CSE 571 Research Project

Project Description

The open-ended final project may be either an implementation of a state-of-the-art algorithm or an algorithmic contribution that extends the current state of the art. Projects must include an evaluation component using a simulator or a real robot. Teams are encouraged to develop their own project ideas, but these pre-approved project suggestions may be helpful to get started. Teams should be consist of three students.

Deliverables

Please submit deliverables in groups via Gradescope.

Proposal (5%) and Progress Report (5%). In the project proposal (up to two pages), students will describe an open-ended research question that they wish to explore, consider some prior work in the field, and discuss their proposed approach. Students must select a robot platform (real or simulated) to experimentally validate their approach. As part of the proposal, teams will outline project milestones and set a timeline for their progress. The course staff will either approve the proposal or request changes. In the progress report, teams will report on successes and unforeseen problems, and provide updated timelines.

Poster Session (40%). The poster session will begin with a series of lightning talks: each team will introduce their problem and robot platform, discuss their approach, and illustrate an interesting result. The students must prepare videos of their projects. Each team is allocated 1 minute of speaking time per team member. After the lightning talks, the instructors will visit each poster to discuss the project in more depth. Teams are encouraged to bring a live demo!

Final Report (50%). The final report will summarize the research project in a three-page extended abstract (page count excludes references).

Timeline

- Proposal, due Monday, Jan. 28
- Progress report, due Thursday, Feb. 28
- Poster session, Thursday, Mar. 21 2:30-4:20pm
- Final report, due Friday, Mar. 22

Suggested Project Ideas

State Estimation and Navigation

- Use a depth or color camera to detect and track an object moving quickly, e.g. a ball bouncing off the ground or a wall. This makes for a good implementation of a combination of particle and Kalman filters.
- Localize with a depth camera (e.g. with a particle filter) in a given floor plan. We have existing laser maps and floor plans of the Allen Center.
- Create an exploration strategy for finding new views of unseen areas of a 2D/3D map. Investigate reconstruction and rendering techniques for such maps. Given an existing map, figure out how to perform probabilistic localization and filtering.
- Develop a navigation system for a powered wheelchair. This involves mounting a sensor on the wheelchair and developing a localization framework so that the wheelchair can autonomously navigate given a map.

Human-Robot Interaction

- Have a robot follow a person. Additionally, using a second camera/Kinect, you could enable a person to point at objects or goal locations that the robot needs to plan to navigate to.
- Implement visual servoing on a manipulator. This involves perception, planning and control. This video shows a preliminary implementation in a food manipulation task. Servoing has to be safe and robust to movements in the person's face to feed accurately and be robust to face rotations and partial face occlusions.
- Implement a simple human-robot handover system on a manipulator. The robot has to track the person, potentially predict where the person will move to and plan accordingly to receive the object from the person's hand while ensuring that there are no collisions and the motion is smooth. This involves perception and planning.

Manipulation

- Implement a simple pick and place system on a manipulator. The robot can pick up an object from a table and place it at a different location. This involves some minor perception for object detection, grasp planning and motion planning.
- Implement a fast local collision avoidance system (in simulation) that reacts to moving objects and people in the scene. There are multiple

methods to speed up the computation (including GPU-based techniques). Find a related paper here.

- Create a benchmark of motion planning algorithms on the manipulator (mainly in simulation). There are many new motion planning algorithms (BIT*, RRT*, TrajOpt, CHOMP, etc.) that can be implemented. OMPL has multiple benchmarking tools available. You can also consider creating your own benchmarking environments of varying difficulty (for instance, in terms of clutter in the manipulator's environment).
- Implement a handover task between two arms of a robot. This can involve picking up an object with one hand and transferring it to another before placing the object at a different location.
- Implement a teleoperation system for a robot manipulator. This can involve a virtual reality controller or some other input device.

Reinforcement Learning

- OpenAI Gym has several robotics and continuous control environments implemented using physics simulators (MuJoCo, Box2D). You can design a new reinforcement learning algorithm (or re-implement an existing one) for these environments. Other robots may also have open-source MuJoCo simulation code that you can use, e.g. the bipedal robot Cassie.
- Deep Q-Networks (DQNs) have been successfully applied to Atari games. Implement the DQN for a simple reinforcement learning task, e.g. cartpole or double pendulum. You can even look at some of the Atari games or other similar game worlds, or try learning directly from pixels rather than the physical state.

Robots and Other Hardware

Robots

- HERB: A bimanual mobile manipulation platform.
- ADA: Single arm manipulation platform mounted on a powered wheelchair.
- RACECARs: Fleet of cars equipped with sensors to allow autonomy.
- Kuri: Social robot.

Note that there is limited TA support for using these robots, so it is recommended to use a platform that one of your teammates has access to. If you want to use one of the platforms above, you are encouraged to meet with the TAs to discuss the availability of the robot, as well as what kind of support is available for working with them. These platforms generally have documentation that will be made available to you, and there may be people external to the course (in addition to the TAs) who can provide some guidance and help you get familiar with the software stack.

Other Hardware

In addition to the robots above, we have access to some standalone RGB-D cameras that you can use for your projects.