

Lab 2 and 3 Notes

EE 346

Autonomous Mobile Robot Navigation and Control

Winter 2022

Due dates

- Lab 1 and 2 are due today (March 23)
- Lab 3 is due two weeks from today, April 6, at the end of the class (4:00 PM)
- The lab hours are not sufficient for you to complete the lab. There is no assignments in this course, and the idea is for you to spend lots of extra hours outside the lab sessions on your lab assignments.
- Failure to complete on time will see a mark reduction of 20% of each day late. E.g., if you have completed Part I (30%) and Part II (30%) but one day late on Part III, you will receive $(30+30+32)\% = 92\%$; if you are two days late (checked on Friday April 8), you will receive 84%.

Additional Information About Lab 3

- You can get a web cam easily if your computer does not have one such as the one on the right.
- Make sure that its resolution is “480P” (640x480 roughly).
- Use `%ls -l /dev/video*` to determine the port number.



Additional Information About Lab 3 (Part I)

- Check if images are being published properly by
`% rosrun image_view image_view image:=/usb_cam/image_raw`
- Replace `/camera/image_raw` with `/mycam/image_raw` where you can find “mycam” by looking at where your camera is publishing images (`% rostopic list`), e.g., example, `/usb_cam/image_raw`
- Replace “-- square 0.107” by the actual square size, e.g., “-- square 0.025”
- `cameracalibrator.py` needs 40+ high-quality images (good lighting and high resolution) before running the calibration algorithm, so be patient.
- The default file name for the calibration result is
`~/.ros/camera_info/head_camera.yaml`
- Part I must be done first before you can attempt Part II.

Additional Information About Lab 3 (Part II)

- Doing Part II without calibration, you will see the error:
“Camera calibration file ~/.ros/camera_info/head_camera.yaml not found”
- When printing an AR-tag, there must be a white border for the tag to be recognizable.
- To see the AR tag that has been recognized, you can “echo” the messages published on the topics /visualization_marker and /ar_pose_marker by
% rostopic echo /visualization_marker
- To terminate roscore and all nodes so as to start a clean run, do
% rosnode kill -a; killall -9 rosmaster; killall -9 roscore

ArUco Marker Detection (<https://github.com/behnamasadi/aruco-ros>)

- Generate a few aruco markers with the process and the Python script described at <https://automaticaddison.com/how-to-create-an-aruco-marker-using-opencv-python/>
- Install the melodic version of ROS package aruco-ros
% sudo apt-get install ros-melodic-aruco-ros
- Change parameter “pixel_format” from “mjpeg” to “yuyv”, in the launch file, “usb_cam_stream_publisher.launch”
- Reproduce the result shown in the video at the end of the GitHub page of “aruco-ros”:
<https://github.com/behnamasadi/aruco-ros>
- DICT_ARUCO_ORIGINAL: 1024 markers, 5x5 bits, 0 minimum distance
- OpenCV page about Aruco markers:
https://docs.opencv.org/3.1.0/d5/dae/tutorial_aruco_detection.html
- Online Aruco marker generator is available at: <https://chev.me/arucogen/>