CMPUT 412: Experimental Mobile Robotics
Winter 2018
Demo #5
Due Date: Thursday, March 8, 2018
Camera Calibration and Pose Estimation

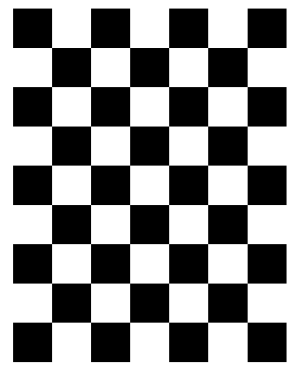
Objectives

- Learn how to calibrate a camera
- Control a robot to dock at a charging station through pose estimation

Procedure

Part I:

Camera calibration refers to the process of determining the intrinsic parameters of a camera, which include its focal lengths, image center, and radial distortion coefficients. Camera calibration is essential in order to use a camera to determine the 3D structure of a scene. The process of camera calibration consists of (a) capturing images of a registration pattern or rig (shown on the right), and (b) passing the images to a calibration software that estimates the camera intrinsics. Fortunately, a camera calibration node is readily available in ROS for you to perform this.



Follow the steps described the MonocularCalibration node to calibrate your Xtion Pro camera. Note that you will need to print the registration pattern first, and that the larger this pattern is (in terms of the width of each square at 108mm by default), the more accurately the camera can be calibrated. Once the camera is calibrated, evaluate the result on radial distortion correction by using the image_proc node in ROS. Test the linear part of the intrinsics (focal lengths and image center) by moving one registration rig with respect to another for a known distance or rotation, project the predicted x- and y-axis of the rig to both images. Compare the predicted and real axes (see the exam on this page used by Part II for how this projection of an axis to an image can be done).

Part II:

You will control your robot to a docking station marked by the registration rig – which you used in Part I – affixed to a wall at a height easily visible from Xtion Pro. This is done continuously computing the relative pose of the rig and the robot and moving the robot towards the goal position until it is reached. Your robot must come to the inside of a 50 cm by 50 cm square in front of the registration rig for the task to be considered completed. OpenCV provides a function for the pose estimation problem called `cv2.solvePnPRansac()` and a tutorial on how to use it can be found on this page. To avoid the problem of finding the registration pattern in the scene, you can assume that the rig is visible initially, at one of the three positions: rough 3m away from the wall and to the left (by 2m), center and right (by 2m) of the registration rig. The same docking solution must work for all three positions.

Marking

If you are able to complete the demos before the end of the lecture, you will receive: 30% for Part I (independent demonstration) and 70% for Part II (group demonstration).

If a student is not able to complete any parts of the demo within the lecture session, you will get a 20% penalty of that marking component, and an additional 20% for each day of delayed demo.