Cognitive Robotics

LM-32 Ingegneria Informatica

2018/2019

BOCCUZZI STEFANO
BOTTIGLIERI ILARIA
PALATUCCI UGO
PASQUALE DONATELLA

Overview

- FUNCTIONAL SYSTEM SPECIFICATIONS
- HARDWARE ARCHITECTURE
- SOFTWARE ARCHITECTURE
- RESULTS AND CONCLUSIONS

Task

FIND A BALL

PUT THE BALL IN THE BOX HAVING THE SAME COLOR

REPEAT STEP 1 UNTIL

1. THERE ARE NOT MORE BALLS
2. THE TIME IS EXPIRED

Functional System Specifications: design choices

DESIGN CHOICES

1. NUMBER OF CAMERAS
2. CAMERA POSITION

HOUGH CIRCLES TRANSFORM

- EXTRACT THE FEATURES FOR CIRCLES DETECTION
- HSV COLOR-BASED RECOGNITION

VIOLA-JONES

- ON BOXES

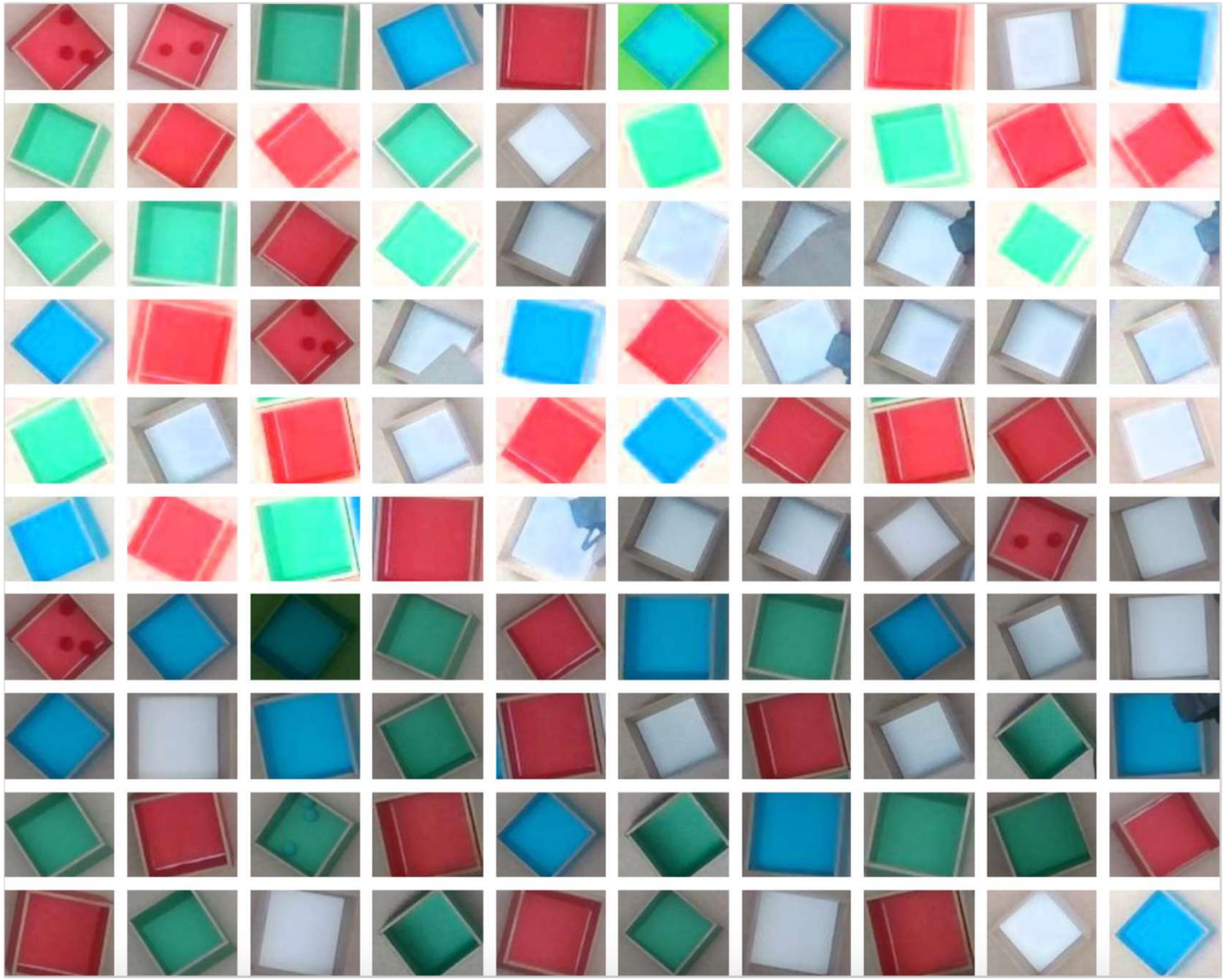
- 3 main possibilities for camera position: **zenithal position**, on **robot's arm**, or use two cameras on **both positions**.
- With two cameras the team expected to be the more accurate and fast set: one camera had to be connected to the Jetson and one had to be connected to the robot.
- The first camera could capture a snapshot of the objects
- The second camera could precisely control the robot movements to grab the balls.
- The team chose to **use a zenithal camera** after some experimentations. The reason was that the precision using this set was enough to grab the balls without corrections.

DESIGN CHOICES

Camera Position

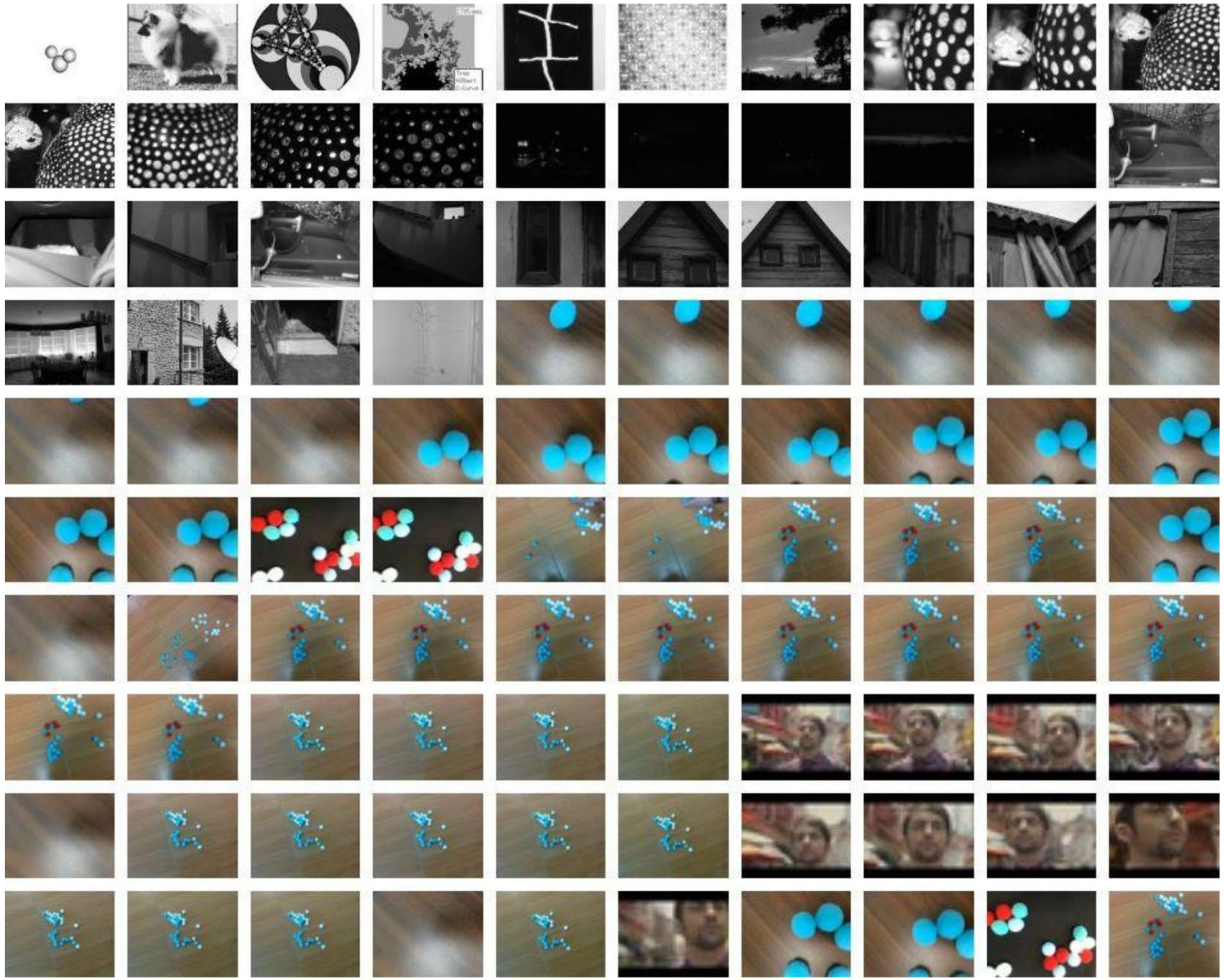
and

Number of Cameras



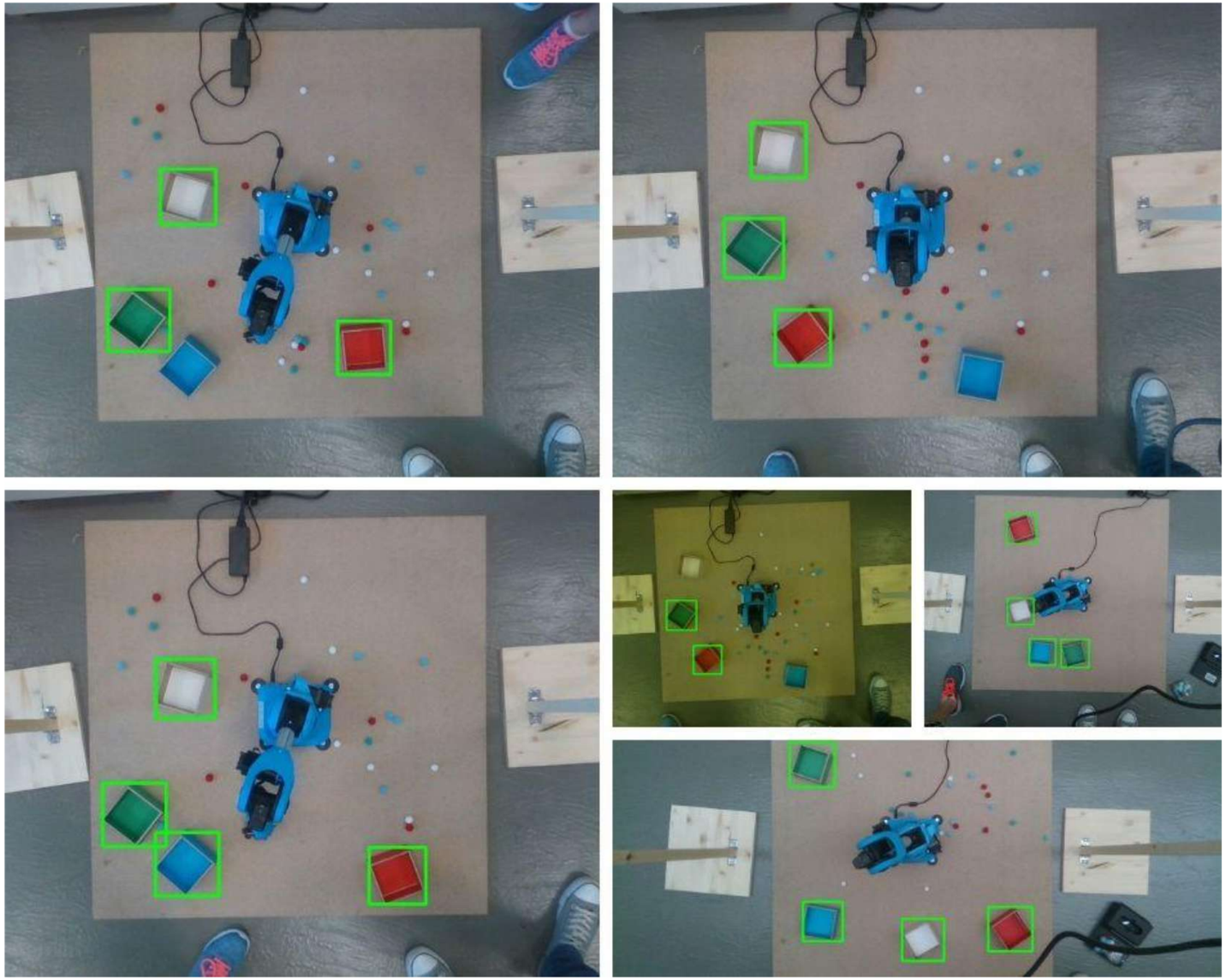
VIOLA JONES ON BOX

Positives 2.423



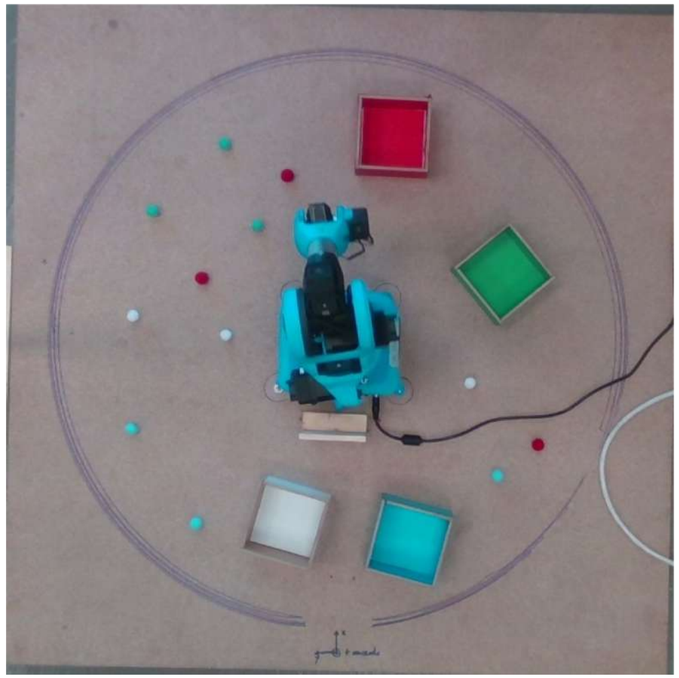
VIOLA JONES ON BOX

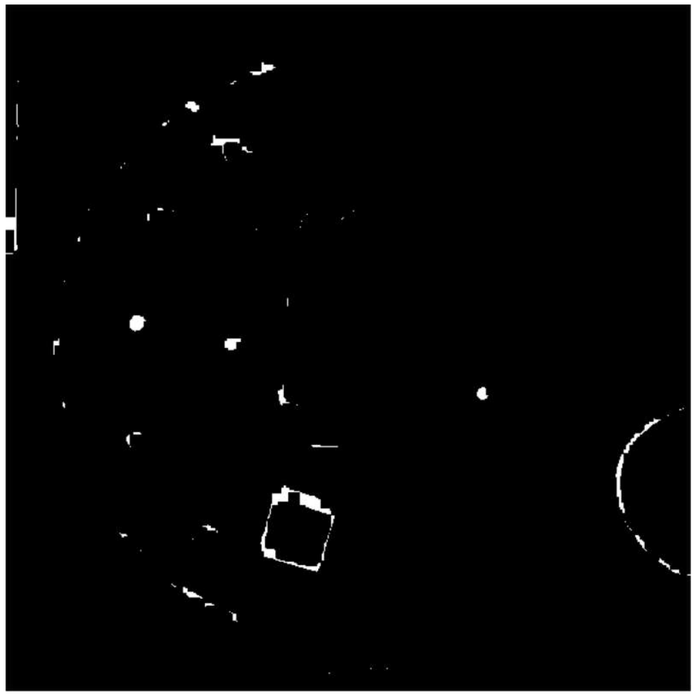
Negatives 16.524

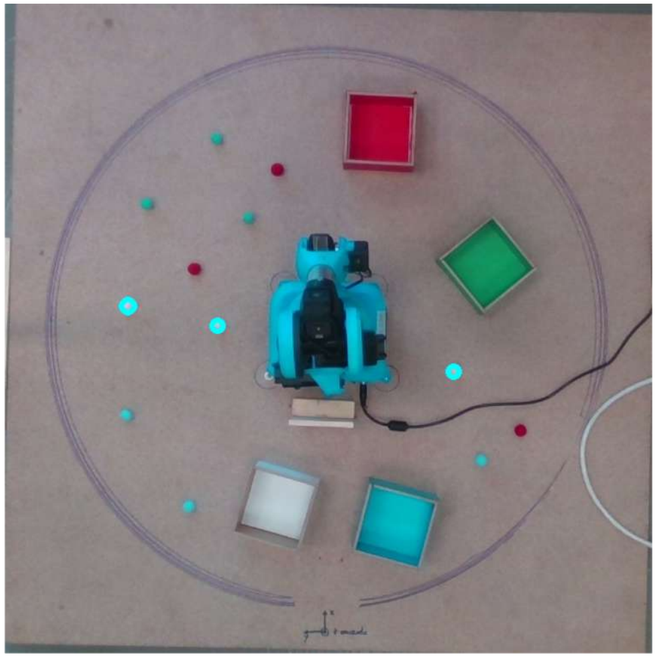


VIOLA JONES ON BOX

Example of detection







Hough transform EXAMPLE OF WHITE BALLS DETECTION

Hardware Architecture

INTEL REALSENSE
DEPTH CAMERA D435

NIRYO ONE

NVIDIA JETSON

System architecture

PROBLEMS ABOUT POWER SUPPLY

SOLUTION:

- POWER SUPPLY WITH A JACK OF 6 MM
- JUMPER INSERTED ON THE JETSON
- USB ADAPTER FOR THE WI-FI NETWORK
- CONNECTION BETWEEN PC AND JETSON VIA ETHERNET CABLE
  - ESTABLISHING ON THE PC A DHCP SERVER

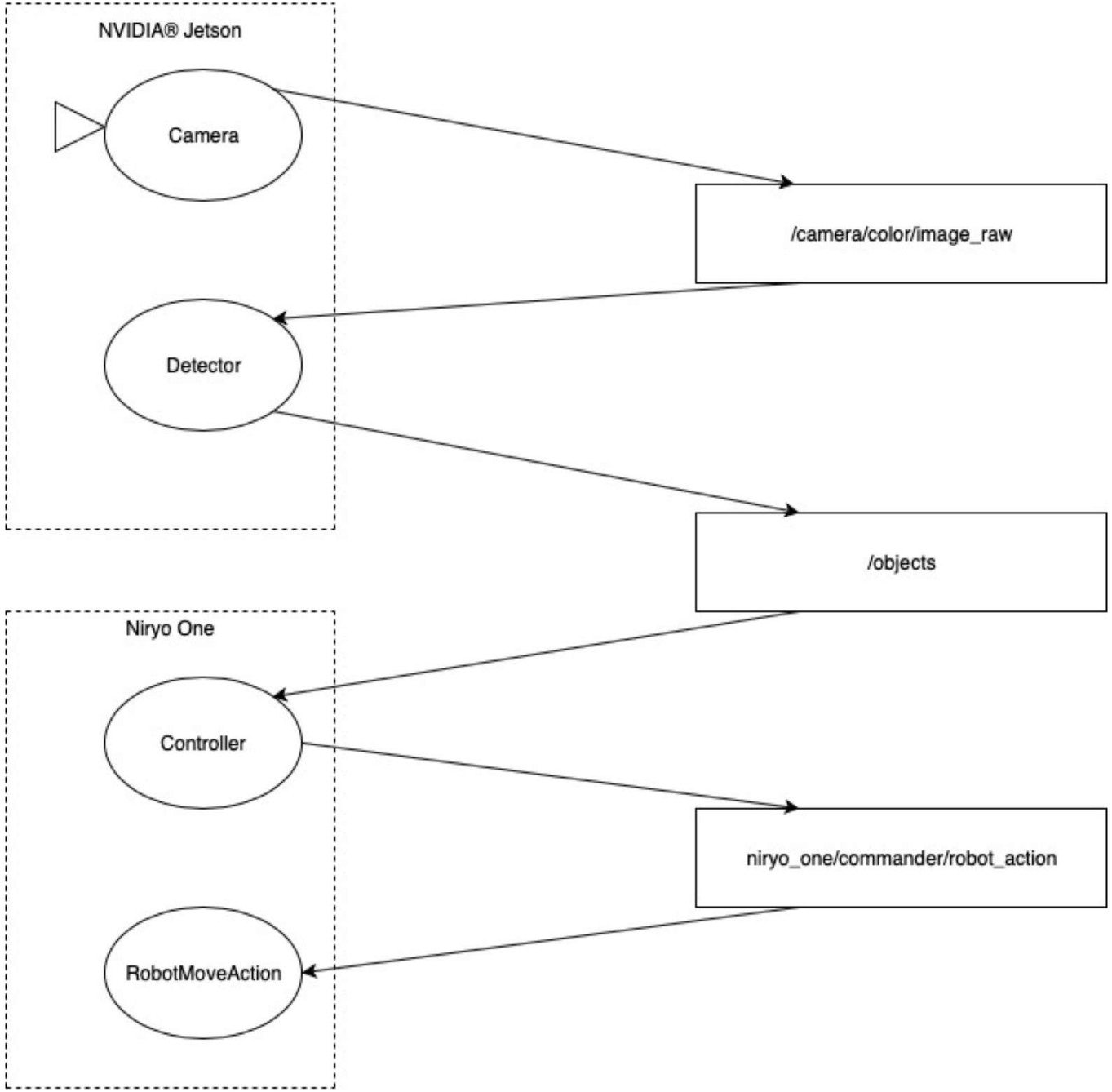
Software architecture

OBJECT_DETECTION.PY

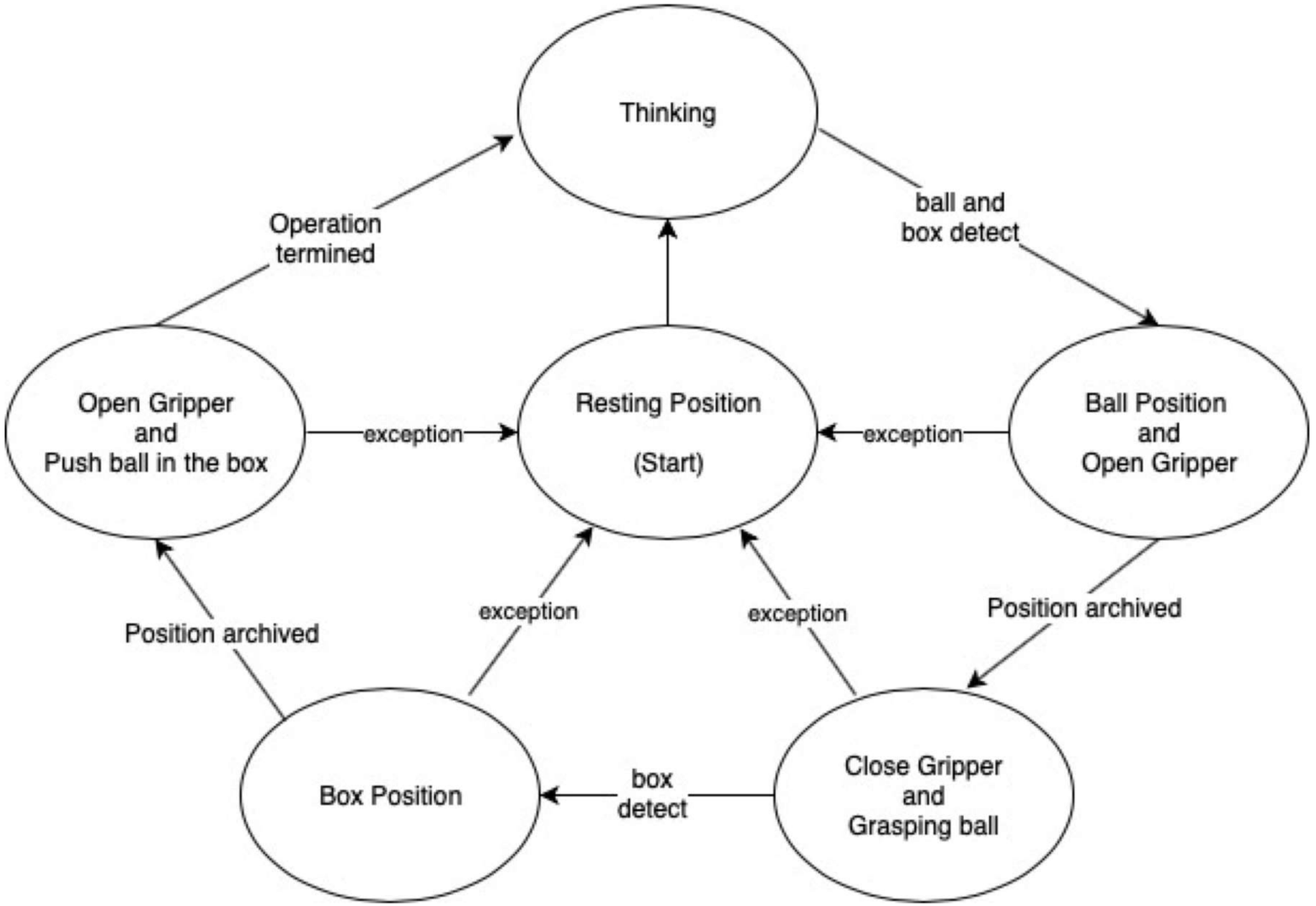
CONTROLLER.PY

ROS SCHEME

ROBOT MOVEMENT SCHEME


NVIDIA® Jetson
Camera
Detector
/camera/color/image_raw
/objects
Niryo One
Controller
RobotMoveAction
niryo_one/commander/robot_action

ROS SCHEME



ROBOT MOVEMENT SCHEME

Results and Conclusions

PROBLEMS:

THE SYSTEM HAVE TO BE **CONFIGURED PROPERLY** IN ORDER TO HAVE AN ACCURATE ROBOT MOVEMENT.

HOW TO IMPROVE THE SYSTEM?

ADD A SECOND CAMERA ON ROBOT ARM

Thank you for the attention !!!