Syllabus: Applied Robotics

Jeffrey Yepez

University of Hawai'i Maui College
Science, Technology, Engineering and Mathematics Department
310 Ka'ahumanu Ave, Kahului, HI 96732-1617
yepez@hawaii.edu

(Dated: August 21, 2017)

I. COURSE DESCRIPTION

This is a 3 credit one-semester course on robotics.

A. Overview

The course introduces robotics-related technologies, including computer programming methodologies, data acquisition methods for sensors (such as infrared and optical imagers) and control methods for actuators and servo motors and microcontrollers. The subject of robotics is treated as an interdisciplinary engineering subject that includes mechanical engineering, electrical engineering, optical engineering, control theory and computer science engineering. The course addresses advanced robotic topics, including autonomous control, machine learning and applied artificial intelligence. Using a hands-on approach to applied robotics, the students in this course write their own controller programs and build their own robot prototypes based on standard microcontrollers (including the Arduino, Lego Mindstorms NXT and EV3 Bricks, and Brick Pi).

The course material includes an introduction to programming based on Scientific Python and Mathematica. The course introduces graphical processing unit (GPU) programming for achieving high-performance computing for robotic tasks and also introduces the topics of neural networks and machine-learning algorithms.

The course introduces numerical methods for object detection, classification and tracking. Finally, the course provides laboratory hands-on applications of the concepts and theories presented throughout the semester.

B. Course plan

Students should have facility with algebra and have familiarity with a high-level functional programming language, such as C or C++. Familiarity with (as well as a willingness to apply) differential and integral calculus is helpful.

Presented is an introduction to scientific programming based on the Python language applied to data acquisition and control of robotic subsystems. Demonstrations in the class will use symbolic mathematics (including Mathematica and SciPy) and the use of a symbolic mathematics software application is highly recommended for the students as an aid to accomplishing the assigned problem sets and robotic algorithm designs.

Each student will select an applied robotics topic of their choosing for a Final Project.

The Final Project will entail:

1. designing and building a digital/analog robotic system for testing
2. writing a controller program for an electromechanical robotic with an optical and/or ultrasonic subsystem
3. implementing and testing the performance of the robotic system in achieving its design goal
4. explaining the tools and engineering methods used to construct and test the robotic system
5. and comparing predictions about the robotic system's expected behavior with the observed behavior during testing.

Each student will summarize his or her robotic system project in written Technical Report form for turn in as a final report.

Basic reviews/refreshers for computer operating system and scientific programming are provided at the outset as needed.

The class meets Monday 12:15-2:30pm and Wednesday 12:15-2:30pm (via VTC as needed)

Office Hours: Fridays 12pm and by appointment

Prerequisites: ETRO 212 and ICS 111 with grade C or better. — This prerequisite cannot be waived.

C. Mathematical skill set

Algebra, functional computer programming and linear electrical circuit theory.

II. PROSPECTUS

1. Introduction
    - Introduction to applied robotics, the course outline and programming foundations
2. Computer operating system, scientific programming and reporting
    - Introduction to the linux operating system and the command-line language
    - Introduction to a program editor
    - Introduction to the Python language and Python libraries, including NumPy, SciPy and NXT Python
    - Introduction to numerical arrays and parallel arithmetic
    - Introduction to numerical data plotting
    - Introduction to technical report writing and typesetting
    - Introduction to numerical regression techniques
    - Installing Raspbian OS on the Raspberry Pi 3
3. System on a chip microcontrollers
    - Introduction to microcontrollers (32-bit ARM-based devices) in embedded applications used in automobiles and home appliances (such as washing machines, microwave ovens, telephones, and computer system peripherals)
    - Controlling GPIO pins (e.g. connected to LEDs) on the Raspberry Pi 3 using Python
    - Controlling motors
    - Collecting sensor data (such as light-color sensor, touch sensor, infrared proximity sensor and ultrasonic sensor)
    - Writing and uploading robotic control programs
4. Robotic actions and autonomous control algorithms
    - Robotic motion and autonomous responses
    - Path following, solving a Rubix cube, book scanning, and other fun problems
5. Midterm
    - Midterm exam (October)
    - In-class review of the solution set
6. General-purpose computing on graphics processing units (GPU computing)
    - Quad Processing Units (QPUs) on the Raspberry Pi 3
    - Compute Unified Device Architecture (CUDA) parallel computing platform and application programming interface model created by Nvidia
7. Neural networks and machine learning
    - Introduction to artificial intelligence (AI)
    - Hopfield neural networks and associative memory
    - Machine learning algorithms for neural network pattern recognition
8. Object detection and classification (time permitting)
    - Edge detection algorithms
    - Nonlinear diffusion and GPU implementations
    - Perception and symbolic representation of physical objects
9. Final
    - Final exam (mid December) and Student Project submitted as a Technical Report

III. PROBLEM SETS

Problem sets are assigned weekly and tailored to the classes progress. Student generally have one week to complete any given assigned problem set.

Selected Problem Sets will be submitted in Technical Report format using scientific typesetting software. Students will be provided with the scientific writing preparation and help.

IV. COURSE REQUIREMENTS AND EVALUATION

Problem sets, labs, written exercises, and computer robotic problems: 30%
(assigned Wednesday and due the following Wednesday)
Midterm examination: 30%
Final exam: 30%
Class participation: 10%

V. METHOD OF INSTRUCTION

- Internet-based video teleconference
- NumPy, SciPy and Mathematica demonstrations
- NXT Python demonstrations
- Class discussions and lab experiments
- Weekly office hours on Friday at 12pm or otherwise scheduled by e-mail request

VI. WIKIPEDIA

Material to applied robotics:

- Robotics
- Microcontroller
- Robot kit
- Open-source hardware
- Autonomous robot
- Cognitive robotics
- Mathematical optimization
- General-purpose computing on graphics processing units
- Artificial neural network
- Machine learning

VII. RESOURCES

- Additional reference text book:
  Ulrich Nehmzow, "Mobile Robotics: A Practical Introduction", 2nd Ed., Springer
- In class demonstrations: Python programs and Mathematica notebooks
- Additional resources and weekly updates will be provided in class.

VIII. DISABILITY STATEMENT

1. If you have a disability and have not voluntarily disclosed the nature of your disability and the support you need, you are invited to contact Lisa Deneen - Disabilities Coordinator at 984-3227 or Telecommunication Device for the Deaf (TDD) 984-3325 or the Text Telephone (TT) replay service at 643-8833.

2. Reasonable accommodations will be provided for students with documented physical, sensory, systemic, cognitive, learning and psychiatric disabilities. If you believe you have a disability requiring accommodations, please notify Lisa Deneen - Disabilities Coordinator at 984-3227 or Telecommunication Device for the Deaf (TDD) 984-3325 or the Text Telephone (TT) replay service at 643-8833. The Disabilities Coordinator will verify your disability and provide the course instructor with recommendations for appropriate accommodations.