# Aarhus University BSc Course Project Description
# Autonomous Mobile Robots (Spring 2022)
# STADS UVA code 285211U006

## Description

Robotics has been, and will continue to be, a game changer in academia and industry. Several companies invest billions of dollars in robotics, and different authorities in different countries have already started discussions on how they can adapt themselves to the revolutionary research activities in robotics and artificial intelligence. Motivated by the factors above, we have defined a project for you within the areas of **control systems, artificial intelligence and robotic vision!**

In this project, you will design control, navigation and detection algorithms for the aerial robot Drone Racer F250 (see Fig. 1) which will be operated in a Vicon motion capture system in Aarhus University Deep Tech Experimental Hub (Skejby Nordlandsvej 301, 8200 Aarhus).

You will use the skills you have learnt throughout the lectures and the problem solving (PS) sessions. Unless specified by the task, you can use any sensor of your choice. The development of the project is as follows:

- Implementing tasks 1,2 and 3 that will be described below in Robot Operating System (ROS), and testing them in Gazebo simulator. The simulator environments will be uploaded to Brightspace.
- Testing the developed software in real world (in Aarhus University DeepTech Experimental Hub).

The three tasks in this project are as follows:

## 1. Hovering

Your first objective is to make the aerial robot in Fig. 1 take off and hover 1m above the ground. The specifications are as follows:

- The aerial robot has to maintain its position, regardless of external disturbances.
- The drone can have an arbitrary yaw angle, hence there is no yaw angle control. Therefore, you only need to design a proportional-integral-derivative (PID) controller for the drone.
- Different settings for the PID controller will be tested and analyzed to elaborate the effect of $K_P$,

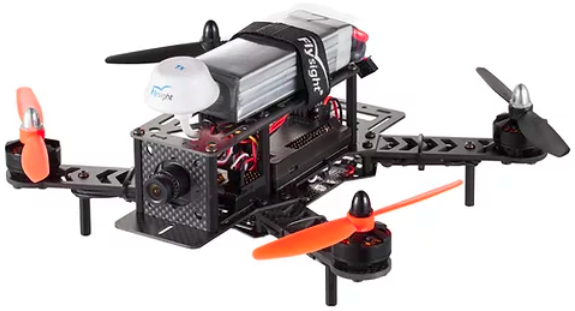

Figure 1. Drone Racer F250.

Figure 2. Deep Tech Experimental Hub

$K_I$ and $K_D$ on the overshoot, rise time, and steady state error of the system.

## 2. Path planning

You will be given the positions of four waypoints and obstacles. The second objective is to follow a prescribed trajectory, while avoiding the obstacles. These are the specifications:

- Waypoint following.
  - You will be given the target waypoints and obstacles locations and boundaries. The trajectory can be defined offline.
  - You will have access to a Vicon system given ground-truth measurements of the real robot's pose.
  - In order to avoid obstacle, the drone needs to maintain yaw angle (i.e. look forward), as it could be dangerous to do otherwise.
- Bonus exercise: online obstacle avoidance. There will be unknown obstacles in the path that the robot has to detect and avoid with an RGB-D camera.

## 3. Object detection

Is this a drone racing or a Mario Kart racing field? Looks like the floor is full of coins! Help your drone on finding them. For that, you will use the drone camera, which is looking to the ground with a $45°$ inclination.

The task specifications are as follows:

- Implement a coin detector.
- There are four different kind of objects now! Implement a network that is able to distinguish each of them. The four different kind of objects are shown in Fig.3

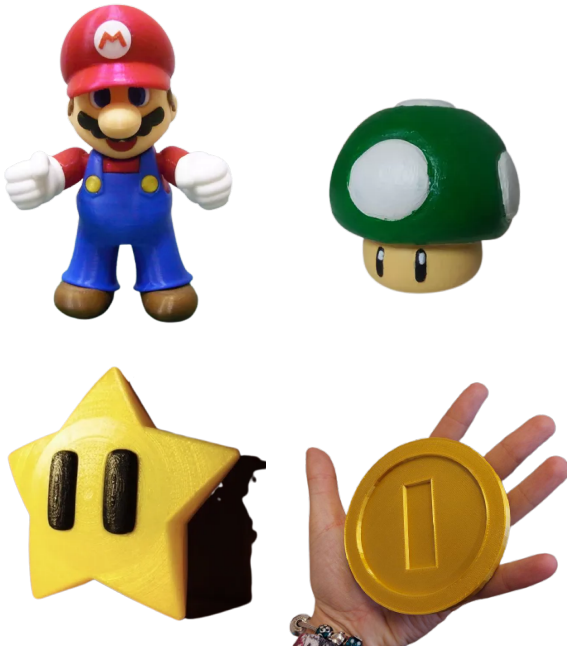

Figure 3. The four different kind of objects for the object detection task. From left to right and from top to bottom: Mario, mushroom, star and coin. The first part of the practice only requires you to detect the coin, the second part requires to classify the four kinds of objects. Note that the colors of the models may differ from the ones displayed here (sometimes Mario enjoys wearing a color different from red).

### Submission rules

There will be teams of three students each. The team members must be registered with the (also in Brightspace). After the completion of the project, each team is expected to deliver the following items:

- A technical report (max 8 pages) with the template in Brightspace and AU digital exam system. The report has to detail the team's methodology, implementation to solve each task, and reports on results and explanations for each of the task.
- The code developed for the project, compressed in a .zip file. Please add instructions regarding how to run your code. (Preferably on Ubuntu 18.04)
- A video file (max 2 minutes and max 50 MB), which should be as descriptive as possible. You may want to include some videos from the real time tests. You can use the GoPro cameras in the lab to record your tests.

Each student in the team is expected to clarify their contributions to the project and should be able to demonstrate the understanding of the method(s) and result(s) presented in the report.

### Deadline

Groups of three students for the project must be formed before February 17th.

The lab will be accessible until 19th May, 2022. We advise you to finish real experiments by that time.

Your technical report, video, and source codes must be uploaded in Brightspace and AU Digital Exam system by 11:59 (before noon time), 9th June, 2022.

### Late submission policy

Late submission is not allowed.