Stavanger, January 27, 2025

University of Stavanger

Faculty of Science and Technology

ELE610 Applied Robot Technology, V-2025

Information

Welcome to this course ELE610 Applied Robot Technology ↗.
The course started with an information meeting in week 2:
January 10th at 0900 in room E464.
Some information was given and the students were assigned to groups. The groups will be listed below in this document and on *canvas*. It is possible to do some changes until you start to deliver the work. Each group is assigned a workplace (named desk) in Laboratory E464.

If you were unable to attend at the information meeting there is no problem. You may contact me (e-mail) so I know you intend to participate, or just come to the laboratory Monday or Friday (around 0900) week 3. You may also consider who you want to be on group with, if you know or have made an agreement with another student. It will be possible to add persons to groups and to do changes on the groups during the course, but from week 5 and onward mainly by dissolving nonfunctional groups.

You should start (laboratory) work from week 3, even though first assignment is due week 5. The course continue until week 17, last assignment due week 15 for half of the students and week 17 for the rest, this because only one robot can be used for the final assignment. If you are unable to attend one or two weeks, it should be possible to finish the course anyway. Some of the work on each assignment can be done from anywhere, but also some work needs to be done in the laboratory.

We look forward to meet you.

Karl Skretting, Department of Electrical Engineering and Computer Science (IDE),
University of Stavanger (UiS), N-4036 Stavanger. E-mail: karl.skretting@uis.no.

1 ELE610

- This course has now two main parts; ABB robot with RobotStudio (RS) and Image Acquisition (IA). Each part has four or five assignments, and each assignment is scheduled for one week (15-20 hours), except RS5 which is scheduled for two weeks.
  - ABB-robot part uses RobotStudio and for assignment RS4 and RS5 also Python. Help and support for this part is given by Karl Skretting and Ståle Freyer.
  - The image acquisition (IA) part uses Python and several imported packages. Help and support for this part is given by Karl Skretting and Asbjørn Stokka (he administers the PCs on E46x).
- Course is 10 credit points. All assignments must be approved before course is finished. There is no exam. The grade is *passed* or *not passed*.
- Each assignment is approved when an acceptable solution is shown to Karl in the laboratory. Also, for each image acquisition assignment a report must be written (pdf-file). For ABB robot assignments it is sufficient to deliver the Robot Studio code and for RS4 and RS5 the Python-code as well. Each group deliver one report, or the code files, on *canvas*.
- Students works together in groups of two (or three) students.
- There are three video lectures available, see Modules in Canvas. These give a brief introduction to the `appImageViewer`-programs (Python Qt) used in the image acquisition assignments.
- The **core time** for this course is Mondays from 0815 to 1600 and Fridays from 0815 to 1600 (as given in TimeEdit, UiS timeplan program). Help will mainly be available in the core time. Each group should be present at the laboratories at least 4 hours of the core time.

  Karl will mainly be in room E464 on Mondays and Fridays before lunch (1200) and may be available other times in his office E438, just knock on the door.
- **Laboratory E458** (ABB robots) is needed for exercises RS2-RS5. This laboratory is generally available all week, but there is only space for two groups at the same time. The robots may be booked for slots of two hours, but with a maximum of two pending reservations. This is handled by our booking system UrLABS ↗.
- The plan for when each group should do each assignment will be shown below.

2 ELE610 Groups spring 2025

In the information meeting January 10th the students was divided into groups. The groups are named by the letter for the workplace, and you find them in *canvas* and below. Each group should be two (or one) students. This spring we will be in room E464, desk N to X.

Groups starting with RobotStudio-part RS1-RS4 weeks 3-7,
then doing Image Acquisition-part IA1-IA4 weeks 8-12 and weeks 13-14 for RS5 and weeks 15-17 for postponed assignments if any.

Desk N: Alexios Savvas Damaskinos and Georgios Giannatos

Desk O: Mahamed Qalbinuur Abdullahi and Tawite Noe Marthe

Desk P: Fabian Lichter and Elisa Julia Tscholakov

Desk Q: Chieu Ha Phuong Duong and Huy Vi Duc Ngo

Desk R: Áron Assani and Judith Zaragoza Lopez

Groups starting with Image Acquisition-part IA1-IA4 weeks 3-7,
then doing RobotStudio-part RS1-RS4 weeks 8-12 and weeks 13-14 for postponed assignments.

Desk S: Katharina Elisabeth Rein and Marzena Katarzyna Switon

Desk T: Raphaël Gauthier

Desk U: Ivanna Lavruk and Kamila Mosińska

Desk V: Théo De Morais and Jesus Sanchez Quiros

Desk W: Halvard Erlandsen and Live Sørbø

Desk X:

3 ABB-part

Some information on *canvas*. Note that the links to the assignments may not be ready yet. The updated assignments should be dated December 2024 or later.

For ABB RobotStudio the RobotStudio web page ↗ may be useful. The first assignment guide you through how to install RobotStudio and check that it is (correctly) installed.

1. Simulation
   - Web page with RAPID documentation ↗. Note that the same documentation is available from Help in RobotStudio program.
   - Assignment on web: `rs1.pdf` ↗
   - Library file: `UISpenholder.rslib` ↗
2. Drawing using Rudolf
   - Assignment on web: `rs2.pdf` ↗
   - Pack and Go file, `UiS_E458_nov18.rspag` ↗
3. Move pucks using Norbert
   - Assignment on web: `rs3.pdf` ↗
4. Spot weld simulation.
   - Assignment on web: `rs4.pdf` ↗
   - Pack and Go file, `UiS_RS4_16nov2023.rspag` ↗
5. Control robot from Python with input from IDS XS camera
   - Assignment on web: `rs5.pdf` ↗

4 Image Acquisition (IA) part

We hope that there will be one IDS XS camera available for each group during the relevant weeks, 8-10 cameras should be available. But there are limited resources for this part as well. The IDS XP camera on the camera rig is needed for assignment IA4. As for now there are two camera rigs available, they are slightly different from each other but we hope that both will work well. However, the students must accept to do most of the work with camera off line.

The first assignment guide you through how to install `python` (Visual Studio or `Anaconda`) and all the needed packages and check that it is (correctly) installed, see `littPy3x.pdf` ↗. The second assignment guide you through how to install IDS software and check that it is (correctly) installed.

You should know that the Python QT programs `appImageViewer1.py`, `appImageViewer2.py`, `appImageViewer3.py` and `appImageViewer4.py` are made partly as example "solutions" to parts of assignments IA2, IA3 and IA4. Even though these examples are not complete solutions, and not intended to be this, parts of the code in these programs may be useful for the complete solutions. Your solutions may start with one of these programs and then change them to what you want, or just copy and paste parts of them into your own code.

1. IA using a smart phone
   - Assignment on web: `ia1.pdf` ↗
   - Document on how to use Python for this course `littPy3x.pdf` ↗
   - Example files `ELE610py3files.zip` ↗
   - The (official) Python tutorial ↗.
   - The (official) Python reference ↗.
   - The W3 Numpy Tutorial ↗.
   - Numpy for Matlab users ↗
   - OpenCV ↗ tutorials
   - OpenCV-Python ↗ tutorials.
2. IA using IDS μEye XS camera.
   - Assignment on web: `ia2.pdf` ↗
   - Camera web page IDS μEye ↗
   - Camera drivers available from IDS web page IDS download ↗, There are drivers available for Windows, 32 bit and 64 bit, and Linux. The same driver can be used for all IDS cameras. Only registered IDS users can download the drivers, the registration is free, and not too

complicated[1]. The SDK manual is also available from the IDS webpage, and an older version (not much has changed) is available from a link in *canvas* assignment IA2.

- The IDS camera driver are somehow difficult to use and some previous students have made a Python class that wraps around the IDS driver and thus makes it easier to use. This file is: `clsCamera.py` ↗. Feel free to expand and improve this class.
- Some years ago I downloaded some example files from IDS, Python files for Qt4 which I modified to Qt5. Perhaps this link is still valid and points to a similar example: IDS techtip ↗,
  - `pyueye_example_main.py` ↗,
  - `pyueye_example_gui.py` ↗,
  - `pyueye_example_camera.py` ↗ and
  - `pyueye_example_utils.py` ↗.

  For more details see `littPy3x.pdf` ↗ where there should be some information on `pyueye`, the IDS Python interface.

3. IA using IDS μEye XS camera
   - Assignment on web: `ia3.pdf` ↗
   - Basically, this assignment expands and completes IA2.
4. IA using IDS μEye CP camera to capture image of rotation disk. This assignment may need additional time which is available at the end of the course.
   - Assignment on web: `ia4.pdf` ↗
   - Uses IDS μEye CP camera (on the rig).
   - IDS Camera web page replace models all the time, you may look at similar cameras, like IDS UI-3160CP Rev.2.1 ↗ or other camera models.
     The "old" models UiS has may not be available any more. Anyway, this data sheet ↗ should be available as I made a copy of it once.

[1] If you dislike registration, students may copy the driver from another student or from one of the stationary PCs in E462 or E464.