EE P 545
The Self Driving Car: Intro to AI for Mobile Robots

Course Instructor: Patrick Lancaster
Email: planc509@cs.washington.edu
Office Hours: TBD

Course Overview: In this course, we will cover topics related to state estimation (particle filters, motion models, sensor models etc), planning/control (search based planners, lattice based planners, trajectory following techniques etc), and perception and learning (object detection, learning from demonstrations etc.). Each of the 4-5 assignments will involve student teams implementing the algorithms learned in lecture on 1/10th sized rally cars. Concepts from all of the assignments will culminate into a partially open-ended final project with a final demo on the rally cars. The course will involve programming in a Linux and Python environment along with ROS for interfacing to the robot.

Course Goals: By the end of this course, students will:

- Program real robot platforms using ROS and Python
- Describe and implement algorithms from state estimation, planning/control, perception, and learning
- Describe how each of these algorithms contributes to the development of autonomous vehicles
- Implement these algorithms on a real robot platform
- Analyze both the theoretical and practical strengths and weaknesses of these algorithms

Grading (Tentative): 60% Assignments, 40% Final Project

Assignments: The course will consist of the following five assignments:

1. **Getting Started:** Introduction to Python, Numpy, ROS, and the robot
2. **State Estimation:** Localization with a particle filter
3. **Visual Servoing:** Line following / Move to object using images
4. **Planning:** Navigation in a known map, integration with state-estimation for closed-loop control
5. **Learning:** Learning robot controls from demonstrations

In addition, a partially open-ended final project will combine concepts from all of these assignments in order to autonomously navigate a track as quickly as possible.

Textbook: There is no required textbook for this course

Prerequisites:
- Proficiency in coding in a procedural language (e.g. C, C++, Python, Java, etc) is required
- Knowledge of basic probability is required
- Experience with Python is recommended