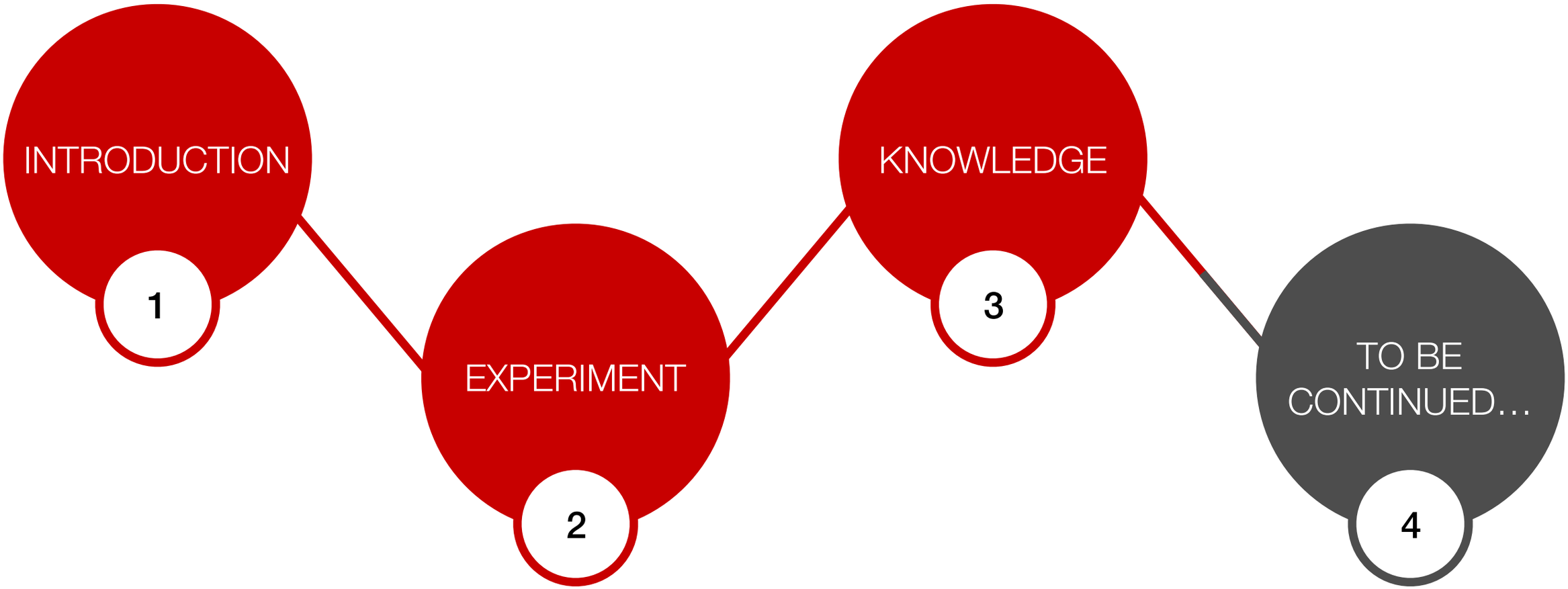
Institut de Robòtica
i Informàtica Industrial

SCRAT

Shared Control Robotics for Assistive Tasks

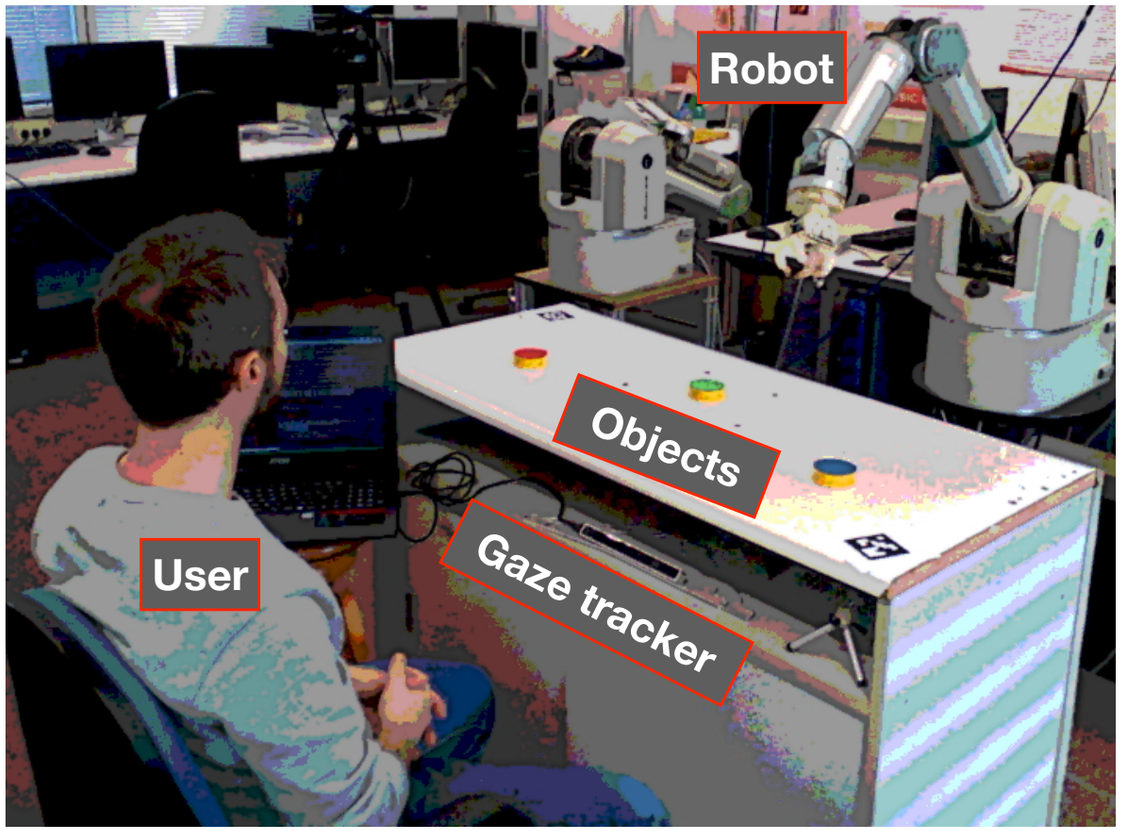
Johannes Heidecke - Eunice Njeri - Alberto Olivares Alarcos - Alejandro Suárez Hernández



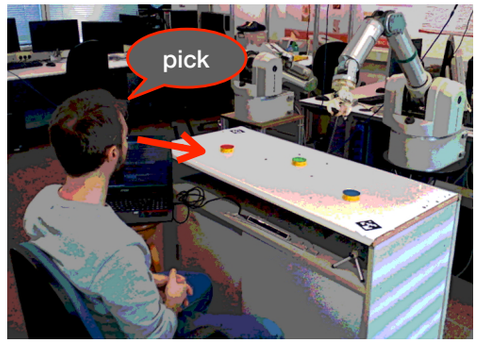
OUTLINE

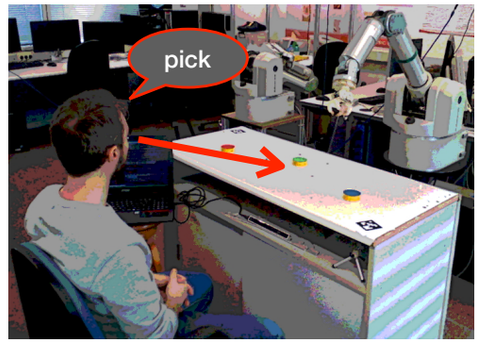
INTRODUCTION

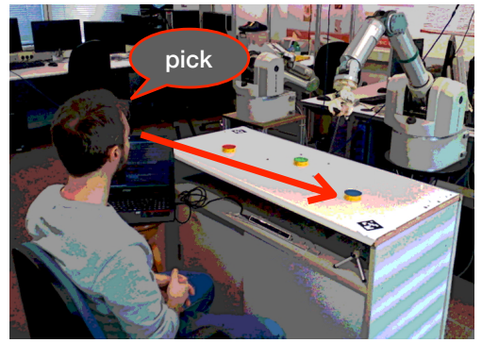
ENVIRONMENTAL SETUP



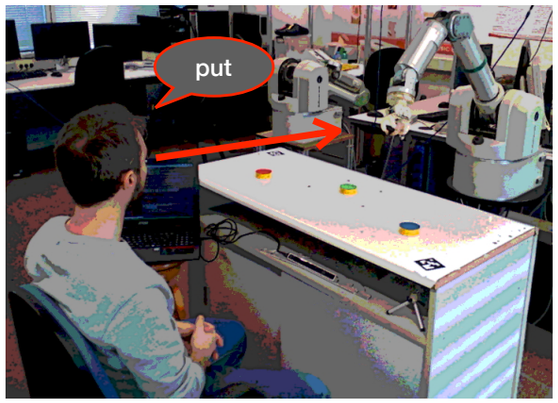
STEP 1 — PICKING AN OBJECT

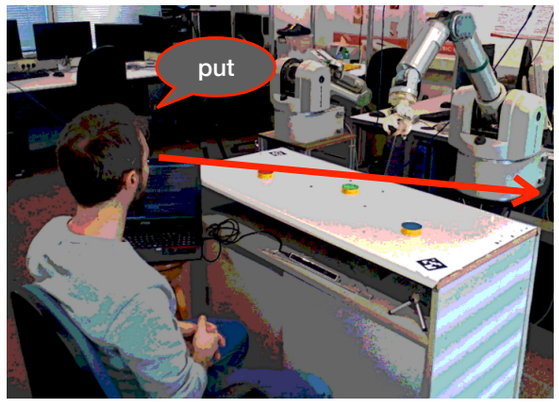




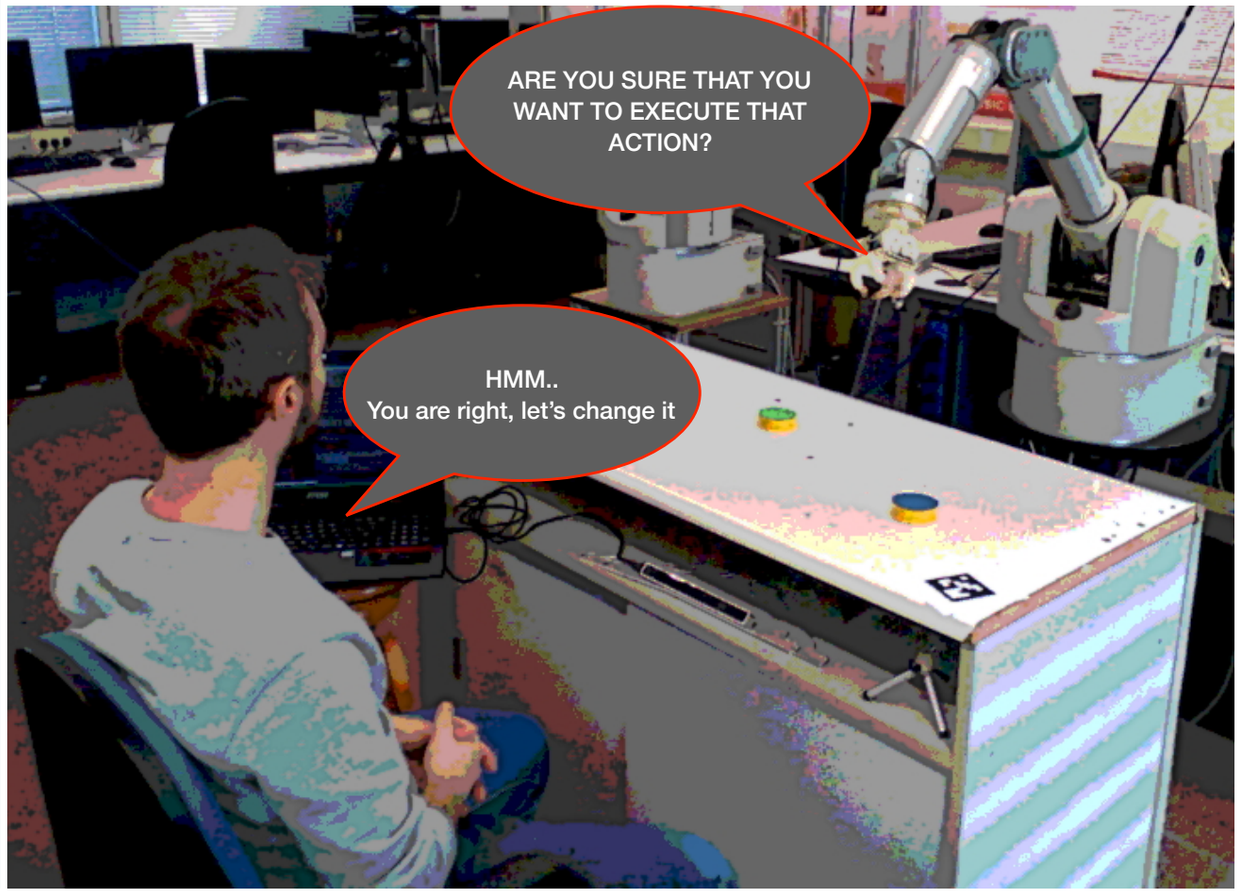


STEP 2 — PUTTING AN OBJECT



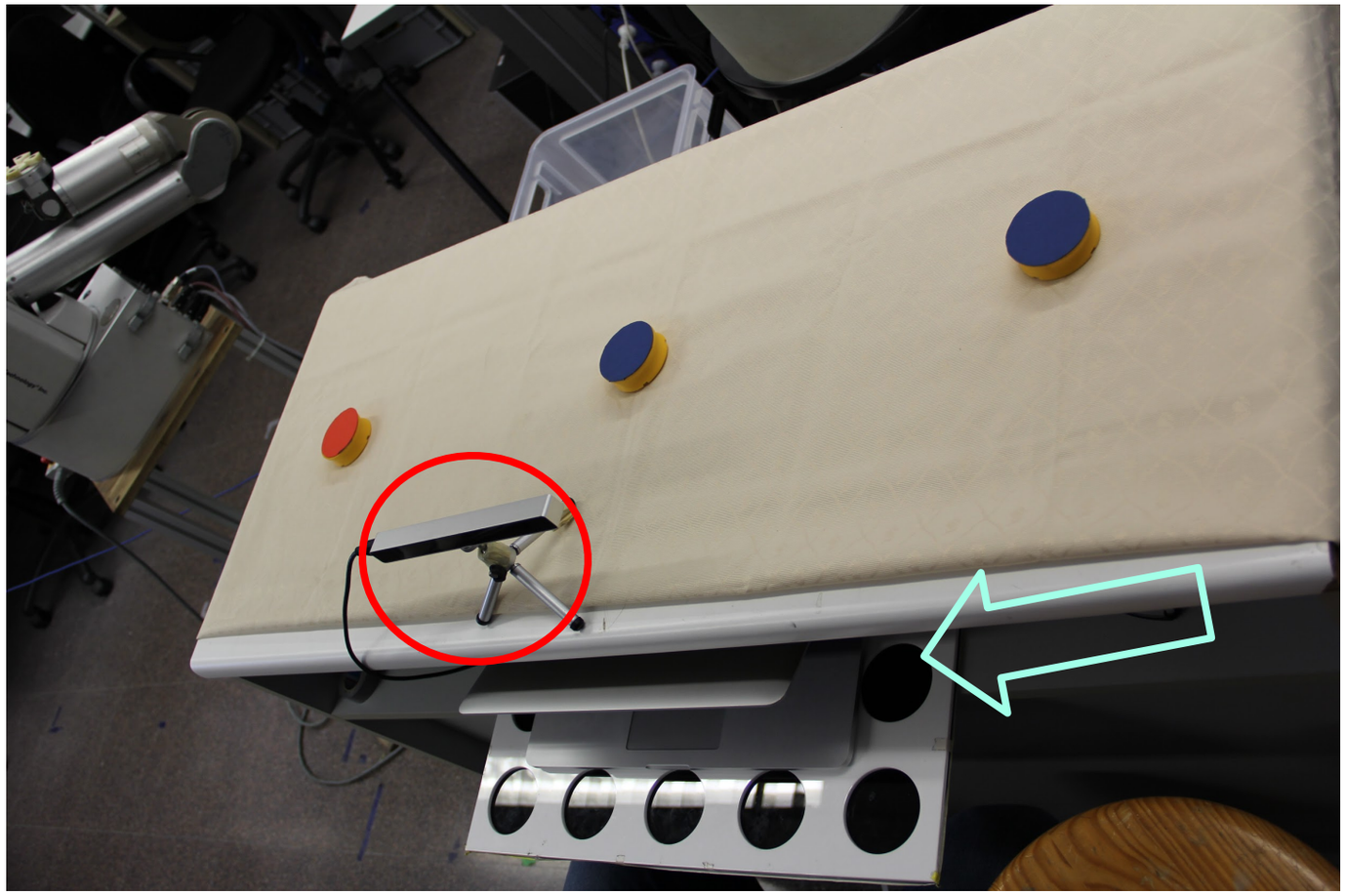


SHARED CONTROL

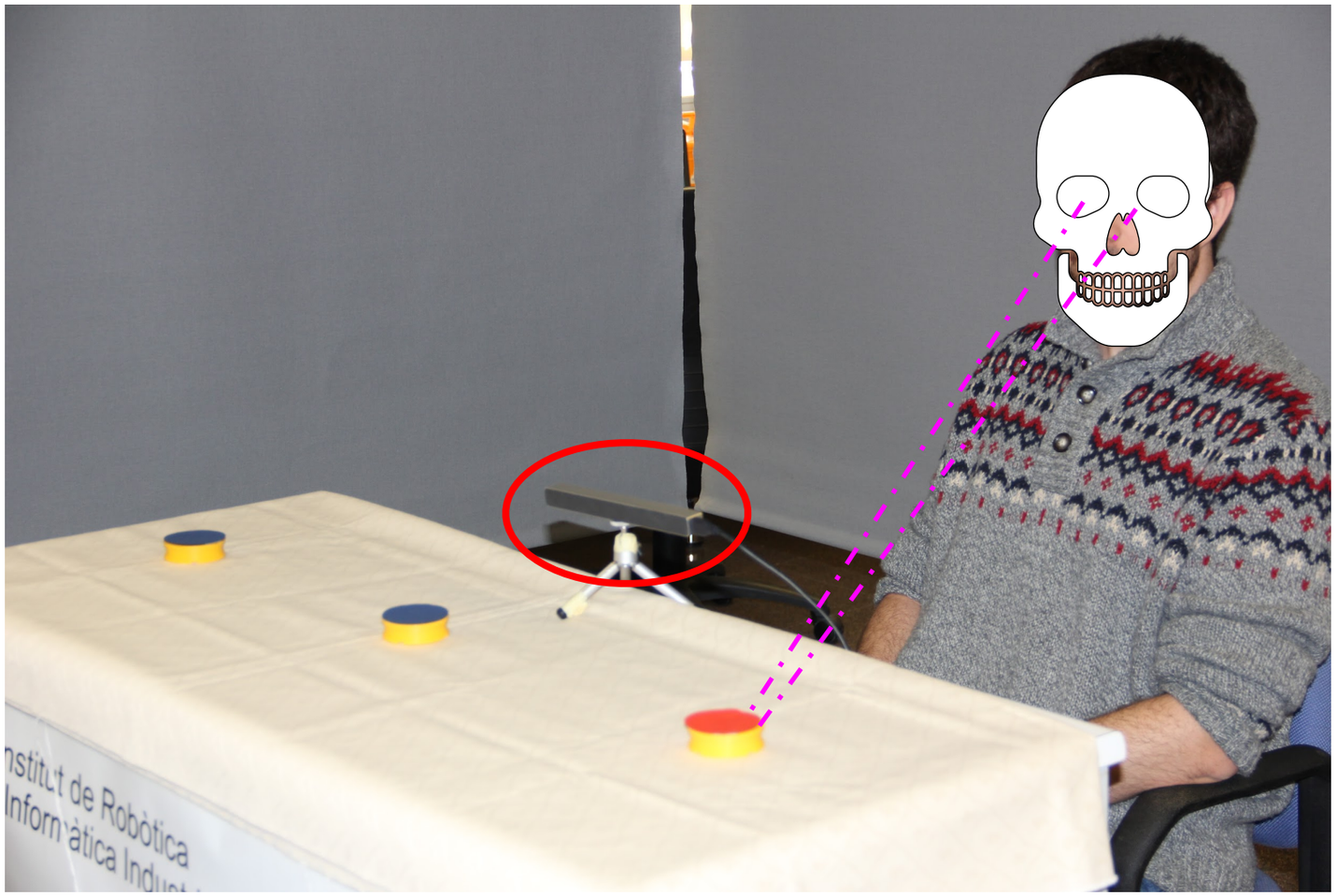


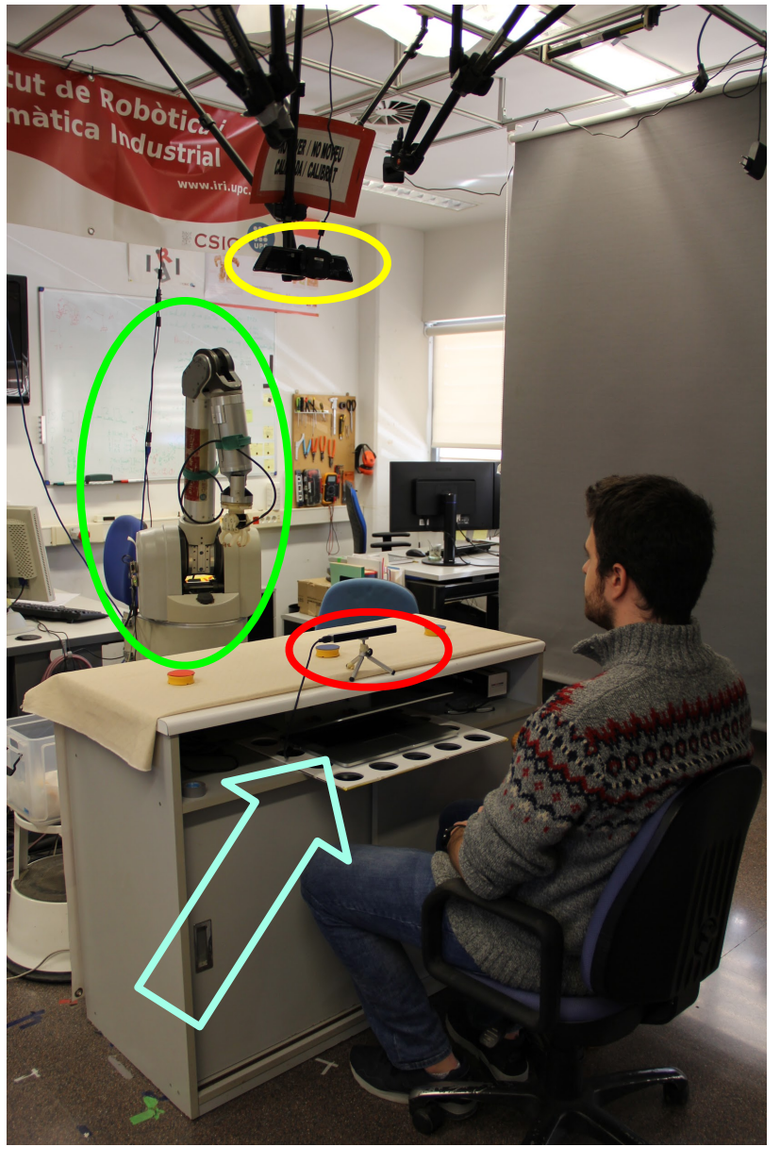
Storyboard

EXPERIMENT

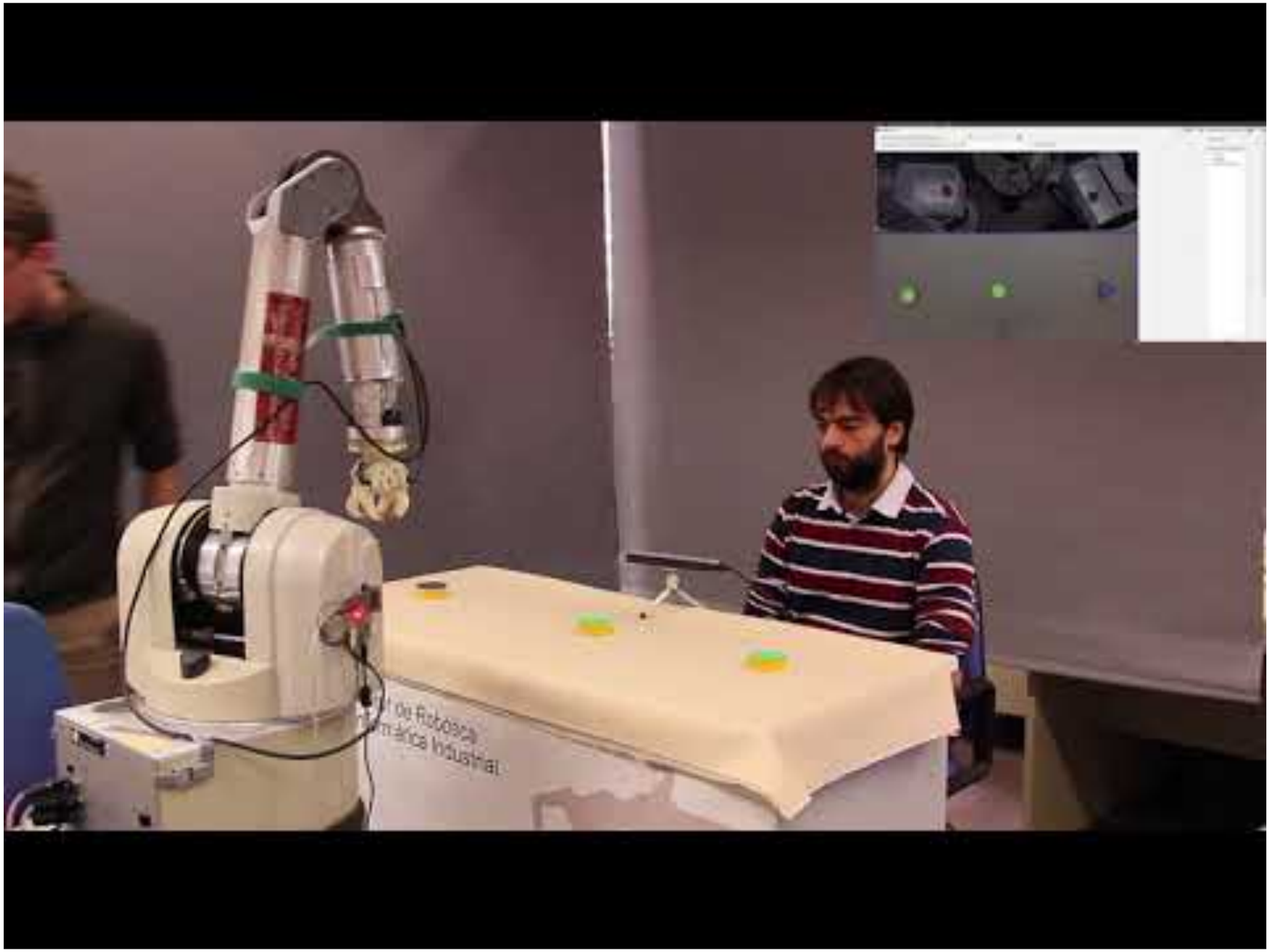


Experimental Setup I





Experimental Setup II



KNOWLEDGE

Pupil's position

All data generated by tracker

Gaze SDK Server (TCP)

All data generated by tracker

ROS Node User server

Action (pick or put)

Action ∈ {pick[L,M,R], put[L,R]}

RGB image + Point Cloud

ROS Node Segmenter

(x,y,z) coords + color

Topic /.../pieces

ROS Node Environment server

Action ∈ {pick[L,M,R], put[L,R], warn}

IRL algorithm

State = (left, mid, right, arm, last_l, last_r)

Pick&place actions

Warning

Architecture

General Purpose Concepts

- User/Operator (instance OR subclass of agent?)
- User (human) related concepts (e.g. eye, gaze, etc.)
- Warning
- Command/order
- Collision
- Trajectory
- RGB Image
- Point Cloud
- Environment
- Segmentation (for images)

Devices (Sensors & Actuators)

- Micro (sound sensor)
- Kinect, gaze tracker (multispectral light sensor)
- Motor (motion actuator)
- Speaker (sound actuator)

Objects

- Furniture (e.g. table, chair, etc.)
- Piece (objects to pick, in this case, they are simple)

Actions

- Picking & Grasping
- Putting & Dropping
- Communicating (note that it involves talking and listening)
- Planning (trajectories)
- Planning (actions)
- Perception
- Looking at (this includes the case in which the human is looking at the object)
- Object detection (segmenting the image)
- *Parameters - Preconditions - Efects*

Spatial Notions

- Left, right, mid, etc.
- Coordinates (e.g. cartesian)
- Position and Orientation

Robot parts

- Link and Joint
- Robotic Arm
- End effector (e.g. gripper)

Programming

- Code (maybe not necessary)
- Algorithm
- Method/Function
- ROS and its keywords (node, topic, service, etc)

Others

- Notion of Property (e.g. rigid, for the objects or the robot's links)

Knowledge

Example of picking an object (written in Common Lisp from CRAM (KnowRob))

Example

```
(let* ((obj-pose (find-object obj))
       (pre-grasp-pos (calculate-pre-grasp obj-pose))
       (grasp-vector (cl-transforms:make-3d-vector 0 0 -0.1))
       (lift-vector (cl-transforms:make-3d-vector 0 0 0.1)))
  (open-gripper side)
  (take-collision-map)
  (with-failure-handling
      ((no-ik-solution (e)
         (move-to-different-place)
         (retry))
       (link-in-collision (e)
         (setf pre-grasp-pos (new-pre-grasp))
         (retry))
       (trajectory-controller-failed (e)
         (retry)))
    (move-arm-to-point side pre-grasp-pos)))
    ...)
```

Q&A TIME

Q

Paulo: Question on the usage of ROS action server (from actionlib)

A

Alberto: No.

Q

Alaa: Did you take ‘the temporal notion between actions’ into account? What about using the properties of the objects (e.g. rigid pieces, rigid robot’s links, etc.)

A

Alberto: We did not consider the time, users were supposed to not give new commands until the action was finished. That is a good idea, we should also include the concept of ‘property’ (see, now is added).

Q

Veera: Which is the bottle neck? That should take into account in order to see how the future reasoning process would affect.

A

Alberto: Probably the perception of the pieces. But we did not measured.

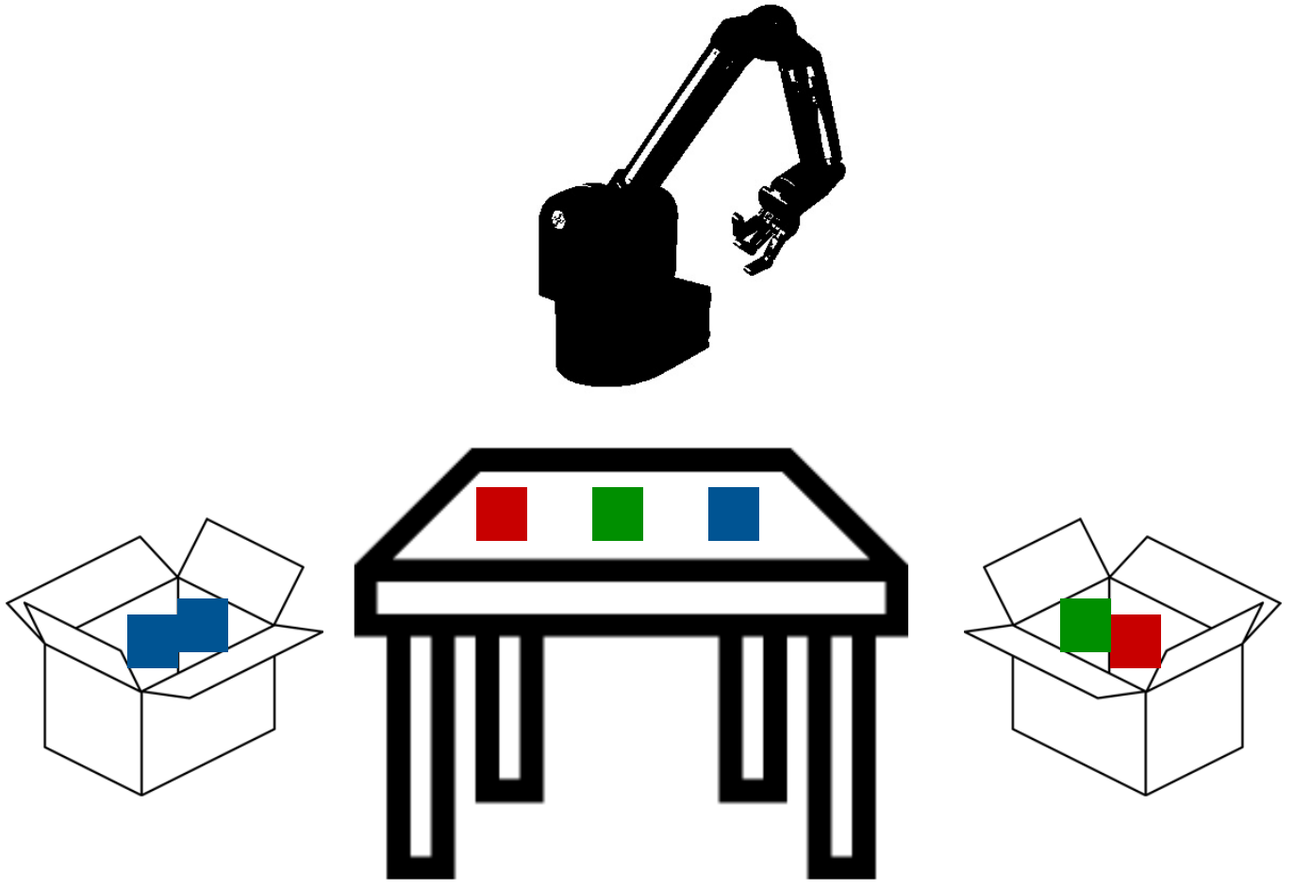
Institut de Robòtica
i Informàtica Industrial

thanks for your attention!

SCRAT

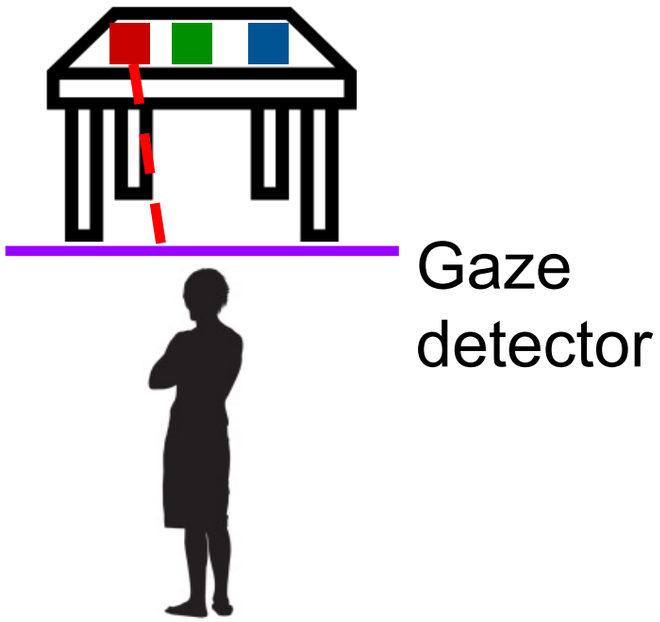
Shared Control Robotics for Assistive Tasks

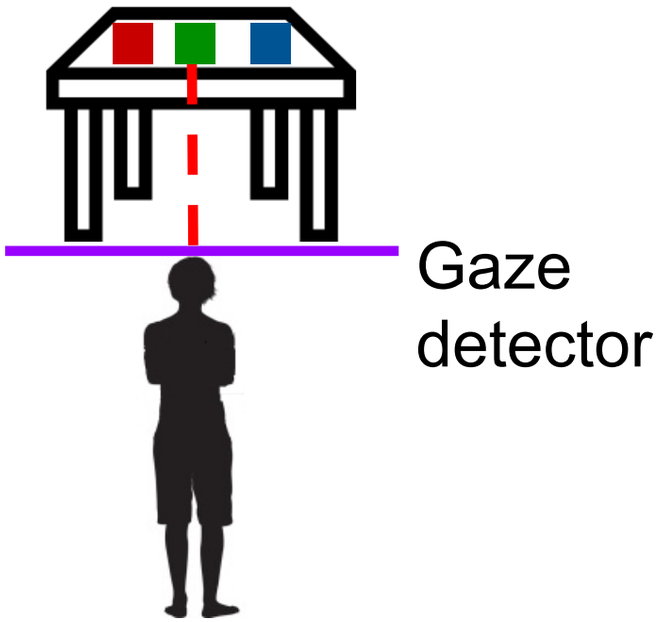
Johannes Heidecke - Eunice Njeri - Alberto Olivares Alarcos - Alejandro Suárez Hernández

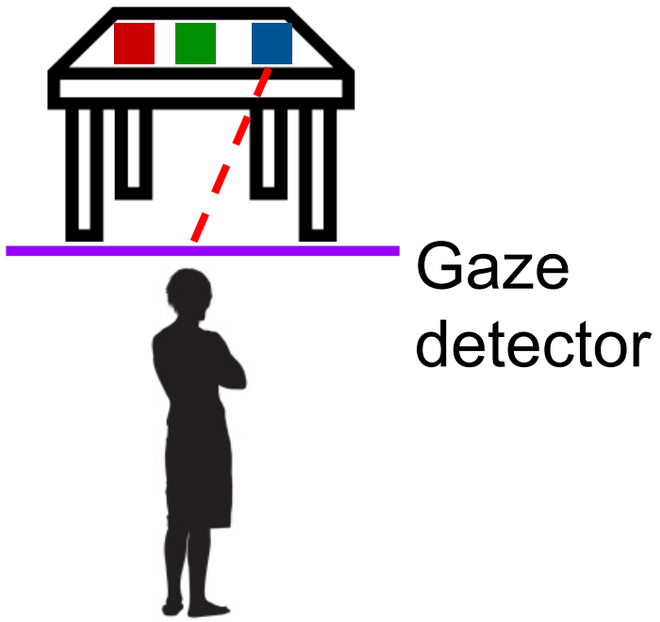


Gaze detector

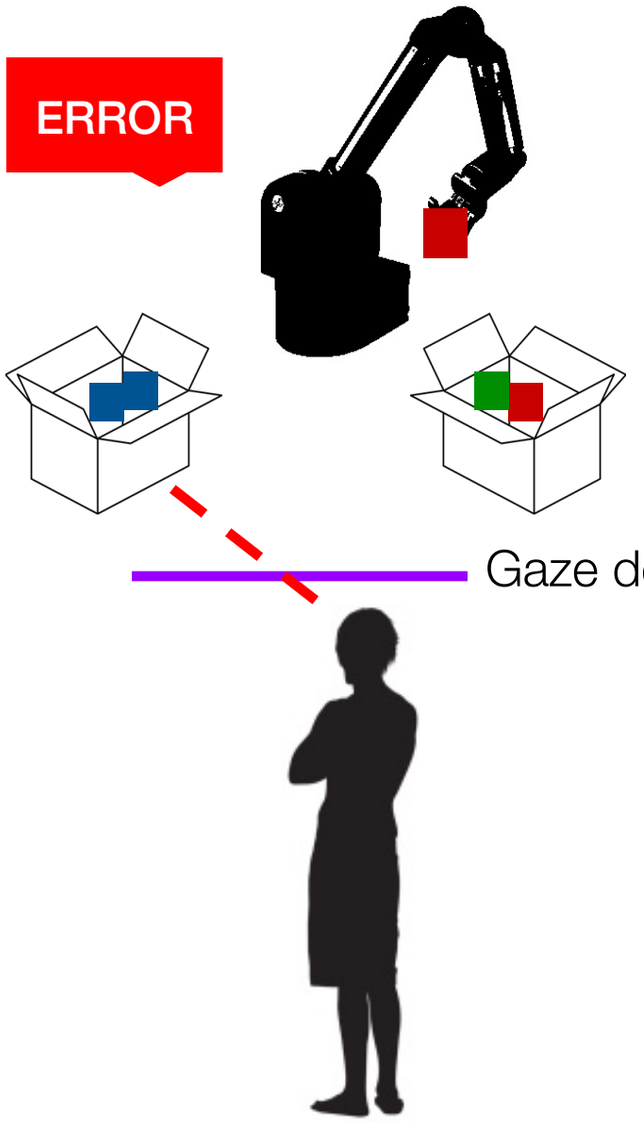
Quick overview — Scenario

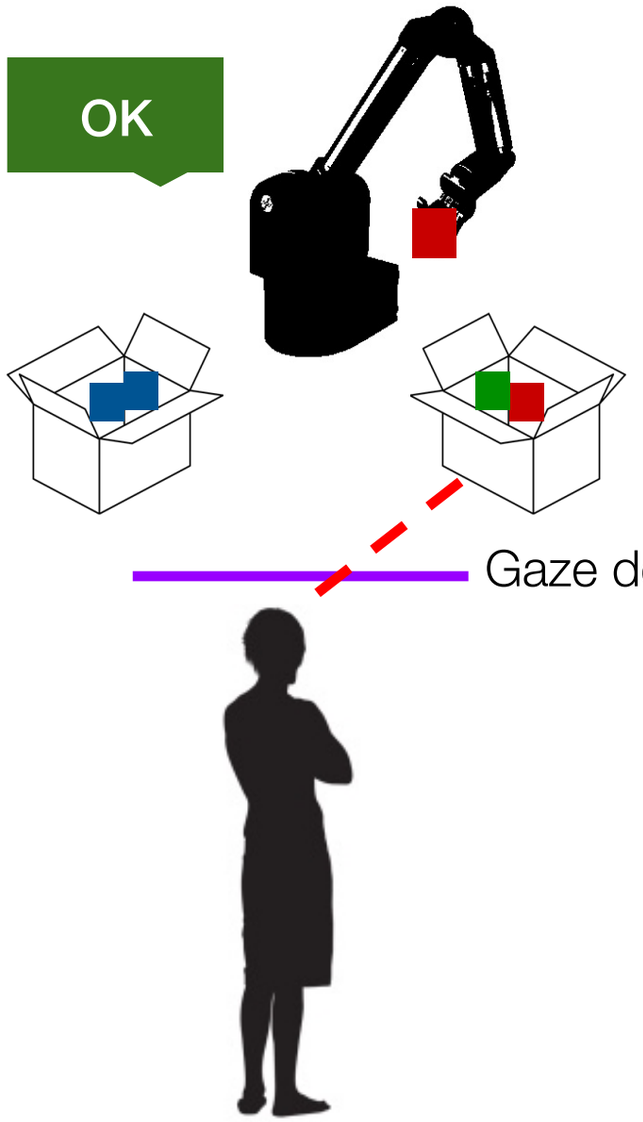






Quick overview — What to pick?





Quick overview — Where to drop?