Project 02 - Pick and Place

MTRN4230 - UNSW School of Mechanical and Manufacturing Engineering

Changelog

- v2: Alternated ground truth coordinates for correct positions.

1 Learning Outcomes

- Apply concepts in computer vision to enable the robot to interact with its environment.
- Apply coordinate transformations to convert between camera and robot frames.
- Utilise path planning to enable the robot to complete tasks successfully.

2 Your Task

Your goal is to implement a basic 2D pick-and-place solution. Using the vacuum gripper, you will move two items, one small and one large, to a target position while avoiding obstacles. The small item must be moved using an artificial potential field–based path-planning algorithm, while the large item may be moved using any path-planning technique.

At the start of your attempt, a demonstrator will randomise the locations of the items. Colliding with an obstacle will immediately terminate the run. You are allowed two attempts total, however, the demonstrator will adjust the locations of the objects after a failed first run, and you are not allowed to modify your solution between runs.

Goal Board

The goal of this assignment is to place the items in the correct locations on the goal board. The final orientations of the pieces may be rotated, provided each piece still fits within its designated goal location. If the items are within the bounds of the goal board and touching it, but not in the correct designated slots, reduced marks will be awarded. There is a 5mm offset around the item slots for clearance.

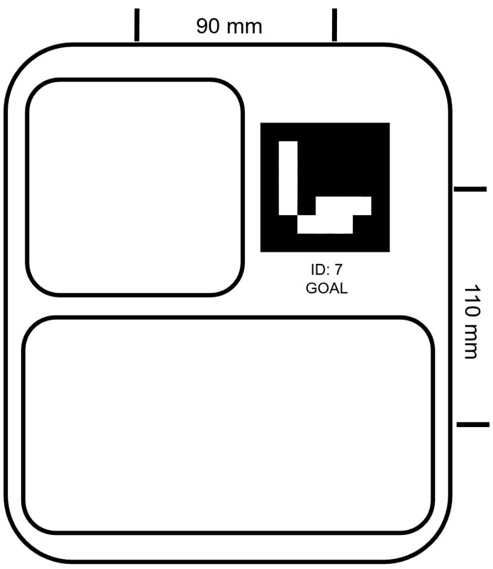

Figure 1: Goal board with measurements.

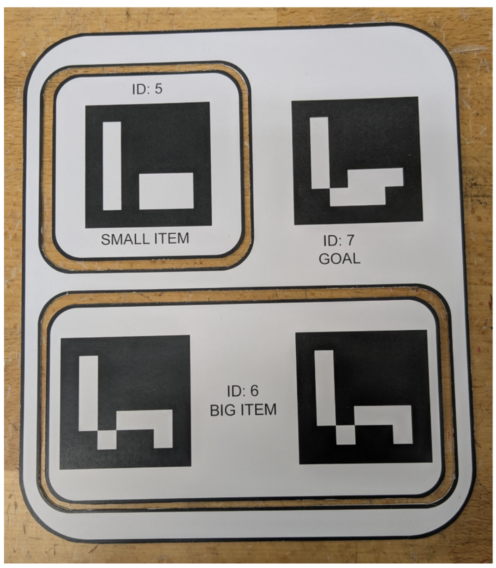

Figure 2: Goal board with items.

Obstacles

Up to three obstacles will be randomly placed on the table at the start of your run. Collisions with an obstacle will automatically terminate the run. A collision is defined as any overlap between a moving item and an obstacle. It is at the discretion of the marker to determine whether a collision has occurred, and their respective decision is final.

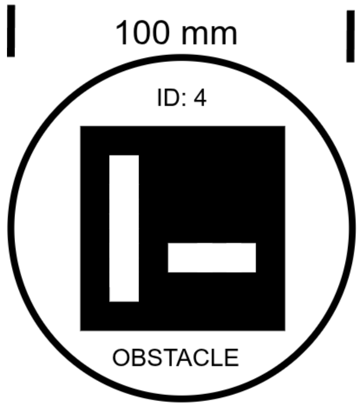

Figure 3: Obstacle.

Items

There are two items, as shown below. Items are not considered obstacles and may be moved through each other freely.

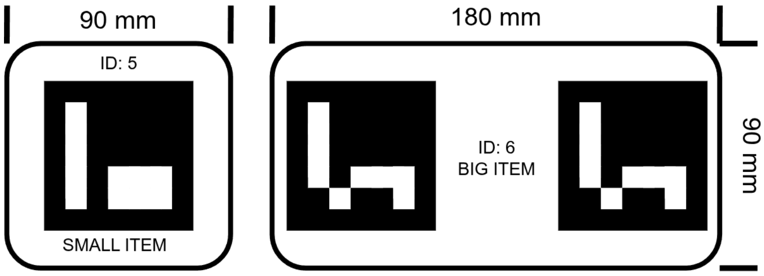

Figure 4: Items.

Vacuum Gripper

To pick up and move items around the gameboard, you will use the vacuum gripper end effectors. Examples of how to use the vacuum gripper can be found in the RTDE example folder under Example 6.

Sample Images

In the same folder as the assignment specification, three sample images are provided that you can use to test your solution. These images are in ideal lighting conditions and may not reflect the true lighting and reflections on the day of your assessment. On the day of the assessment, you will capture images of the state of the game using the webcam. You are encouraged to take additional sample images of the gameboard yourself to test on.

2.1 Part A: Transformation (4 Marks)

You will need to use MATLAB to complete the following:

- Use the top-down view provided by the webcam to identify the table markers and relevant ArUco markers.
- Calculate the homography matrix and perform a perspective transformation.
- Obtain the pose of the small item, the goal, and the obstacles relative to the robot's base joint. You have been provided with four ground truth coordinates (see Appendix) to complete the perspective transform.
- Crop the warped image to the field size, then mark the obstacles with a red dot, the goal with a green dot, and the small item with a blue dot.
- Display the resulting image.

To achieve full marks for Part A, you must display a figure that meets all the listed criteria and resembles the image shown in Figure 5. A marking breakdown can be found in Section 3.

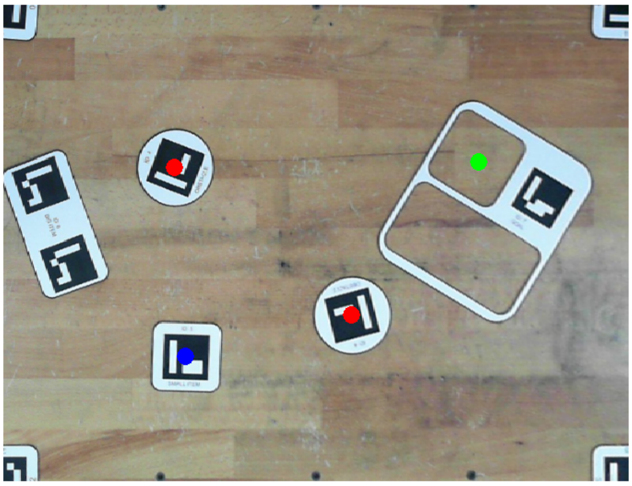

Figure 5: Example Part A result.

2.2 Part B: Artificial Potential Field (4 Marks)

In Part B, you will implement an Artificial Potential Field (APF) algorithm to plan a path that would move the small item to the goal location in the correct position.

At the start of each trial, the demonstrator will randomly place the items. You will never be provided with an impossible configuration where a local minimum would occur using the standard APF approach. You may use the lecture material as a guide for implementing the APF algorithm, however, you do not need to replicate it exactly and may make modifications to improve the robustness of your solution. You will need to use MATLAB to complete the following tasks:

- Create a new figure to visualise the APF.
- Place a green dot for the start location, a red dot for the obstacle, and a blue dot for the goal in the robot coordinate frame.
- Use the `quiver` command in MATLAB to display the vector field on the cropped figure.
- Step through the vector field with a step size of 5 mm until the goal is reached, and display the resulting trajectory as a purple line.

Below is a generated figure illustrating the expected output for Part B (Figure 2.2), Your solution does not have to exactly match the examples shown; these are for demonstration only. You are encouraged to combine all figures of Task A and B into the same figure, provided the information is clear. A marking breakdown can be found in Section 3.

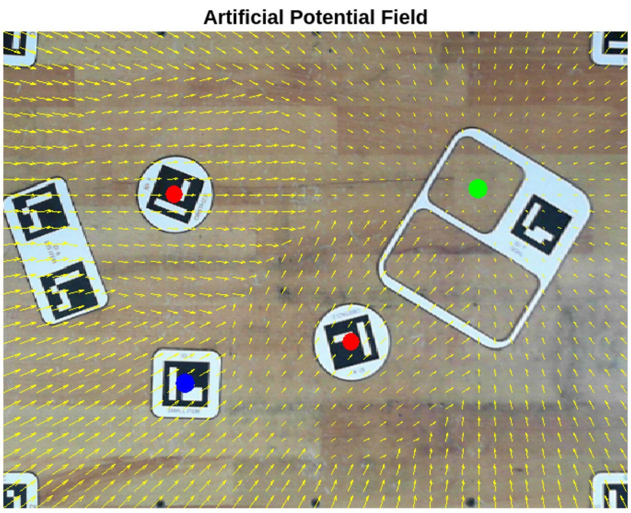

Figure 6: APF example using quiver.

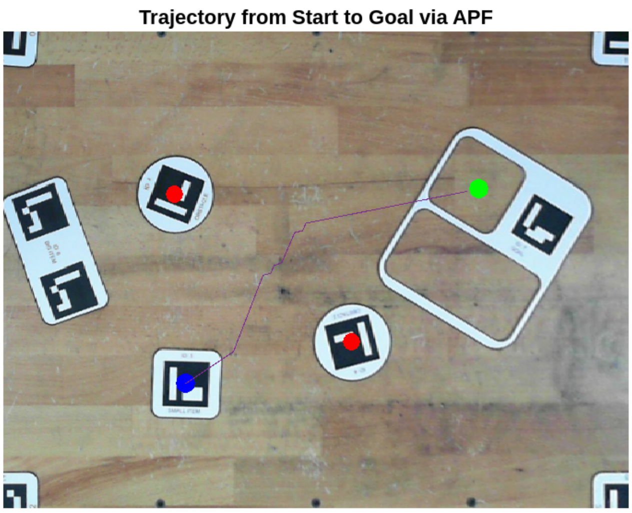

Figure 7: APF with path.

2.3 Part C: Small Item Movement (5 Marks)

Use the vacuum gripper to move the small item to the goal location and place it in the correct slot with the correct orientation. This must be done by following the trajectory generated using the APF path from Part B. The item must be moved at a height of 6 mm above the table. A marking breakdown can be found in Section 3.

2.4 Part D: Big Item Movement (5 Marks)

The big item has two additional considerations compared to the small item. Firstly, the big item must be picked up from the midpoint of the two conjoined ArUco markers; failing to do so will result in a loss of marks. Secondly, you must account for the rotation of the item while avoiding obstacles and placing it in the goal location. For this task, you may use any path-planning method of your choice and implement it in a separate file. No specific hints or approaches will be given; it is up to you to effectively solve this problem. The item must be moved at a height of 6 mm above the table. A marking breakdown can be found in Section 3.

2.5 Part E: Inverse Kinematic Trajectory (2 Marks)

Instead of using the existing pose control MATLAB RTDE commands, you need to create your own using the RVC Toolbox inverse kinematics solver. Then, use the RTDE array joint position commands to complete Part C. To receive marks for this part, you must clearly explain your code implementation to the marker and demonstrate it working for Part C. No specific hints or approaches will be given; it is up to you to effectively solve this problem.

3 Marking Criteria

This assignment is worth 20% of the total course mark. The breakdown of marks for this assignment is described in Table 2.

Criteria	**Marks**	**Description**
Part A	4 marks total	1 mark – Warped image is displayed. 1 mark – Warped image is cropped. 2 marks – Warped image is correctly labelled.
Part B	4 marks total	2 marks – Vector field is displayed in the robot coordinate frame. 2 marks – Trajectory is generated and displayed in the robot coordinate frame.
Part C	5 marks total	1 mark – Moves to the small item. 1 mark – Picks up the small item. 1 mark – Places the small item on or within the goal board. 2 marks – Places the small item in the correct position.
Part D	5 marks total	1 mark – Picks up the large item from the midpoint of the conjoined markers. 2 marks – Moves to the goal without intersecting the obstacle. 1 mark – Places the big item on or within the goal board. 1 mark – Places the big item in the correct position.
Part E	2 marks total	2 marks – Successfully implements and demonstrates inverse kinematics movement for the Part C task.

Table 1: Mark allocation for the assignment.

4 Submission

You must implement your solution in a **single MATLAB `.m` file**. When executed, this file should allow the Demonstrator/Marker to select which part (A–E) they wish to test.

- The assignment will be marked during your Week 12 lab session.
- If you are unable to attend your scheduled lab, you must apply for *special consideration.*
- Submit your complete solution to Moodle as a single `.m` file by **Friday of Week 12, 21:00**.

5 Plagiarism

If you are unclear about the definition of plagiarism, please refer to What is Plagiarism? — UNSW Current Students.

You could get zero marks for the assignment if you were found:

- Knowingly providing your work to anyone and it was subsequently submitted (by anyone), or
- Copying or submitting any other persons' work, including code from previous students of this course (except general public open-source libraries/code). Please cite the source if you refer to open source code.

You will be notified and allowed to justify your case before such a penalty is applied.

Appendix - Ground truth locations

Table 2: Ground truth table marker positions (mm).

Marker	X (mm)	Y (mm)
0	-990	60
1	-230	60
2	-990	-520
3	-230	-520