

Aerial Robots

TP0: Introduction to PX4

Linux is our development environment

1. If running Windows, reboot your computer
2. Select Linux in the boot menu that appears

GitHub for source code management

1. Create an account on GitHub: <https://github.com/>
2. Submit your GitHub username here: <https://go.epfl.ch/submit-username>
3. Wait for email confirmation...

Manual

- Contains detailed descriptions for all exercises
- You can find the [manual](#) on Moodle

Forum

- Don't hesitate to create topics and ask questions on the forum!
- The [forum](#) is also on Moodle

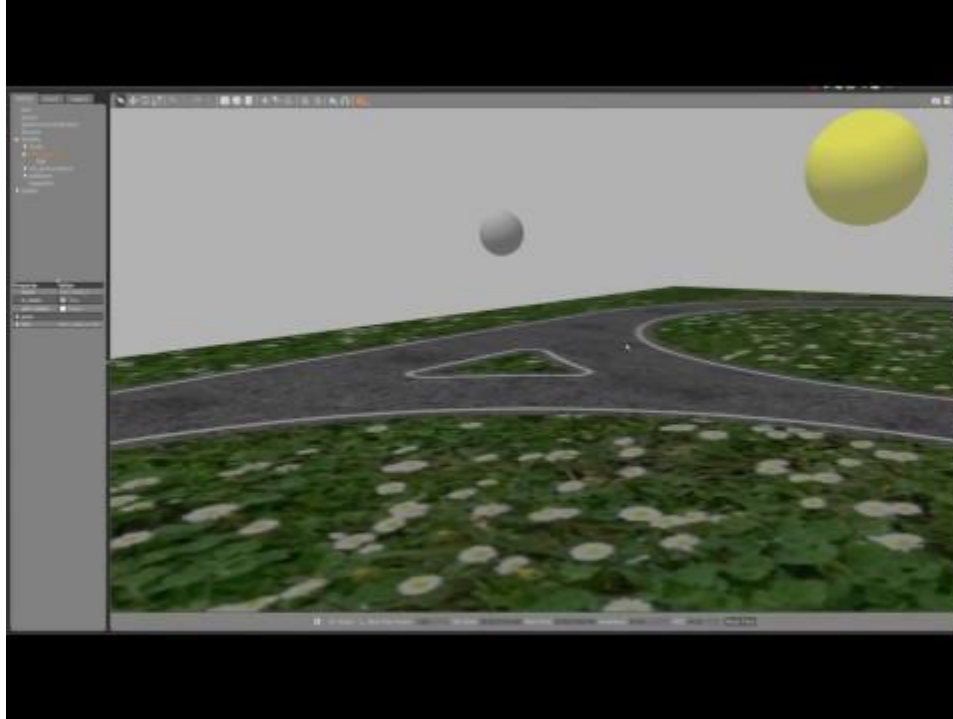
Lab exercises

3 Tasks :

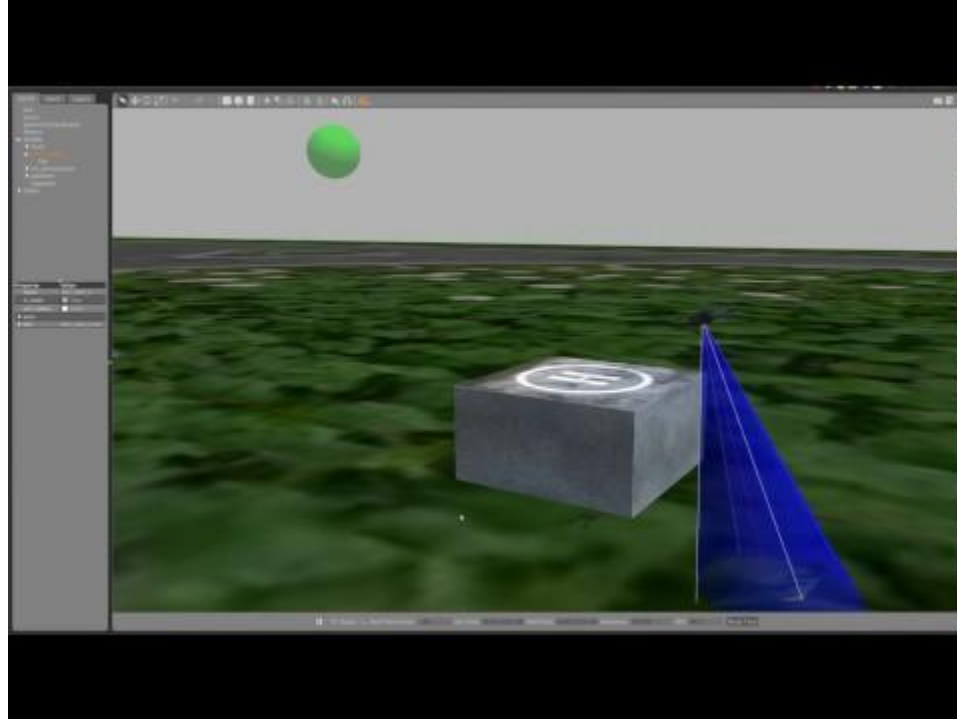
1. **Waypoint navigation** (2h)
2. **Sonar landing** (2h)
3. **Target following** (14h)
 - a. Transformation for camera to local frame (2h)
 - b. Tracking of moving target using a Kalman Filter (6h)
 - c. Landing on the moving vehicle based on position estimation (6h)
- **Final integration** (4h)

Inspired by the [MBZIRC 2017 challenge](#)

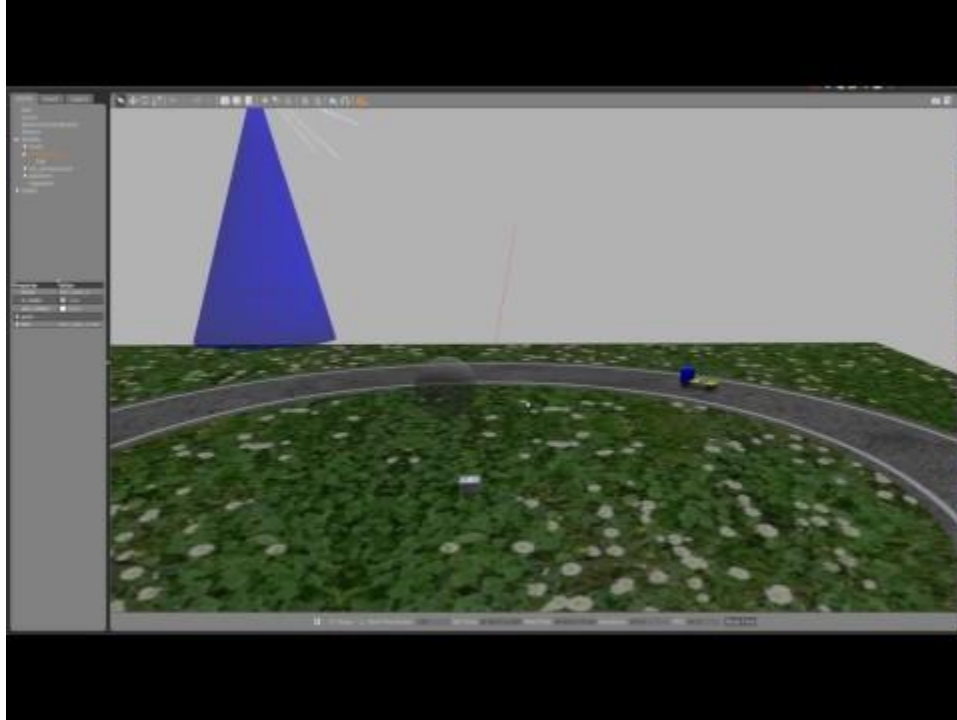
Task 1: Waypoint navigation



Task 2: Sonar landing

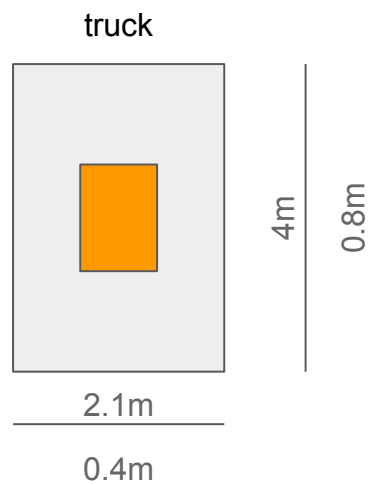
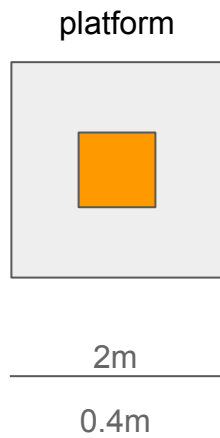
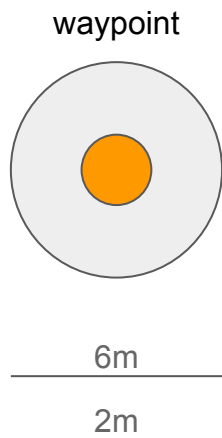


Task 3: Target following



Evaluation metrics

1. 300pts: Time to complete all tasks (every task has a timeout = 10 min)
 - This is competitive among students: best timing will get 300pts, timeout 0pts
2. 300pts: Accuracy for each task
 - Full score: 100 pts, linearly decreases to 0 with distance
3. The best performance out of 5 attempts will be considered



Full score area

Clone your private repository

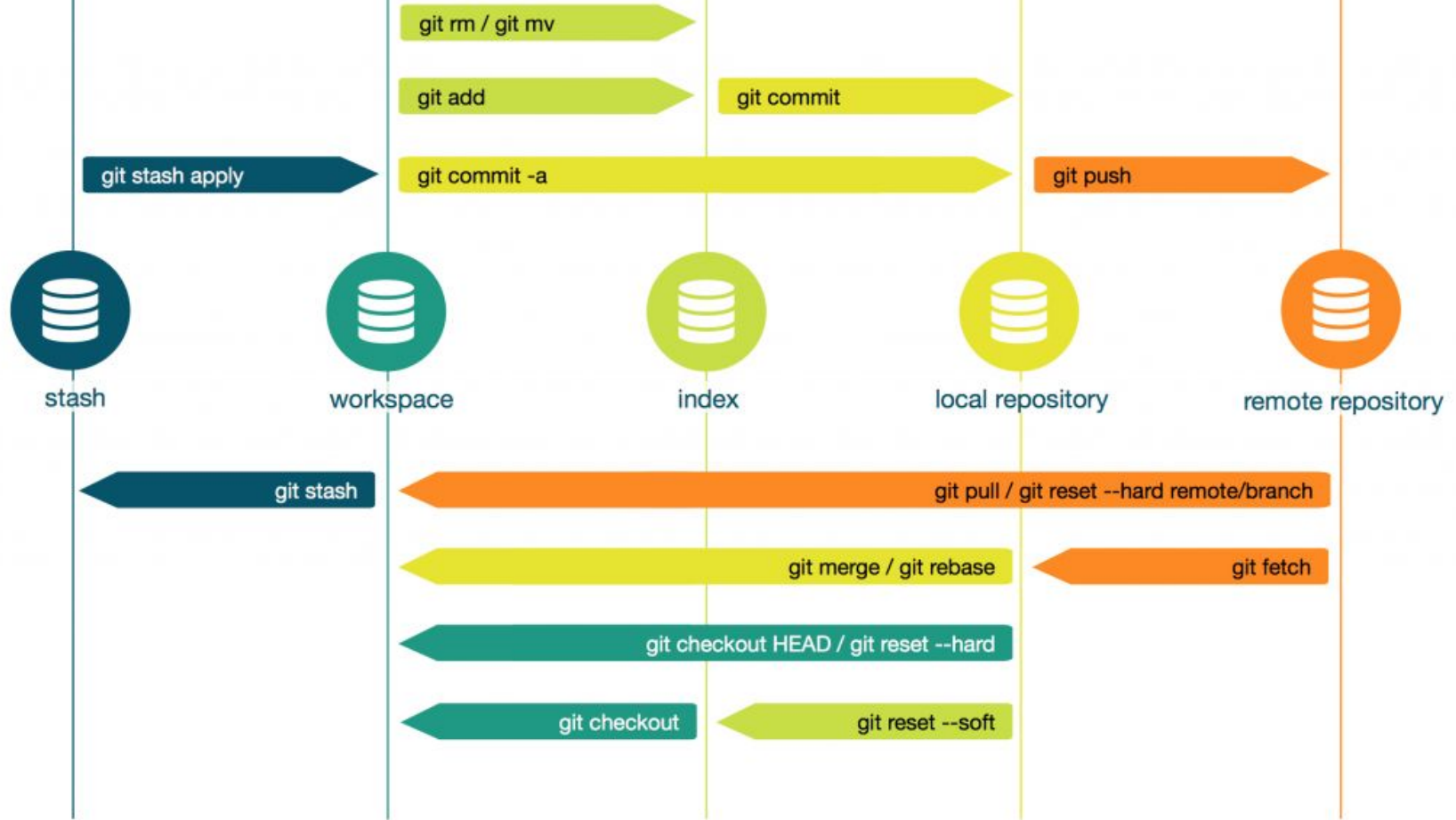
```
git clone
```

```
https://github.com/dronecourse-epfl/dronecourse-student-SCIPER
```

Where SCIPER is your 6 digit sciper number as in 123456

git data transport commands

patrickzahnd.ch



Hello Sky

- Your first application running along the PX4 autopilot
- Follow the tutorial here: <https://go.epfl.ch/hello-sky>
- **Important:** we are not using default PX4 but our own version
 - To build: `make dronecourse_hellosky`

Save your work to your remote repository

1. `git add <file name(s)>`

(ex. `git add src/examples/px4_simple_app/px4_simple_app.c`)

- This will add the file to your next commit

2. `git commit -m "<commit message>"`

(ex. `git commit -m 'Implementation of HelloSky tutorial'`)

- This will save your changes locally.

3. `git push`

- This will publish your changes to the online repository. Now you can sync from a different machine and access older versions online.

Note: you can repeat 1. and 2. several times and push multiple commits

Important: do the following after every lab session

1. Push all of your local changes to the remote repository
 - a. `git add --all && git commit -m "commit message" && git push`
2. Remove your repository from the computer
 - a. `cd .. && rm -rf dronecourse-student-<your sciper>`
3. Log out of browser sessions (Github, Webmail, Facebook, Twitter, etc.)
4. Reboot the computer

Useful git commands

- **clone** - copy the content of a remote repository to your local folder
- **pull** - **fetch** and **merge** changes from a remote repository to your local folder
- **add** - add a certain file to your list of files to stage
- **commit** - commit your changes to the local repository
- **push** - publish your committed changes from your local to the online repository
- **status** - get a list of files that have been changed since your last commit
- **diff** - list the differences between the old and new version of a modified file