University at Buffalo
The State University of New York
Department of Computer Science and Engineering

CSE 468/568: Robotics Algorithms

TTh 9:30-10:50 am

TBD

Karthik Dantu
kdantu@buffalo.edu
http://www.cse.buffalo.edu/faculty/kdantu
Davis 331
Office Hours: TBD
(716) 645-2670

Course Description: This course is intended to be a comprehensive introduction to robotics algorithms for a senior undergraduate/first-year graduate student. It is a Computer Science course, and introduces the student to well-known algorithms in making a simple robot autonomous.

Prerequisite(s): Engineering Probability, Data Structures, and Programming in C++/Python/Java. Knowledge of linear algebra is a plus.

Credit Hours: 4 (468)/ 3 (568)

Text(s): *Introduction to Autonomous Mobile Robots*, 2nd Edition
Author(s): Ilah Nourbaksh, Ronald Seigward, Davide Scaramuzza
ISBN-13: 978-0262015356
Supplementary text: *Probabilistic Robotics*
Author(s): Sebastian Thrun, Wolfram Burgard, Dieter Fox
ISBN-13: 978-0262201629

Course Objectives:

At the completion of this course, students will be able to:

1. Mobility: Wheeled locomotion (2.3-2.3.1.5, p35-42)
2. Kinematics: Wheeled kinematics, coordinate transforms (3-3.2.3.2, p57-66, or from notes)
3. Sensing: Sensing, sensor classification and sensor characterization (4-4.1.2.3)
4. Sensing: Various types of sensors (4.1.4-4.1.11, p115-142)
5. Sensing: Range sensing and line fitting algorithms (4.7.1-4.7.2.8, p243-258), range histograms (4.7.3, p259-262)
6. Sensing: Vision, image formation, camera model, perspective projection (4.2-4.2.3.4, p142-159)
7. Sensing: Image processing (4.3-4.5.4.2, p195-225), SIFT (4.5.5.1, p227-232)
8. Sensing: Structure from stereo (4.2.5-4.2.6.1, p169-187)
9. Estimation: Probabilistic modeling of robot localization and mapping
10. Estimation: General Bayes' formulation of reasoning under uncertain conditions
11. Estimation: Specific formulations of localization and mapping
12. Planning: Given a map, understand how a robot can navigate from a location to another location
13. Planning: Randomized algorithms for planning

14. Task Allocation: In multi-robot systems, study algorithms that allow robots to perform task allocation
15. Applications: Study full multi-robot systems in deployment/research that combine all the algorithms learnt in this class and beyond

While this is a reasonable outline, the instructor reserves the right to change the syllabus as he sees fit. Please check the course site for the latest changes.

Grade Distribution:

Programming Assignments/Homeworks	50%
Midterm Exam	20%
Final Exam	20%
Class and Piazza partcipation	10%

Grades will be on a curve. From prior experience, I have seen different performance from the undergraduate (468) and graduate (568) students. Therefore, for fairness, I'll grade students on two separate curves, one for graduate students and another for the undergraduate students.

Course Policies:

- **General**
  - Computers are not to be used unless instructed to do so.
  - Exams are closed book, closed notes.
  - **No makeup quizzes or exams will be given unless discussed on a per case basis well in advance of the exam.**
- **Grades**
  - Graded on a curve with B+ roughly being the median.
  - Grades will be maintained in the myUB course system. Students are responsible for tracking their progress by referring to the online gradebook and report any discrepancies.
- **Labs and Assignments**
  - Students are expected to work independently. **Offering** and **accepting** solutions from others is an act of **plagiarism**, which is a serious offense and **all involved parties** will be penalized according to the Academic Honesty Policy. Discussion amongst students is encouraged, but when in doubt, direct your questions to the professor, or teaching assistant.
  - Unless addressed with the professor well ahead of a deadline, late submissions will be penalized. 10% penalty for submissions that are 0-24 hrs late; 25% penalty for submissions that are 24-48hr late; 50% for submissions that are 48-72 hr late and 100% after.
- **Attendance and Absences**
  - Attendance is expected each class. Participation in class and on piazza carries 10% of the grade. The professor will not entertain any grade changes toward the end of the course if the student has not participated during the semester.
  - Students are responsible for all missed work, regardless of the reason for absence. It is also the absentee's responsibility to get all missing notes or materials.

Plagiarism Policy:

- This course has several programming assignments mostly based on the ROS programming environment. We use **sophisticated code checkers** to check for code copied from assignments from this class as well as submissions from prior editions of the class. It is **very improbable** that you will be able to fool the code checker.
- Along the same lines, if you see someone else's code to understand the logic, it is probable that our code checker will flag this as plagiarized since your code will be influenced by what you saw and will look structurally similar.
- Discussing programming logic with other students is acceptable. However, this is a slippery slope, and the more detailed these discussions are and the more they tend toward actual code, the more likely the code checker will flag this as plagiarism.
- Copying snippets of code from online resources is also considered plagiarism. When you are in doubt, please check with the professor for clarity.