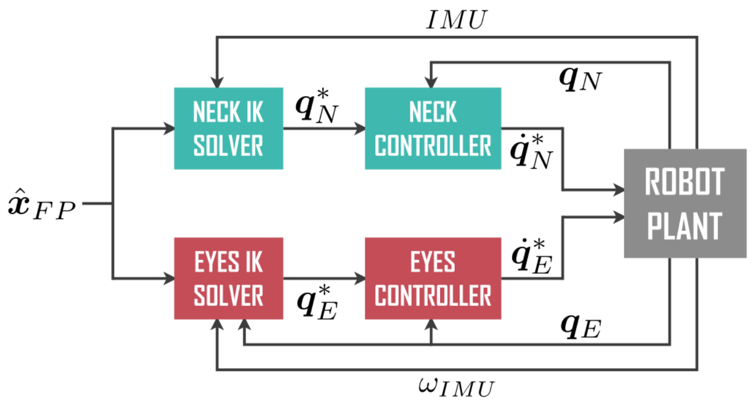
$$\begin{cases} q_d = argmin_{q \in R^n}(\|\alpha_d - K_\alpha(q)\|^2 + \beta(q_{res} - q)^T W(q_{res} - q)) \\ s.t. \begin{cases} \|x_d - K_x(q)\|^2 < \varepsilon \\ q_L < q < q_U \end{cases} \end{cases}$$

- Quick convergence (<10ms)
- Scalability
- Singularities and joints bound handling
- Complex constraints:

$$\begin{cases} \min_{\dot{q},\dot{x}} \frac{1}{2}\left((\dot{q} - \dot{q}_d)^T W_q (\dot{q} - \dot{q}_d) + (\dot{x} - \dot{x}_d)^T W_x (\dot{x} - \dot{x}_d)\right) \\ s.t. \;\; \dot{x} = J \cdot \dot{q} \end{cases}$$

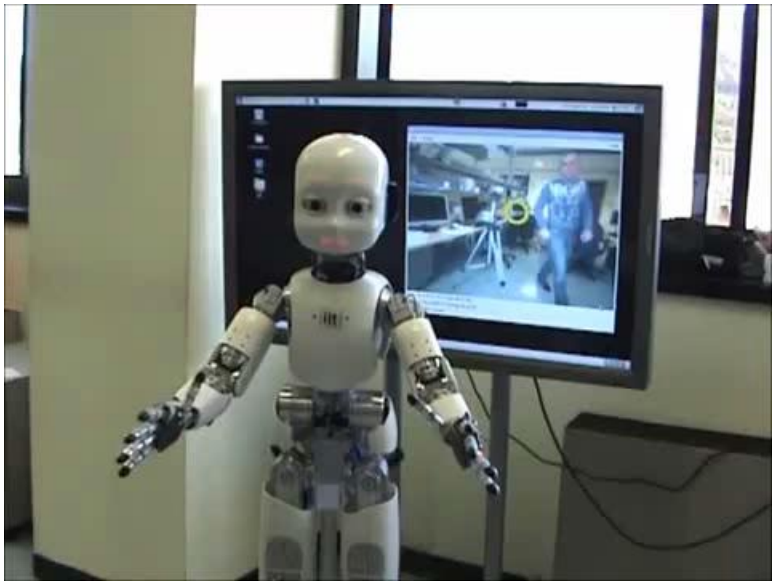
SW2 - Inverse Kinematics and Gaze Control





- The iCub's head has 6DoF
- The fixation point can be seen as the end-effector of a virtual kinematic chain that starts from the neck base
- Similar techniques apply

SW2 - Coordinated Cartesian and Gaze Control



- The red ball is detected thanks to a particle filter tracker
- The tracker provides the 3D position of the ball w.r.t. the robot

- The Cartesian controller steers the arm toward the target 3D point
- The Gaze controller moves the robot's gaze in the same direction
- The Force/Torque sensors make the robot compliant

SW2 - Dynamics

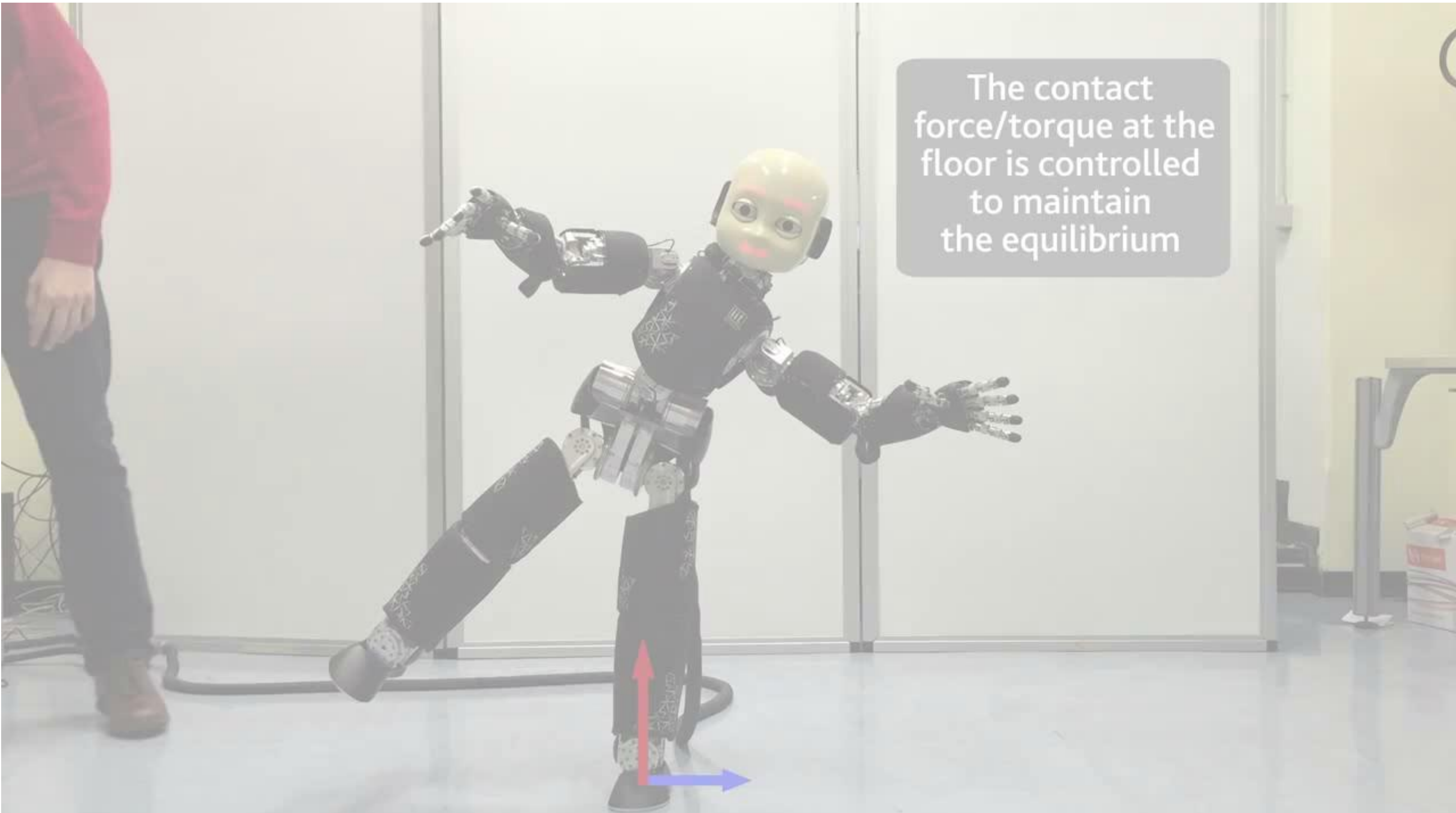
SW2 - Dynamics is (theoretically) solvable..

$$\mathrm{M}(\mathrm{q})\dot{\mathrm{v}}_{\mathrm{q}} + \mathrm{C}(\mathrm{q}, \mathrm{v}_{\mathrm{q}})\mathrm{v}_{\mathrm{q}} + \mathrm{g}(\mathrm{q}) = \begin{bmatrix} 0 \\ \tau_{\mathrm{q}} \end{bmatrix} + \mathrm{J}^{\top}(\mathrm{q})\mathrm{f}$$

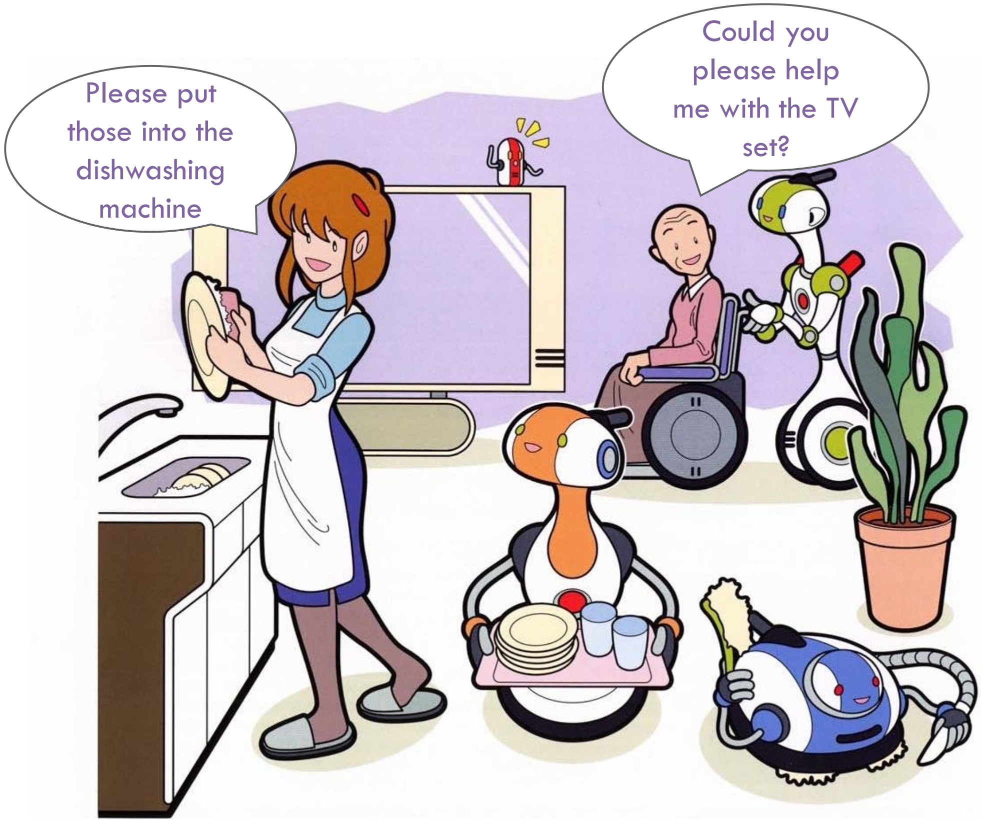
$$\mathbb{M}(\mathbb{q})\dot{\mathbb{v}}_{\mathbb{q}} + \mathbb{C}(\mathbb{q}, \mathbb{v}_{\mathbb{q}})\mathbb{v}_{\mathbb{q}} + \mathbb{g}(\mathbb{q}) = \begin{bmatrix} 0 \\ \tau_{\mathbb{q}} \end{bmatrix} + \mathbb{J}^{\top}(\mathbb{q})\mathrm{f}$$

$$\begin{bmatrix} \mathrm{J}(\mathrm{q}) & \mathbb{J}(\mathbb{q}) \end{bmatrix} \begin{bmatrix} \dot{\mathrm{v}}_{\mathrm{q}} \\ \dot{\mathbb{v}}_{\mathbb{q}} \end{bmatrix} + \dot{\mathrm{J}}(\mathrm{q}, \mathrm{v}_{\mathrm{q}}) + \dot{\mathbb{J}}(\mathbb{q}, \mathbb{v}_{\mathbb{q}}) = 0$$

..but hard to implement!



SW3 - Computer Vision & Machine Learning



SW3 - Computer Vision & Machine Learning

The iCub puts the plates into the dishwashing machine

actions objects tools

SW3 - Computer Vision & Machine Learning

Actions | Objects | Tools

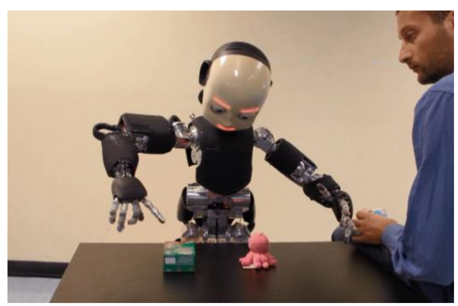
learning actions

learning objects

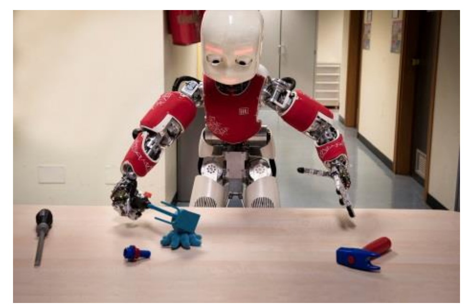
learning tools

Learn

recognizing actions



recognizing objects

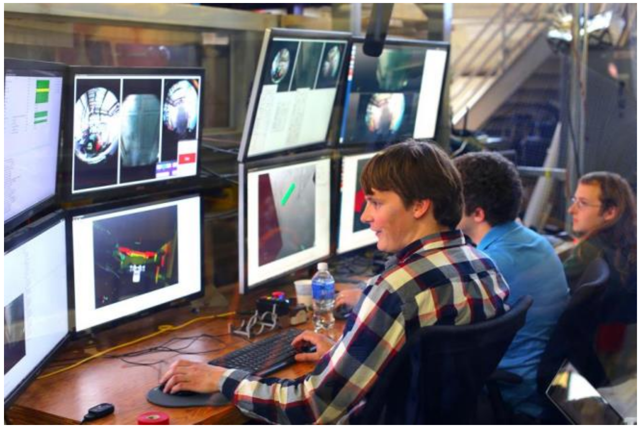


using tools

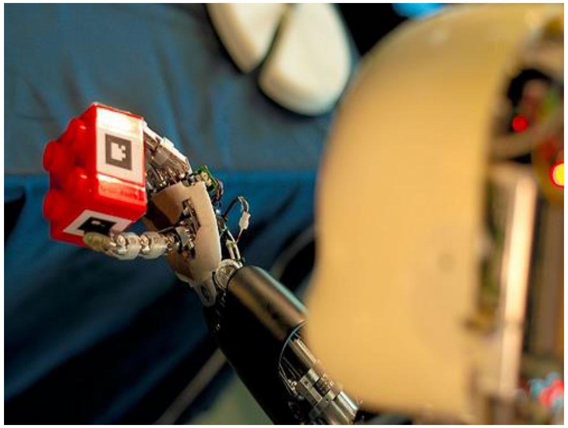
Use

SW3 - Computer Vision for Robotics

Teleoperation



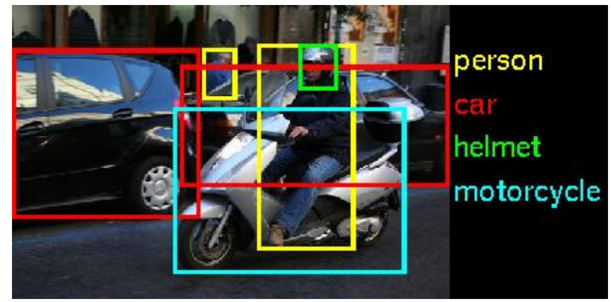
Markers



Structured Environment

3D reconstruction & strong supervision

SW3 - Breakthrough in Computer Vision



Deep Learning

+

Big Datasets

=

Approaching human performance on the same dataset!

There are better ways to do that!

Human-Robot Interaction

Self-Supervision

HRI is a natural application for visual recognition

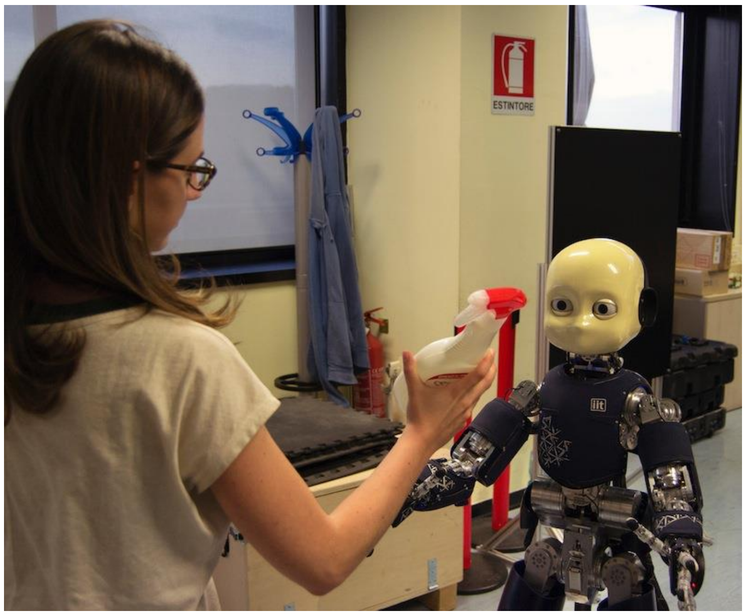
In robotics strong cues are often available, therefore object detectors can be avoided

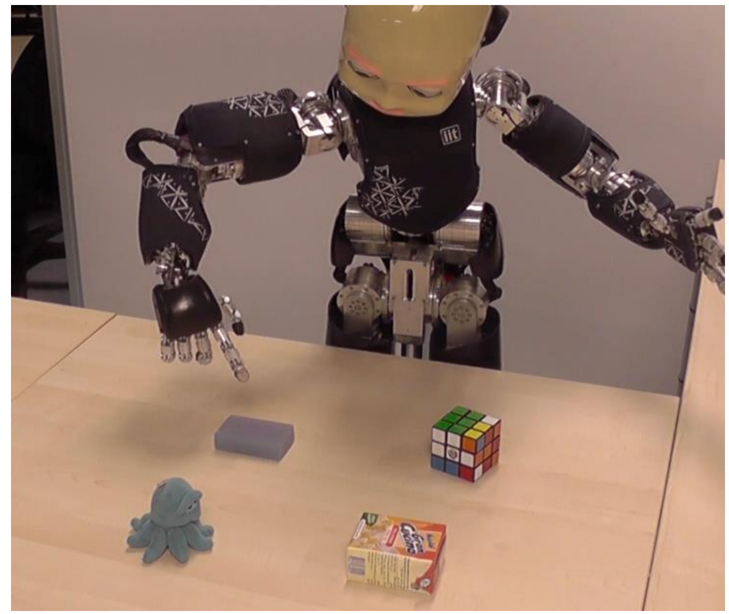
Recognition as tool for complex tasks: grasp, manipulation, affordances, pose

kinematics

motion

Semi-autonomous Learning

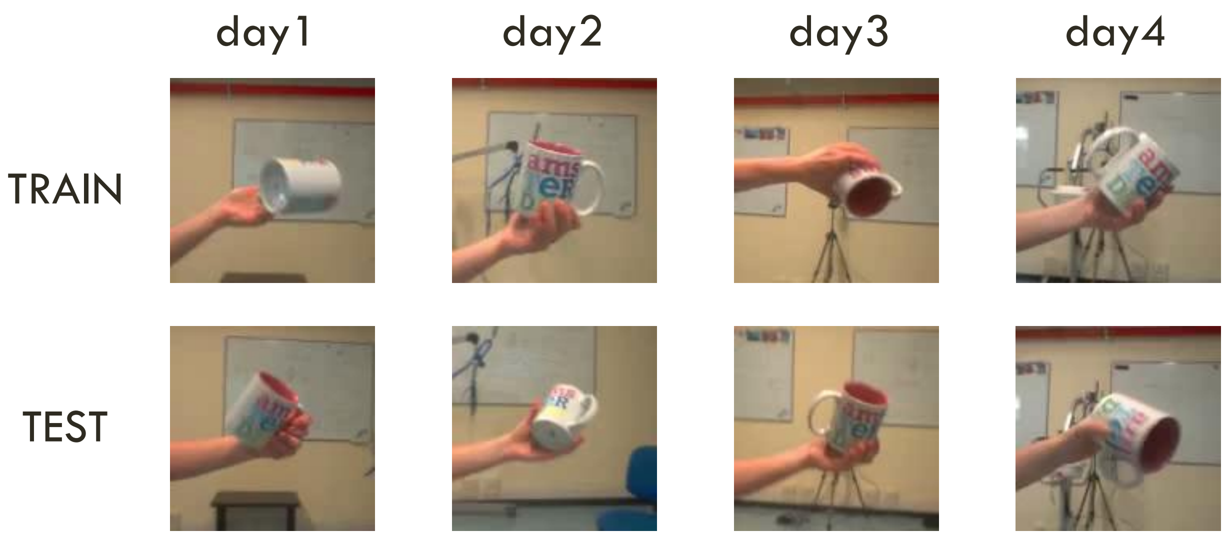




iCub World 2.0 Dataset

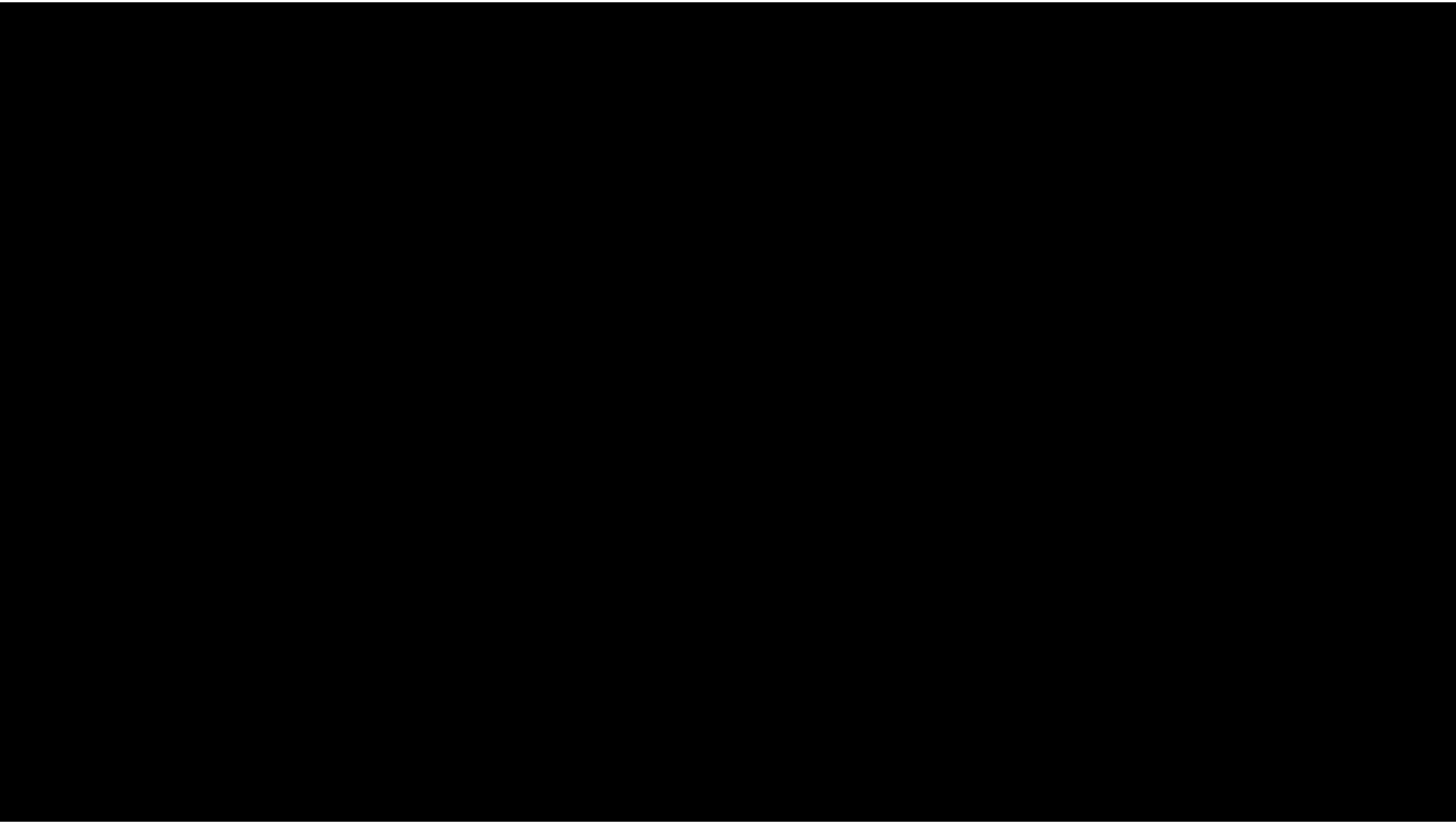
- Growing dataset collecting images from a real robotic setting
- Tool for benchmarking visual recognition systems in robotics
- 28 Objects, 7 categories, 4 different acquisition sessions → ~50K Images
- http://www.iit.it/en/projects/data-sets.html

iCub World 2.0 Dataset



- Growing dataset collecting images from a real robotic setting
- Tool for benchmarking visual recognition systems in robotics
- 28 Objects, 7 categories, 4 different acquisition sessions → ~50K Images
- http://www.iit.it/en/projects/data-sets.html

Interactive Objects Learning



Thank you! And thanks to:

Giorgio Metta

Lorenzo Natale, Francesco Nori

Ugo Pattacini, Vadim Tikhanoff, Marco Randazzo, Carlo Ciliberto, Daniele Pucci, Francesco Romano, Giulia Pasquale, Sean Ryan Fanello, Ali Paikan, Jorhabib Eljaik, Silvio Traversaro

The iCub Facility