

Assignment 2

Using the Kalibr package

From: <https://github.com/ethz-asl/kalibr/wiki/installation>

Install the docker following the instruction in the above page. It is recommended to use an AMD64 architecture and a Linux OS.

Read the documentation of Kalibr (<https://github.com/ethz-asl/kalibr/wiki>) and watch the tutorials.

To run kalibr apply the following commands:

```
>cd kalibr
>mkdir data
>FOLDER=$(pwd)/data
>xhost +local:root
```

Download the bag files from the course webpage to ~/kalibr/data/

```
>docker run -it -e "DISPLAY" -e "QT_X11_NO_MITSHM=1" \ -v "/tmp/.X11-
unix:/tmp/.X11-unix:rw" \ -v "$FOLDER:/data" kalibr
```

```
>source devel/setup.bash
>roslaunch kalibr kalibr_calibrate_cameras --bag /data/GX010164.bag --
target /data/aprilgrid.yaml --models pinhole-radtan --topics
/gopro/image_raw/compressed
```

After finishing, a report is produced.

Performing calibration online

Read through the https://wiki.ros.org/camera_calibration page, and follow the tutorial for a monocular camera. Print a calibration target, attach it to a flat surface, and use "--size" and "--square" appropriate for your calibration target.

Run the docker for kalibr again with an extra flag:

```
>docker run -it --device=/dev/video0 -e "DISPLAY" -e
"QT_X11_NO_MITSHM=1" -v "/tmp/.X11-unix:/tmp/.X11-unix:rw" -v
"$FOLDER:/data" kalibr
```

Please note: the “--device=/dev/video0” gives access to your camera.

```
>source devel/setup.bash
>roscore&
>sudo apt install ros-noetic-usb-cam
>roslaunch usb_cam usb_cam_node &
```

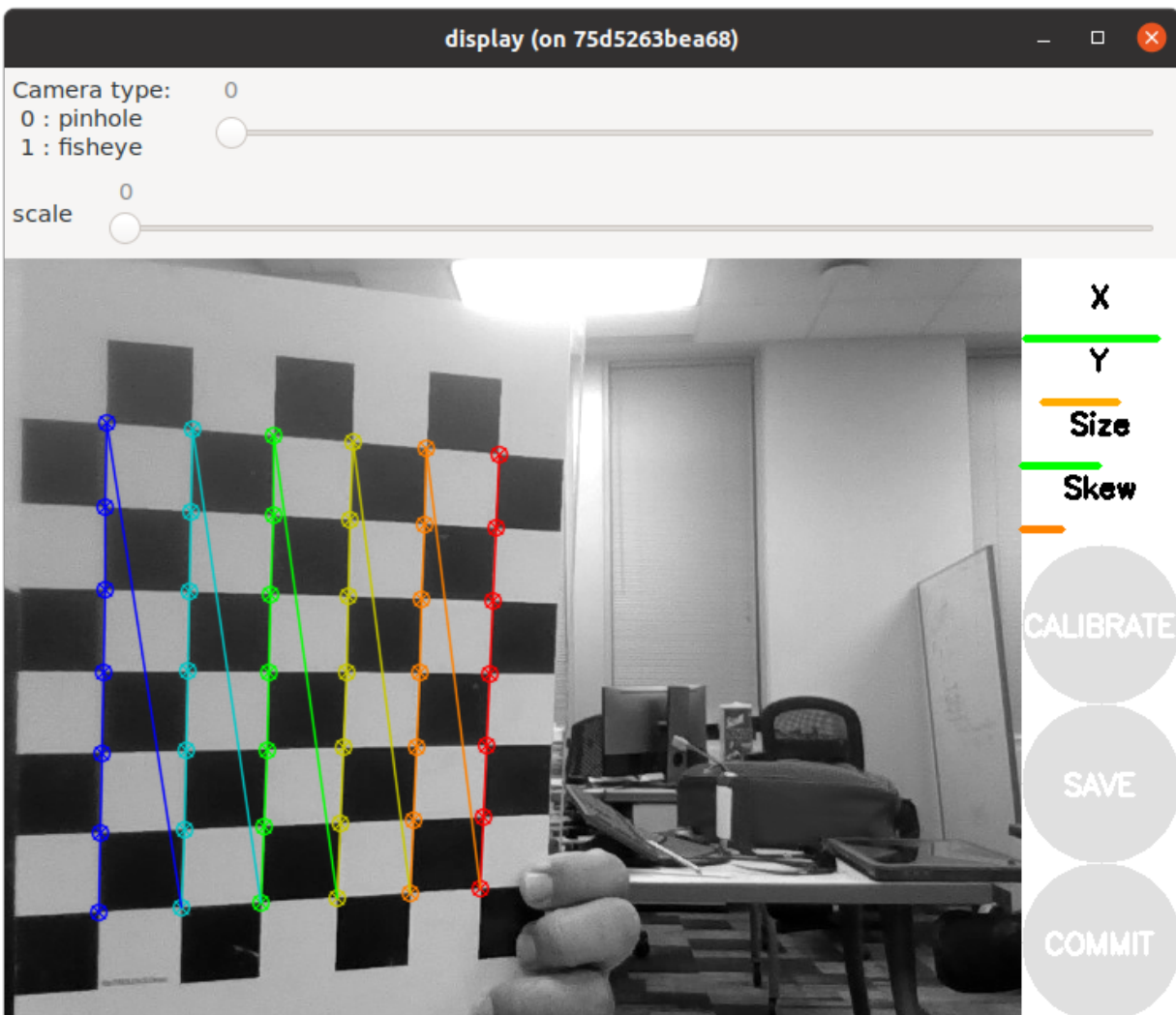
You can see the image using:

```
>roslaunch image_view image_view image:=/usb_cam/image_raw
```

Run the calibration:

```
>roslaunch camera_calibration cameracalibrator.py --size 6x7 --square
0.03 image:=/usb_cam/image_raw camera:=/usb_cam
```

Your results should look like:



Produce a report with the results from the two calibrations. Include coverage screen shots from kalibr report. Try different camera models and report on the results.