Instructor:  
Prof. Marco Pavone  
Office: Durand 261  
Email: pavone@stanford.edu  

Course Assistants:  
Somrita Banerjee  
Email: somrita@stanford.edu  
Abhishek Cauligi  
Email: acauligi@stanford.edu  
Boris Ivanovic  
Email: borisi@stanford.edu  
Mengxi Li  
Email: mengxili@stanford.edu  
Joseph Lorenzetti  
Email: jlorenze@stanford.edu  

Location and time: Remote: Synchronous, Tuesday and Thursday, 10:30am – 11:50am;  
Friday 10:00am - 11:20am. Friday lectures are optional for those enrolled in AA 174A. For  
all others, the homework will require material taught during the Friday lectures.  

Course format: This course will be taught in a “flipped classroom” format. We will  
release pre-recorded lecture videos for each week on the preceding Friday at  
https://canvas.stanford.edu/courses/123351  
In class, we will focus on summarizing and highlighting key concepts, providing examples that will further the understanding of course content and help with the homework, and answering questions. Questions can be submitted prior to the lecture through the Google Form  
https://forms.gle/8smGsRR6eiYcqsjfl7  
by 7pm the day before the lecture or asked live. Students should watch the pre-recorded videos before attending each scheduled lecture.  

Office hours:  
Prof. Pavone: Tuesdays, 1:00 – 2:00pm (Remote: Synchronous), and by appointment.  
Course assistants: Tuesdays, 2:00 – 4:00pm, and Thursdays, 4:00 – 6:00pm, Remote:  
Synchronous.
Sections:

- Monday, 10:30am – 12:30pm, acauligi, Remote: Synchronous.
- Monday, 3:00pm – 5:00pm, jlorenze, Remote: Synchronous.
- Tuesday, 4:00pm – 6:00pm, borisi, Remote: Synchronous.
- Wednesday, 10:30am – 12:30pm, somrita, Remote: Synchronous.
- Thursday, 2:00pm – 4:00pm, mengxili, Remote: Synchronous.

Units

- AA174A: 5.
- AA 274A/CS 237A/EE 260A: 3 or 4. Taking this class for 4 units entails additionally completing a paper review at the end of the quarter.

Prerequisites:

- Familiarity with programming (e.g., CS 106A or equivalent). Previous experience with Python would be helpful, but is not required.
- College calculus, linear algebra (e.g., CME 100 or equivalent).
- Basic probability and statistics (e.g., CME 106 or equivalent).

Course websites:

- For course content and announcements: http://asl.stanford.edu/aa274a/
- For course-related questions: https://piazza.com/stanford/fall2020/aa274a
- For homework submissions: https://www.gradescope.com/courses/175144
- For lecture videos: https://canvas.stanford.edu/courses/123351
- For urgent questions: aa274a-aut2021-staff@lists.stanford.edu
- To submit pre-lecture questions: https://forms.gle/8snGsRR6eiYcqsjf7

Textbooks: There is no required textbook. Recommended reading material:

Course content: Basic principles for endowing mobile autonomous robots with planning, perception, and decision-making capabilities. Algorithmic approaches for trajectory optimization; robot motion planning; robot perception, localization, and simultaneous localization and mapping (SLAM); state machines; introduction to methodologies for reasoning under uncertainty, e.g., (partially observable) Markov decision processes. Extensive use of the Robot Operating System (ROS) for demonstrations and hands-on activities.

Course goals: To learn the theoretical, algorithmic, and implementation aspects of main techniques for robot autonomy, in particular modeling & controls, motion planning, perception, localization & SLAM, state machines, and decision making. To learn how to apply such techniques in applications and research work by leveraging the Robot Operating System (ROS). With this class, the student will:

- Gain a fundamental knowledge of the “autonomy stack” behind self-driving cars, drones, and mobile autonomous robots in general;
- Be able to apply such knowledge in applications and research work by using ROS;
- Devise novel methods and algorithms for robot autonomy.

Course structure and homework policy: The class comprises four modules, roughly of equal length, namely:

1. motion control and planning (09/15 – 10/02);
2. robotic perception (10/06 – 10/16);
3. localization and SLAM (10/20 – 11/05);
4. state machines, decision making, and system architecture (11/10 – 11/19).

There will be a total of four problem sets. Rules:

- Because of the multiple topics that will be pursued in the course, it is important to keep up with the assignments. To account for unforeseen extraordinary circumstances, students are given a total of 6 free late days that may be used for the homeworks; a maximum of 3 late days will be allowed on a given assignment.

- Cooperation is allowed in doing the homework. You are encouraged to discuss approaches to solving homework problems with your classmates, however you must always prepare the solutions on your own. You must write on your problem set the names of the classmates you worked with. Copying solutions, in whole or in part, from other students or any other source will be considered a case of academic dishonesty.

- Homework submissions must be typeset (e.g., in LaTeX or Word.)

Sections: In addition to lectures, students are expected to sign up for a 2 hour section time that they will attend once per week. These sections will provide a chance for students to work on hands-on exercises that complement the lecture material and develop skills necessary for the final project. Part of your grade will come from attending, participating in, and completing tasks in your section each week. You must sign up for a section time using the link posted on the course website by 09/18/20. These sign ups are first-come first-serve.

Participation on Piazza: Piazza will be the main tool for class discussion. A student will get an extra point each time he/she (1) asks a question about lecture material; (2) answers a question about lecture material; or (3) answers a question about homework. Questions or answers should be endorsed by one of the CAs in order to receive credit. A student can accrue a maximum of five extra points. Additional details will be provided in a pinned note on Piazza.

Final project: For the final project, students will be assigned the task of deploying the autonomy stack incrementally built through the problem sets on a TurtleBot robot within a simulation environment (namely, Gazebo), with the ultimate goal of enabling self-driving capabilities in a mock urban environment. This task will involve combining all the skills the students have learnt from the class. More details about the final project will be given during the week of October 26, 2020.

Course grade calculation:

- (20%) final project.
- (60%) homework.
- (20%) sections.
- (extra 5%) participation on Piazza.
Schedule:
* lectures for AA 274A / CS 237A / EE 260A only (optional for AA 174A)

<table>
<thead>
<tr>
<th>Date</th>
<th>Topic</th>
<th>Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>09/15</td>
<td>Course overview, mobile robot kinematics</td>
<td></td>
</tr>
<tr>
<td>09/17</td>
<td>Introduction to the Robot Operating System (ROS)</td>
<td></td>
</tr>
<tr>
<td>09/18</td>
<td>* No lecture</td>
<td>HW1 out</td>
</tr>
<tr>
<td>09/22</td>
<td>Trajectory optimization</td>
<td></td>
</tr>
<tr>
<td>09/24</td>
<td>Trajectory tracking &amp; closed loop control</td>
<td></td>
</tr>
<tr>
<td>09/25</td>
<td>* Advanced methods for trajectory optimization</td>
<td></td>
</tr>
<tr>
<td>09/29</td>
<td>Motion planning I: graph search methods</td>
<td>HW1 due, HW2 out</td>
</tr>
<tr>
<td>10/01</td>
<td>Motion planning II: sampling-based methods</td>
<td></td>
</tr>
<tr>
<td>10/02</td>
<td>* No lecture</td>
<td></td>
</tr>
<tr>
<td>10/06</td>
<td>Robotic sensors &amp; introduction to computer vision</td>
<td></td>
</tr>
<tr>
<td>10/08</td>
<td>Camera models &amp; camera calibration</td>
<td></td>
</tr>
<tr>
<td>10/09</td>
<td>* Stereo vision</td>
<td>HW2 due, HW3 out</td>
</tr>
<tr>
<td>10/13</td>
<td>Image processing, feature detection &amp; description</td>
<td></td>
</tr>
<tr>
<td>10/15</td>
<td>Information extraction &amp; classic visual recognition</td>
<td></td>
</tr>
<tr>
<td>10/16</td>
<td>* Modern robotic perception</td>
<td></td>
</tr>
<tr>
<td>10/20</td>
<td>Intro to localization &amp; filtering theory</td>
<td>HW3 due, HW4 out</td>
</tr>
<tr>
<td>10/22</td>
<td>Parameteric filtering (KF, EKF, UKF)</td>
<td></td>
</tr>
<tr>
<td>10/23</td>
<td>* Nonparameteric filtering (PF)</td>
<td></td>
</tr>
<tr>
<td>10/27</td>
<td>EKF localization</td>
<td>Final project released</td>
</tr>
<tr>
<td>10/29</td>
<td>EKF SLAM</td>
<td></td>
</tr>
<tr>
<td>10/30</td>
<td>* Monte Carlo localization and particle filter SLAM</td>
<td></td>
</tr>
<tr>
<td>11/03</td>
<td>No lecture (Election Day)</td>
<td></td>
</tr>
<tr>
<td>11/05</td>
<td>Multi-sensor perception &amp; sensor fusion</td>
<td></td>
</tr>
<tr>
<td>11/06</td>
<td>* No lecture</td>
<td></td>
</tr>
<tr>
<td>11/10</td>
<td>Software for autonomous systems</td>
<td>HW4 due</td>
</tr>
<tr>
<td>11/12</td>
<td>State machines</td>
<td></td>
</tr>
<tr>
<td>11/13</td>
<td>* No lecture</td>
<td>Final project check-in</td>
</tr>
<tr>
<td>11/17</td>
<td>Decision making under uncertainty</td>
<td></td>
</tr>
<tr>
<td>11/19</td>
<td>Reinforcement learning</td>
<td></td>
</tr>
<tr>
<td>11/20</td>
<td><strong>Final Project Demo</strong></td>
<td></td>
</tr>
</tbody>
</table>
**Students with documented disabilities:** Students who may need an academic accommodation based on the impact of a disability must initiate the request with the Office of Accessible Education (OAE). Professional staff will evaluate the request with required documentation, recommend reasonable accommodations, and prepare an Accommodation Letter for faculty. Unless the student has a temporary disability, Accommodation letters are issued for the entire academic year. Students should contact the OAE as soon as possible since timely notice is needed to coordinate accommodations. The OAE is located at 563 Salvatierra Walk (phone: 723-1066, URL: [https://oae.stanford.edu/](https://oae.stanford.edu/)).