CS6341 Robotics Project Mid-term Report

Professor Yu Xiang

October 21, 2025

1 Mid-term Report (5%)

The project mid-term report should be prepared using the the ICRA double column latex format. A useful online LaTex tool is Overleaf `https://www.overleaf.com/`. We have the ICRA latex template accessible here via overleaf: `https://www.overleaf.com/read/rwmhwnwjkrmc`. You can download a copy of the template or make a copy in overleaf for your own project, and then edit it. If you are already using this template for the project proposal, you can extend the project proposal for this mid-term report.

In this project mid-term report, please describe the following items according to your progress:

- **Title**. The title of your project.
- **Team Members**. List the names of the team members.
- **Abstract**. Give an overview of the project.
- **Introduction**. Describe the motivation of the project, i.e., why do you want to work on this problem. Then describe an overview of the framework/method/system.
- **Related Work**. Discuss the related work of your project.
- **Method**. Describe your solution for the project. For example, describe each component of the framework in details. Try to use figures to illustrate the method instead of only using text. "A picture is worth a thousand words".
- **Experiments**. Describe what experiments you have done so for. If there are some experimental results to show, add them to the report. You can use figures and plots to show these results.
- **Next Steps**. Describe what are the next steps for the project.
- **References**. Cite related works in the report.

Evaluation criteria: The evaluation of the mid-term report will be focused on the introduction, the method and the experimental setup. So try to provide details on these parts. It is okay if there is no much experimental result in this mid-term report. In this case, you shall write down the next steps clearly.

Minimum page requirement: 3 pages. The report should be at least 3 pages with the ICRA format (excluding references, i.e., without references, the content should be at least 3 pages). You can go beyond 3 pages, but make sure it is less than 6 pages (excluding references).

An example ICRA paper: you can check the structure of the following paper for reference `https://yuxng.github.io/Papers/2020/meng_icra20.pdf`.

2 Control of the SO-101 Arm (5%)

Additionally, a video demonstrating the successful control of the SO101 Arm should be submitted along with the report. The robot should be cycled through the predefined joint configurations shown below, it could cycle using keyboard presses or in a timed manner (change position every 3 seconds). Calculate the amount of time it took for the robot to get to it’s target position upon receiving the command and print out the final joint position; be sure to show the final joint position and time taken in the video through a terminal output.

Joint	**Position 0**	**Position 1**	**Position 2**
Shoulder Pan	0.0°	45.0°	-45.0°
Shoulder Lift	-30.0°	-20.0°	-10.0°
Elbow Flex	30.0°	30.0°	0.0°
Wrist Flex	70.0°	60.0°	-30.0°
Wrist Roll	0.0°	-30.0°	30.0°
Gripper	0.0	50.0	30.0

2.1 Code Fragment: Sending a Position

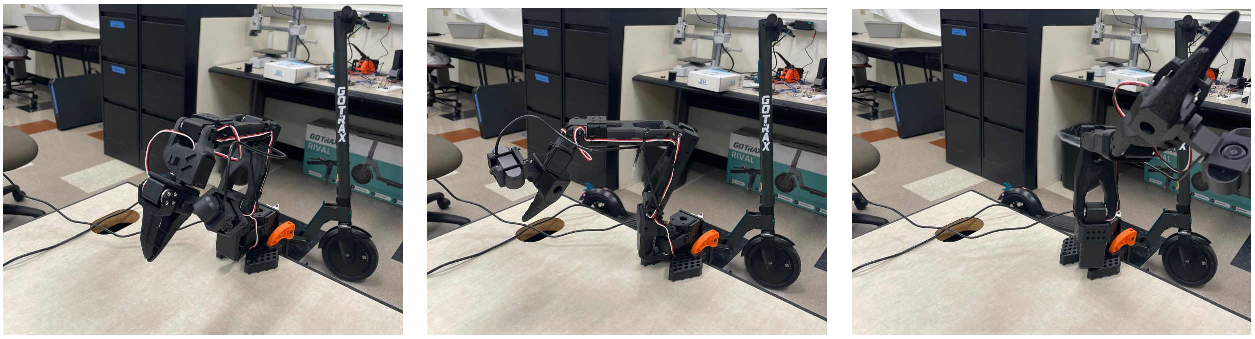
```python
from lerobot.robots.so101_follower import SO101Follower, SO101FollowerConfig

robot_config = SO101FollowerConfig(port="/dev/ttyACM0", id="demo_follower")
robot = SO101Follower(robot_config)
robot.connect()
position = {
    'shoulder_pan.pos': 0.0,
    'shoulder_lift.pos': 0.0,
    'elbow_flex.pos': 0.0,
    'wrist_flex.pos': 0.0,
    'wrist_roll.pos': 0.0,
    'gripper.pos': 0.0
}

# Send joint position to robot
robot.send_action(position)

# Reading joint posiitons
joint_pos = robot.get_observation()
```

2.1.1 Hints

- Use the same Anaconda environment you used for callibration to run the code. The python libraries referenced above should be already installed.
- Remember to change the robot id to the one you named your robot and use the corresponding port.
- Use a tolerance of +/- 1° to calculate the time it took the robot to achieve the target position.

2.2 Submission

- Code: Submit your Python script (e.g., `so101_control.py`).
- Video: Show the arm cycling through positions, with terminal output of final joint positions and the time taken.



(a) Position 0 (b) Position 1 (c) Position 2

Figure 1: SO101 robotic arm at three predefined positions.