FP: Final Project

CSCI 3302: Introduction to Robotics

FP Planning due April 04th 11:59pm - extra credit [10%] to top 3 best documents!
FP Check-In due April 21st 11:59 pm – extra credit [10%] to top 3 best progress!
FP Submission due May 04th 11:59pm

Read the instructions very carefully

Remember this is 25% of your grade. Team members will be graded individually.

You will be working with your lab group for the final project. There are no bonus points for early submission on the final project. Goals for the first part of the final project are as follows:

Part 1: Create Project Report Doc
(Note that you will constantly update this report and it will serve as your final report during the project submission)

- Add each team member's name and e-mail to the top of the document.
- Add sections for:
  - Deliverables – **This section should have leads responsible for each part.**
  - Implementation Plan and interim deadlines
  - Demo – This section should discuss what tiers you plan to finally achieve and how you are going to showcase them
- Create a list of **deliverables**, where each deliverable is a component of your final project. You are **strongly** encouraged to subdivide deliverables into more actionable steps, such as:

  - [ ] Create Vision System – Lead: ____ Deadline: ____
    - [ ] Install OpenCV Python Package & Verify it works with simulated webcam
    - [ ] Capture images of target objects and identify their color ranges
    - [ ] Write color threshold algorithm to locate colored objects
  - [ ] Implement robot-side controller – Lead: ____ Deadline: ____
    - [ ] Create state machine with states for "Listen for command" and "Navigate to Pose"
    - [ ] Verify that commands are being received over the robot's Serial port

  This list is just an example and you do not necessarily need these components. The goal is to plan out how your result will look like, what it will require, and who is doing what and when.
- For each higher level deliverable, write an **implementation plan** detailing the steps necessary to go from design document to implemented product.

Include a target completion date and a designated 'lead developer' on your team for each item.

It is often helpful to specify tests for each major deliverable, indicating that you have come up with a way to determine if a particular component is working or not.

- Write a short script detailing how you will demonstrate your final project once it is completed, in such a way as to showcase its various components. It is **required** that you show off the methods used in order to achieve a certain tier. It is also required to show and report the total objects that you collect. For example, if you are using A* to navigate to the objects, show graphical output of your planner and maybe even relevant code snippets.

Have ONE person submit a copy of your team's project proposal to the "Final Project Planning" assignment on Canvas. Report any issues directly (and as early as possible!) to the professors/TAs via Piazza or E-mail, particularly in cases where a teammate did not contribute or where consensus couldn't be reached for teammate roles/responsibilities so we can unblock you as efficiently as possible.

Part 2: Understand the project requirements

1. You are provided with a grocery supermarket world and the Tiago robot that has a mobile base and a manipulator for you to use. The mobile base is the same as in the labs and you will use it to move your robot around and the manipulator (arm) will be used to pick-and-place objects into the basket attached to the robot. The robot is equipped with a LiDAR and a camera. You can add/remove sensors, but you will need to describe what sensing you require in your report.
2. There are objects that need to be collected spread across the store and your goal is to collect the maximum number of objects possible.
3. Your score will depend on the **number of objects you collect (20%)** and **the methods you use to collect them (80%)**.
4. You should NOT collect any objects that are not goal objects. Each goal object has 2 points and there are 2 bonus objects (**GREEN CUBES**) that are worth 4 pts each that can only be collected if you are able to do obstacle avoidance manipulation planning (No, you cannot move the obstacles away to clear your path).
   Your goal objects are **ONLY YELLOW CUBES**.

5. This task will require these 5 components:
   a. **Mapping** - You need Lab 5 like mapping to do planning for your robot's base movement.
   b. **Localization** - You need to know the robot pose in order to execute your plan.
   c. **Computer Vision (CV)** - You need CV to find out the objects of interest.
   d. **Planning for Navigation** - You need to do path planning in order to avoid obstacles and reach a point where you can then use the robot arm to pick-and-place the objects.
   e. **Manipulation** - You need to plan using the arm to pick-and-place the objects in the basket.

6. The point division for the above mentioned sections are mentioned in the table below. The method you develop to fulfill a section will affect your score. For example - If you use a basic method, you will only fulfill the 1st tier for a section and score lesser points.

   Out of 100 pts available for the project, it is possible to score up to 140 pts including the bonus pts, but that will require some serious effort. We strongly recommended that you achieve lower tiers first and then swap the individual modules/methods with better methods as you progress.

Mapping	Tier	Points -- 12
Manual	1	6
Autonomous	2	12
SLAM (Manual/Autonomous)	3	18

Localization		12
WeBots GPS/Compass	1	8
Odometry	2	12
SLAM / MCL	3	18

Computer Vision		18
WeBots recognition API	1	6
Color blob detection (HW3) (Recognition on Camera but use only color data, not recognition ID to identify which block obtained from webots API)	2	18
Machine / Deep Learning or any kind of object localization	3	28

Planning for Navigation		14
Teleoperation	1	5
A*	2	8
RRT	3	14
RRT w/ Path Smoothing	4	18

Manipulation		24
Trajectory: Hardcoding in Joint Space	1	12
Teleoperation in Cartesian Space (requires IK)	2	24
IK	2	24
Autonomous: Task-Level Planning + Obstacle Avoidance (IK + Hardcoded Waypoints)	3	30

Total Points (in %)	100
Objects	20
Completing Tiers	80
Bonus Objects Points (4 pts each object)	8
Bonus Tier Points	32
Maximum Points	**140**

You can notice that for some sections you can score some extra bonus points (in blue) by using a more sophisticated (usually sophistication is directly proportional to the generalization level of your method) method. The suggested methods for the tiers are not mandatory or exhaustive and you are welcomed to use other methods to perform the task.

7. Here are short explanations on some of the methods mentioned above. You don't need to code up a method entirely, you are welcomed to use Github repos with proper attribution. **Please keep your GitHub repository private so as not to share your code with others.** We will check for plagiarism. You are also welcome to use previous assignment submissions as your starter code.
   a. SLAM : Stands for Simultaneous Localization and Mapping. SLAM requires to create the map using only odometry. You can implement your own solution using knowledge about the environment or use open source tools. There are a bunch of SLAM implementations in python for 2D maps that you can use. For example - https://github.com/AtsushiSakai/PythonRobotics
   b. IK : Stands for Inverse Kinematics. You can implement your own solution that is custom made for this environment, or you can use a library to perform IK for the arm (given a desired endeffector pose, figure out the joint angles needed to achieve that). We recommend using - https://github.com/Phylliade/ikpy
   c. Teleoperation : Controlling the robot wheels or end-effector using keyboard keys manually.
   d. MCL stands for Monte Carlo Localization. You can implement a discrete probability map or a particle filter. Using MCL using a map that has been recorded using GPS will give you extra points in the localization, but not in the mapping category.
   e. You can use color blob detection to figure out the location of a block.

f. Path smoothing is pruning unnecessary waypoints obtained via A*/RRT and/or making it smoother with less sharp turns. Anything better than RRT works as an equivalent tier. For example- RRT*.
g. Obstacle avoidance in the context of manipulation will involve planning (with something like RRT). The IK can only take the end-effector to one waypoint. In order to avoid obstacles you need to manually figure out a series of waypoints and then use IK for each of them to create a collision-free trajectory.

8. It is the course staffs' decision to decide the appropriate tier for your method in case the tier for your method is not clear or doesn't match up with our suggested tiers. This division of points and tier-wise points are subject to minor change based on various factors like class progress etc.

9. You can also see the total points division on the lower-right side of the above image. The final project is worth 100 points (See the syllabus for the project's contribution towards your grade). There are extra bonus 40 points available (32 from higher tiers and 8 from 2 specially placed objects).

   The final **deliverables** are the **final version of the report, code** and **a demo video**. Check the Canvas "Final Project Submission" assignment to see this in detail.

Late submissions cannot be accepted due to the timing of when grades are due.