Series 03 – Camera & Communication

Robotics, BSc Course, 2nd Sem., Dr. Julien Nembrini, Marta Visetti

Handout on March 24 2025 Due on April 6 2025

Reading

— Study the lecture notes, including links, and source code available on Moodle

Sections to be completed in the template report

— *2.3 Sensors → Camera*
— *3.5 Behaviours → Object recognition*
— *3.6 Behaviours → Multi-robot coordination*

1 Camera measurements

On two different e-puck robots, measure the camera response values on the three color channels for two different blocks of similar colors (either red, blue or green). To understand the response of the new cameras, compare two robots from the following IDs :

a) 4xxx (but not robot 207)
b) 5xxx (only robot 216 and not 203 or 214)

Use the provided controller and python script to generate and represent the data using histograms, in order to display how many camera pixels have a specific color response values. Summarize and discuss your results in the *2.3 Sensors → Camera* section of the report template.

2 Color and object recognition measurements

On a real e-puck, measure the efficiency of both object- and color-detection models implemented in the API for different types of targets at different distances. Switch on the lights of the arena, position the robot in the middle, and scatter a few targets (different colored blocks, the black ball and another e-puck) across the field. The robot will turn upon itself and record detections.

Run the provided controller *S03_object_color_recog_record.py* to save data, and the script *S03_object_color_recog_plot.py* to plot the data. If the recognition model made false recognition (position outliers), the camera frames with added detection layer will be saved in your folder as pictures (folder `./img`).

Describe your setup, select the most meaningful graphs produced by the plotting script, compare the results of color- and object-recognition, and finally discuss potential false recognitions obtained in the *2.3 Sensors → Camera* section of the report template, **adding a subsection** after the discussion about the basic camera.

Do not forget to update the library

```
pip install --upgrade unifr_api_epuck
```

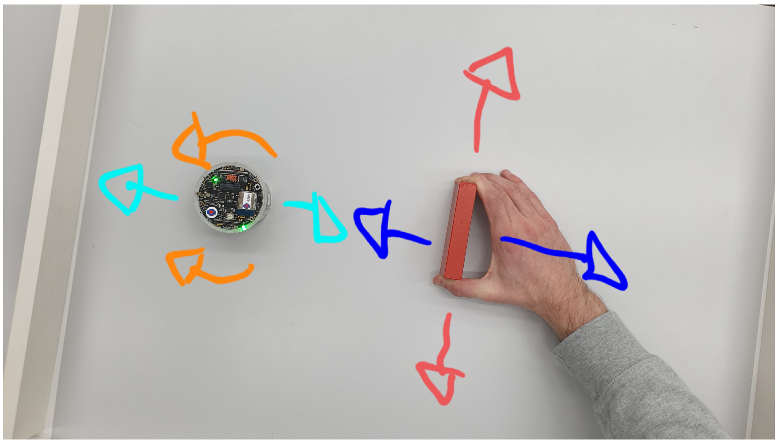

FIGURE 1 – Object LOVER : actions on the block and reaction of the robot

3 Object LOVER

Implement a controller where the robot uses the object *and* color detection API to achieve a LOVER behavior with respect to a red block.

When the robot doesn't see a "Red Block" or a "Red" color patch in its field of view, it doesn't move. When a "Red Block" or a "Red" patch is detected, the robot will try to reach an equilibrium position at approximately 10cm of the block (too far away for the prox sensors). If the block is moved laterally, the robot must adapt its orientation. If the block is moved further or closer, the robot must adapt its position (see Figure 1). The robot should be able to ignore a second block of another color inside its field of view.

Hint : implement 2 different ds, one responsible for keeping the block in the center, one responsible for mainting the distance, using different attributes of the block/color detection

Discuss the process and your solution, and compare the performance and responsiveness of both methods in the *3.5 Behaviours → Object recognition*. Record a short video to illustrate the solution, including difficult situations (multiples blocks, quick movement of the block, etc...)

4 Detection sharing

Modifiy a simple EXPLORER controller that detects color blocks with its camera, such that two robots running the same controller both switch all their LEDs on when they see a similar object configuration (e.g. when robot 1 and robot 2 both see exactly one red block and two green ones, for instance), and off otherwise.

Hint : make robots send themselves messages containing the current detected configuration.

Discuss the process and your solution in the *3.6 Behaviours → Multi-robot coordination* section of the report template. In particular, motivate your choice of algorithm between object or color recognition. Draw the FSM of the controller of *a single robot* with special detail on the events. Record a short video to illustrate the discussion.

5 Abstract, Introduction & Conclusion

Once you have completed your report, re-read it and write an abstract, an introduction and a conclusion to it. Submit the completed report.

Abstract It states the report's important contributions and should be synthetic and understandable by itself.
Introduction It exposes the larger context and draws the reader towards the specifics of the report.
Conclusion It summarizes and closes the report.

Although we do not formally follow the IMRad model, these links help you to understand the different roles of these specific sections.

— `https://writingcenter.gmu.edu/guides/writing-an-imrad-report`
— `https://writingcenter.gmu.edu/guides/writing-an-abstract`
— `https://writingcenter.gmu.edu/guides/imrad-reports-introductions`
— `https://writingcenter.gmu.edu/guides/imrad-reports-conclusions`

Hint : Reading abstract, introduction and conclusion should give a good idea of the report's content.