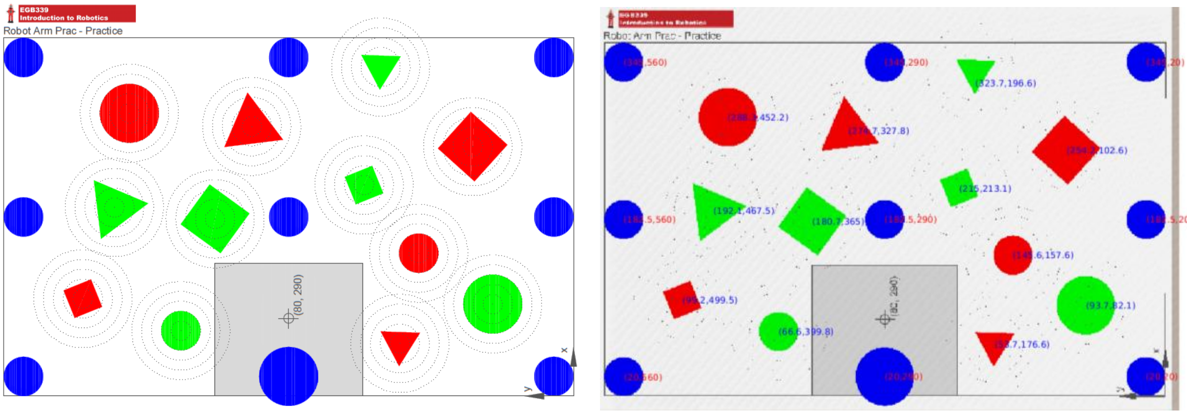
EGB339 – Kinematics Prac Exam

Updated: *12th of October 2020*

Due Date: Friday Week 13 (11:59pm)

1. Pick and Place Challenge

Your task for the kinematic practical assessment is to program a robot arm to move three small cylinders from their initial position on a given worksheet (Figure 1) to a destination position designated by a set of coloured shapes (Figure 3). In addition, you will need to use computer vision methods to calculate the coordinates of each shape for the pickup and place locations for the three cylinders.



(a) (b)

Figure 1a - Example Exam Sheet. 1b – Example exam sheet coordinates

2. Simulator

You will be provided with a simulation scene that includes a 4 DoF robotic arm and a camera sensor that cannot be modified (see Figure 2). A MATLAB library to interface with the simulation environment and to command the joints of the robot will also be provided. The simulation environment for this assessment will be the CoppeliaSim simulator. A lightweight scene player can be downloaded for free from this website (https://www.coppeliarobotics.com/downloads). Alternatively, the full install of the education edition of the simulator can also be downloaded and installed but is not required.

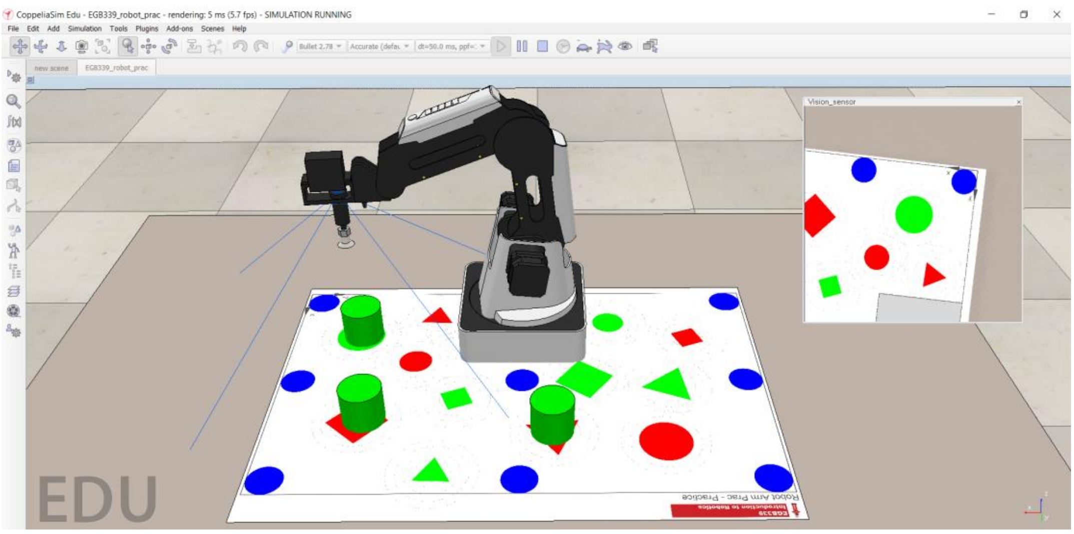


Figure 2 - CoppeliaSim Simulation Environment

3. Matlab Function

In order to easily assess your solution, each group will need to submit their code as a single MATLAB .m file and include a single function that takes four input arguments **(dobot_sim, init_positions, dest_positions, image_flag)**. The last argument indicates whether the input positions are passed in as images or coordinates.
Note: Peter Corke's toolbox functions will ***not*** *be available to you during the assessment.*

4. Submission Process

For your solution to be marked you will be required to upload your code in one **.m** file (please see the example **.m** file supplied with the prac exam files) by the due date. In addition to your single MATLAB file you will also be required to submit a **.json** file which provides flags for the marker to use to determine how to execute your code and mark your solution. An example json file is also provided in the prac exam files and how they are used to determine your marks can be found within the example marksheet.

An example marking script has also been provided which can be used to check that your solution is compatible with the marking system.

5. Important Details

A worksheet (see Figure 1) will be placed underneath the robot arm and will be different to practice exam sheets. The shapes other than the grey square and the blue calibration marks will be in the set [small, large], [red, green], [triangle, circle, square].

- The blue circles are calibration markers. Note that one blue circle is larger than the others.
- The shapes will appear at most once in every combination of size (large or small), shape (circle, triangle and square), and colour (red and green). That is, there cannot be a shape that has the same combination of size, shape and colour as another; there will be one small green triangle, one large red circle, etc.
- The shape and size are independent of colour. That is, all large triangles are the same size, all small circles are the same size, etc.
- The worksheet image is to be taken by the vision sensor at the end of the arm using the getImage() function call. The vision sensor has been programmed to make the robot invisible and the objects invisible so only the exam sheet can be seen to make the shape detection easier.

The initial positions and goal positions will be provided in the form of two additional small test sheets and will be passed as input to your function as images during the assessment.

- Each test sheet contains three coloured shapes (red/green, circle/square/triangle, small/large).
- The test sheets will be of fixed resolution, similar to examples given
- We will run your code twice, with different test sheets and give you the higher mark.

The first sheet provides the initial positions, the second provides the destination positions. An example is provided below (Figure 3)

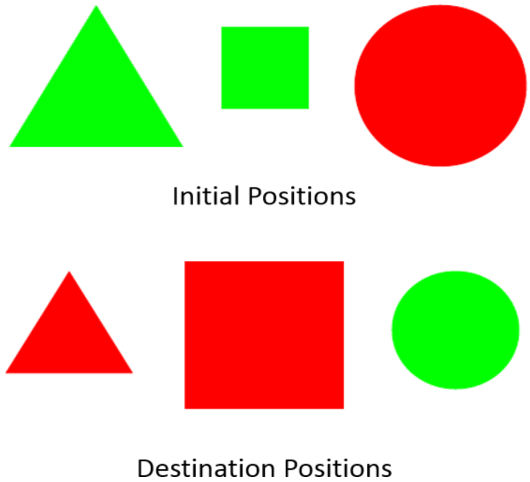


Figure 3- Example of test sheets

- The two test strips will be different for each of your two attempts.
- You will not get to choose which of the two test strips you are given as the initial position sheet and which is the destination position sheet.

In the example scenario given by the test strips above, your task is to move the cylinder placed on the large green triangle to the small red triangle, the cylinder from the small green square to the large red square, and the cylinder from the large red circle to the small green circle.

Note:

- The colour of the cylinders is of no importance.
- The order of cylinder placement does not matter if the cylinder on the leftmost shape on the initial position sheet ends up on the leftmost shape of the destination positions sheet, the cylinder on the centre shape on the initial position sheet ends up on the centre shape of the destination position sheet, etc. That is for the above example, small green square to large red square, large green triangle to small red triangle, and large red circle to small green circle is acceptable. But small green square to small red triangle, large green triangle to large red square, and large red circle to small green circle is not acceptable.
- The cylindrical blocks will not be visible on any of the sheets when photos are taken. That is, the cylindrical blocks will not appear in any of the photos you take but will be placed on the large worksheet during the pick and place challenge.

6. Tasks for the team:

As a team, you must develop a program that will perform the following tasks:

1. Automatically identify the coloured shapes (shape, colour and size) of the test sheets you are given and calculate their coordinates on the bigger worksheet. This resembles the vision prac task from the first half of this unit. For each shape on the test sheets, return the calculated coordinates for the initial positions, then the destination positions. The coordinates should be returned in units of mm.

   a) If your vision component produces the correct coordinates of the shapes on the worksheet for the initial position and destination position test sheets, you will receive full marks and can proceed to task 2. This will be judged based on the accuracy of the coordinates with the true coordinates.

b) **Alternatively:** You can agree to forfeit these marks and the initial and final coordinates will be given so that you don't require any vision code and proceed to task 2. We will have a flag in the marking code to indicate you would like to forfeit and the system will pass in the true coordinates instead.

2. Move all cylinders from their initial positions to the correct destination positions as specified on the test sheets or the received coordinates. You will receive marks based on how accurately the cylinder is placed on the shape after the script terminates.
Note: You will not be allowed to reposition <u>any</u> cylinders during the attempt with anything other than the robotic arm.

   For example: If you pick and place the first cylinder correctly, and then knock it over when placing the second cylinder, you will be marked for accuracy based on the cylinder's final location once the script has terminated.

   For example: If you are moving the first cylinder and accidentally knock the second cylinder away from its initial position, you will not be able to manually reposition the cylinder.

   Alternatively: Move the tooltip of your robotic arm in the following sequence.

   a) First initial position, then first destination position;
   b) Second initial position, then second destination position;
   c) Third initial position, then third initial position.

   Note:
   - The cylinders will be removed from the worksheet if you elect to perform this alternative task.
   - You will receive marks based on the position accuracy of the arm when moving to the specified shapes.
   - The order of steps a, b and c do not matter as long as the initial positions correspond with the correct destination positions.

   Examples:

   - Moving from the centre shape on the initial position test strip to the centre shape on the destination strip, moving from the leftmost shape on the initial position test strip to the leftmost shape on the destination strip, then moving from the rightmost shape on the initial position test strip to the rightmost shape on the destination strip is an acceptable sequence.
   - Moving from the centre shape on the initial position test strip to a shape other than the centre shape on the destination strip is a partial example of a sequence that would lose marks.