UNSW AUSTRALIA

Course outline

Semester 1 2017

Never Stand Still | Engineering | Mechanical and Manufacturing Engineering

MTRN4110

ROBOT DESIGN

Contents

1. Staff contact details

Name: Dr Jose Guivant
Office: 510D, J17
Tel: (02) 9385 5693
Email: J.Guivant@unsw.edu.au

Consultation Times: To be agreed with students, before week 2.

2. Course details

Credit Points

This is a 6 unit-of-credit (UoC) course, and involves 5 hours per week (h/w) of face-to-face contact.

The UNSW website states "The normal workload expectations of a student are approximately 25 hours per semester for each UoC, including class contact hours, other learning activities, preparation and time spent on all assessable work. Thus, for a full-time enrolled student, the normal workload, averaged across the 16 weeks of teaching, study and examination periods, is about 37.5 hours per week."

This means that the student should aim to spend about 9 h/w on this course. The additional time should be spent in making sure that the students understand the lecture material, completing the set assignments, further reading, and revising for any examinations.

Contact hours

	Day	**Time**	**Location**
Lecture	Friday	09:00 – 11:00	Webster Theatre B (F Hall B)
Projects/ Lab (slot1)	Wednesday	09:00 – 12:00	Mechatronic Lab 212 (J18)
Projects/ Lab (slot2)	Wednesday	12:00 – 15:00	Mechatronic Lab 212 (J18)
Projects/ Lab (slot3)	Thursday	09:00 – 12:00	Mechatronic Lab 212 (J18)
Projects/ Lab (slot4)	Thursday	12:00 – 15:00	Mechatronic Lab 212 (J18)
Projects/ Lab (slot5)	Friday	11:00 – 14:00	Mechatronic Lab 212 (J18)
Projects/ Lab (slot6)	Friday	14:00 – 17:00	Mechatronic Lab 212 (J18)

Summary of the course

This course focuses on the design and implementation of the perception and control capabilities of an autonomous robotic platform. When we say "design and implementation of autonomous platforms" (in 2017, those are hexapod platforms), we are implicitly talking about the hardware (mechanical and electronic components) and the software (low and high level modules) for the tasks of perception and control of the robot. This year, 2017, we will focus on both parts; the Hardware and the Software components of the robot. For the hardware part, we focus on expanding the capabilities of already implemented platforms, by designing and implementing robotic arms to be installed on the current hexapods. In addition to that, the students (working in small teams of about four members) will install all the sensing capabilities in each of the robots, i.e. state of art 3D cameras, 6DoF inertial units, and other sensors (less impressive ones such as the usual ultrasonic and infrared range sensors), in addition, the servos used for the designed/implemented robotic arms.

The software component of the projects involves designing and implementing all the relevant modules, for making the platform intelligent enough for performing certain complicated tasks. The low level part will include the real time reading and preprocessing of sensors' data (3D camera, IMU, etc.) and the low level operation of the 18 servos of the platform's legs and the two additional ones for the robotic arm. The on-board CPUs will be small but powerful computers (multi core, like a good laptop), where your implementations in C/C++ will run. Having such available processing power, certain high level modules will be also implemented, such as for performing inverse kinematic (making the platforms to walk), terrain modelling (based on 3D imagery), optimal path planning (based on 3D modeling) and many more. Communication with external processing nodes (your laptops or Lab's desktop PCs), via Wi-Fi, will allow additional processing (e.g. using Matlab) and interaction with human operators, when required.

For being able to properly implement many of those diverse components, we will need to apply some related theory, which will be presented in the lectures, in addition to relevant technical discussions (i.e. about sensors and components, manuals, communication protocols, programming approaches, electronic circuits, etc.).

For those reasons, half of the course is lecture-based. In the other half, the students will put in practice those concepts.

Aims of the course

At the conclusion of this course, it is expected that you will be able to:

- Improve skills in programming (C/C++ and Matlab);
- Improve skills in applying mathematics and programming concepts for solving problems related to processing and control.
- Acquire experience working with a diversity of sensors including concepts for processing sensors' measurements in a real-time fashion and many other concepts needed for implementing the robot's perception and control capabilities.
- Understand the multiple components of an intelligent robotic system.
- Understand and apply specific concepts such as kinematic models, localization, mapping and certain advanced control techniques (Dynamic Programming, MPC) which can be used in diverse areas of application.
- Get experience working with real components (sensors, actuators, batteries, protection circuits, mechanical components, analog inputs/outputs, digital Inputs/outputs, communication links, etc.)

Concepts included in this course are useful for other disciplines, in research, development and industrial application.

Student learning outcomes

This course is designed to address the below learning outcomes and the corresponding Engineers Australia Stage 1 Competency Standards for Professional Engineers as shown. The full list of Stage 1 Competency Standards may be found in Appendix A.

After successfully completing this course, you should be able to:

Learning Outcome		EA Stage 1 Competencies
1.	Understanding of the general theory of Dynamic Programming (and specifically the Dijkstra's algorithm) and MPC for non-linear control problems.	PE1.1 Comprehensive, theory-based understanding of underpinning fundamentals
2.	Understanding the Kinematic models of wheeled and legged platforms. Understanding of 3D Attitude definition and estimation; triangulation and other localization approaches.	PE1.1 Comprehensive, theory-based understanding of underpinning fundamentals
3.	Be able to develop software for applying the theory and actually solving complex problems. Get experience in using state of the art sensors and actuators, used in Robotics and Autonomous Systems (digital servos, RGB -Depth cameras, inertial systems and diverse usual low cost sensors)	PE2.3 Application of systematic engineering synthesis and design processes

3. Teaching strategies

Teaching of this course is through lectures (to cover the theory) and laboratory and project sessions to put the concepts in practice. Certain parts of the projects are individual tasks; other parts are solved in groups. In any case, attendance is necessary.

The provision of the learning environment in the laboratory is to facilitate you to develop confidence in managing laboratory tasks as projects. Demonstrators in the laboratories are there to provide you all the guidance and assistance is managing the laboratory tasks and projects.

Example source code for parts of the projects is provided, in order to help in the understanding and full implementation of the projects. Manuals and technical details of the components, used in the projects, are explained in the labs and during part of the lectures. Except certain minor materials, all the components are provided to the students by the course (sensors, actuators, computers, electronic components, batteries, basic platforms, etc.).

Complexity of the projects is incremental, in order to allow the student to finally complete the solution of a complex problem.

Access to these real sensors and platforms is a relevant feature of this course, to gain experience for future industrial and research activities.

4. Course schedule

Topic	**Date**	**XY***	**Lecture Content**	**Suggested Readings**
Introduction / sensors used in MTRN4110	week 1	LR	Introduction: about course and platforms (Hexapods). Sensors used in 4110: Inertial unit (IMU), 3D cameras, Ultrasonic, Infrared, etc. Actuators: Servos Interfacing with sensors and actuators. Communication Protocols. Real-time demos in class.	Moodle lecture notes
3D attitude Estimation.	week 2	LR	3D attitude. 3D rotations. 3D coordinate frames. Integrating gyroscopes for estimating 3D attitude. Calibrating IMU's gyros. Implementing related processing in a computer. Examples and real-time demos in class.	Moodle lecture notes
Kinematic Models	week 3	LR	2D Kinematic models of wheeled platforms. 3D Kinematic models of legged platforms. Inverse Kinematic for controlling platforms. Examples and demos in Real Time, in class.	Moodle lecture notes
Robots under limited resources.	week 4	LR	Electronic circuits for protecting and monitoring the platform's health and energy. Energy and power concepts. Batteries. Understanding relevant information in manuals. Communications resources. Sharing data among robots, and with processing nodes. Data rates / bandwidth. Estimating necessary bandwidth.	Moodle lecture notes

2D and 3D perception	weeks 5,6	LR	Raw 3D data from sensors: RGB-D cameras. Using our cameras in our projects. Processing data: Detecting usual 3D indoor flat surfaces (floor, walls). Detecting 2D features. Applying well known Mathematics: Least squares. Singular Value Decomposition (SVD) Feature Extraction from 2D and 3D imagery. Tracking OOIs (Objects of Interest). Data association. Hough Transform in 2D	Moodle lecture notes
Localization	Week 7	LR	2D (3DoF) Localization of the platform. Dead Reckoning. Triangulation. 3D attitude estimation based on accelerometers and 3D imagery.	Moodle lecture notes
Modelling context	week 8	LR	Terrain modelling from 2D and 3D perception. Discrete representation. Occupancy Grids.	Moodle lecture notes
Control and Planning	week 9	LR	Basic control. Optimal control/planners: Dynamic Programming (Bellman Principle of Optimality). Dijkstra’s approach, Model Predictive Control (MPC)	Moodle lecture notes
Combining Perception and Control.	week 10	LR	Discussing about final projects. Examples and demos in class.	Moodle lecture notes
Discussion / Future work	week 11	LR	State of the art. Papers about Robotics..	Moodle lecture notes
Discussion / Future work	week 12	LR	Revision / Consultation about everything (Final project). Future work.	Moodle lecture notes
Optional Lecture	week 13	LR	Revision / Discussion (if students request it) or Contingency extra time.	Moodle lecture notes

*Note: LR = lecture Room = Webster Theatre B (F Hall B)

5. Assessment

Assessment Overview

Assessment task	Length	Weight	Learning outcomes assessed	Assessment criteria	Due date, time, and submission requirements	Deadline for Absolute fail	Marks returned
Projects	4 projects	54%	1,3	Refer to assignments specifications for exact details.	See details in the section about Projects	See details in the section about Projects	See details in the section about Projects
Final exam	2 hours	46%	1,2,3	All course content from weeks 1-12	Exam period, date TBC.	N/A	Upon release of final results

Necessary conditions in order to pass the course:

a) The exam mark must be 50/100 or higher.

b) The total mark of the project component must be 50/100 or higher.

Projects

Assessment task	Length	Weight (of Project component)	Learning outcomes assessed	Assessment criteria	Due date, time, and submission requirements	Deadline for Absolute fail	Marks returned
Task 1	Completely operational software and hardware	12%	3	Refer to assignment specification for exact details (*).	Meeting with a demonstrator during week 4.	1 week later	< 10 days later
Task 2	Completely operational software	12%	3	Refer to assignment specification for exact details (*).	Meeting with a demonstrator during week 6.	1 week later	< 10 days later
Project 1	Completely operational software	15%	1,3	Refer to assignment specification for exact details (*)	Meeting with a demonstrator during week 9.	1 week later	< 10 days later
Project 2	Completely operational software	15%	1,3	Refer to assignment specification for exact details (*).	Meeting with a demonstrator during week 12 or 13.	First Monday after week 13. (aka “week14”)	By Friday “week 14”, 11:30PM

(*) Provided via Moodle; 2 weeks before the official release of the project.

Assignments
All hardware and software implementations, and results, must be explained to your demonstrator. A significant portion of the marks are the result of your knowledge demonstration, during your meeting with the demonstrator.

A short quiz (for all the students in a lab session), before the demonstration, is usually required by the demonstrators. In such cases, the quiz would commence 10 minutes past the nominal starting time of the lab/project session. Students who are not able to attend a demonstration session must apply for special consideration.

At the end of the demonstrations, you will submit parts of your software and reports (if required, by a date which will be defined in the project release) in a zipped file, via a Moodle submission site. Details about the format and name convention for programs files and reports will be specified with the release of the tasks and projects.

Presentation
All submissions are expected to be neat and clearly set out. Your results are the pinnacle of all your hard work and should be treated with due respect. Presenting results clearly gives the marker the best chance of understanding your method; even if the numerical results are incorrect.

Submission
Late submissions will be penalised up to 5 marks per calendar day (including weekends). An extension may only be granted in exceptional circumstances. Where an assessment task is worth less than 20% of the total course mark and you have a compelling reason for being unable to submit your work on time, you must seek approval for an extension from the course convenor ***before the due date***. Special consideration for assessment tasks of 20% or greater must be processed through student.unsw.edu.au/special-consideration.

It is always worth submitting late assessment tasks when possible. Completion of the work, even late, may be taken into account in cases of special consideration.

Where there is no special consideration granted, the 'deadline for absolute fail' in the table above indicates the time after which a submitted assignment will not be marked, and will achieve a score of zero for the purpose of determining overall grade in the course.

Marking
Marking guidelines for assignment submissions will be provided at the same time as assignment details to assist with meeting assessable requirements. Submissions will be marked according to the marking guidelines provided.

Examinations
You must be available for all tests and examinations. Final examinations for each course are held during the University examination periods, which are June for Semester 1 and November for Semester 2.

Provisional Examination timetables are generally published on myUNSW in May, for Semester 1.

For further information on exams, please see the Exams section on the intranet.
The exam's duration is two (2) hours. It involves substantial part of the theory (presented in the lectures) and also questions about the projects that were solved by the students.

Calculators
You will need to provide your own calculator, of a make and model approved by UNSW, for the examinations. The list of approved calculators is shown at
student.unsw.edu.au/exam-approved-calculators-and-computers

It is your responsibility to ensure that your calculator is of an approved make and model, and to obtain an "Approved" sticker for it from the School Office or the Engineering Student Centre prior to the examination. Calculators not bearing an "Approved" sticker will not be allowed into the examination room.

Special consideration and supplementary assessment
For details of applying for special consideration and conditions for the award of supplementary assessment, see the School intranet, and the information on UNSW's Special Consideration page.

6. Expected resources for students

All the academic material is provided by the lecturer (lecture notes, example data, software libraries, example code, electronic components, sensors and equipment, certain tools). Some minor components may be required to be provided by the students.

In addition to the real-time data provided by the sensors, datasets of typical measurements are provided for allowing the students to perform play-back sessions and work at home when needed, for refining/debugging programs for processing tasks.

If you wish to explore any of the lecture topics in more depth, then other resources are available and assistance may be obtained from the UNSW Library.

One starting point for assistance is: https://www.library.unsw.edu.au/

7. Course evaluation and development

Feedback on the course is gathered periodically using various means, including the myExperience process, informal discussion in the final class for the course, Moodle's forums and the School's Student/Staff meetings. Your feedback is taken seriously, and continual improvements are made to the course based, in part, on such feedback.

In this course, recent improvements resulting from student feedback include:

- Included complex robotics projects, using real platforms, developing real implementations in the Hardware components.

8. Academic honesty and plagiarism

UNSW has an ongoing commitment to fostering a culture of learning informed by academic integrity. All UNSW students have a responsibility to adhere to this principle of academic integrity. Plagiarism undermines academic integrity and is not tolerated at UNSW. *Plagiarism at UNSW is defined as using the words or ideas of others and passing them off as your own.*

Plagiarism is a type of intellectual theft. It can take many forms, from deliberate cheating to accidentally copying from a source without acknowledgement. UNSW has produced a website with a wealth of resources to support students to understand and avoid plagiarism: student.unsw.edu.au/plagiarism The Learning Centre assists students with understanding academic integrity and how not to plagiarise. They also hold workshops and can help students one-on-one.

You are also reminded that careful time management is an important part of study and one of the identified causes of plagiarism is poor time management. Students should allow sufficient time for research, drafting and the proper referencing of sources in preparing all assessment tasks.

If plagiarism is found in your work when you are in first year, your lecturer will offer you assistance to improve your academic skills. They may ask you to look at some online resources, attend the Learning Centre, or sometimes resubmit your work with the problem fixed. However more serious instances in first year, such as stealing another student's work or paying someone to do your work, may be investigated under the Student Misconduct Procedures.

Repeated plagiarism (even in first year), plagiarism after first year, or serious instances, may also be investigated under the Student Misconduct Procedures. The penalties under the procedures can include a reduction in marks, failing a course or for the most serious matters (like plagiarism in an honours thesis) even suspension from the university. The Student Misconduct Procedures are available here:
www.gs.unsw.edu.au/policy/documents/studentmisconductprocedures.pdf

Further information on School policy and procedures in the event of plagiarism is available on the intranet.

Although students are encouraged to discuss concepts with colleagues, the work in the projects is INDIVIDUAL. Programs and reports are individual work.

9. Administrative matters

All students are expected to read and be familiar with School guidelines and polices, available on the intranet. In particular, students should be familiar with the following:

- Attendance, Participation and Class Etiquette
- UNSW Email Address

- Computing Facilities
- Assessment Matters (including guidelines for assignments, exams and special consideration)
- Academic Honesty and Plagiarism
- Student Equity and Disabilities Unit
- Health and Safety
- Student Support Services

Jose Guivant
15 February 2017

Appendix A: Engineers Australia (EA) Competencies

Stage 1 Competencies for Professional Engineers

	Program Intended Learning Outcomes
PE1: Knowledge and Skill Base	PE1.1 Comprehensive, theory-based understanding of underpinning fundamentals
	PE1.2 Conceptual understanding of underpinning maths, analysis, statistics, computing
	PE1.3 In-depth understanding of specialist bodies of knowledge
	PE1.4 Discernment of knowledge development and research directions
	PE1.5 Knowledge of engineering design practice
	PE1.6 Understanding of scope, principles, norms, accountabilities of sustainable engineering practice
PE2: Engineering Application Ability	PE2.1 Application of established engineering methods to complex problem solving
	PE2.2 Fluent application of engineering techniques, tools and resources
	PE2.3 Application of systematic engineering synthesis and design processes
	PE2.4 Application of systematic approaches to the conduct and management of engineering projects
PE3: Professional and Personal Attributes	PE3.1 Ethical conduct and professional accountability
	PE3.2 Effective oral and written communication (professional and lay domains)
	PE3.3 Creative, innovative and pro-active demeanour
	PE3.4 Professional use and management of information
	PE3.5 Orderly management of self, and professional conduct
	PE3.6 Effective team membership and team leadership