Homework 3: Vision Intro

Implement this homework by modifying *detect.py* file provided to you. Hand in this homework by uploading the .zip archive via the ELMS website.

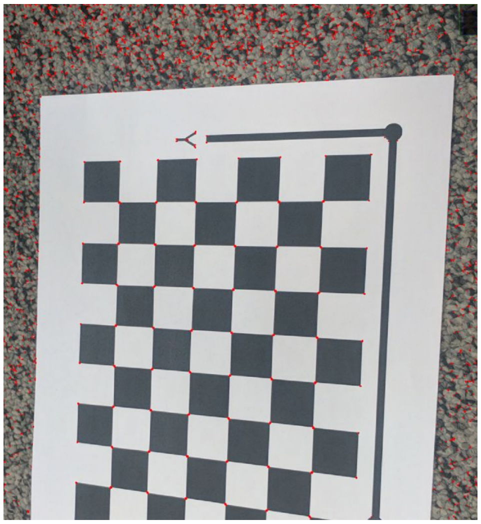
Most of the guidelines (as well as starter code) are designed for Python. C++ developers will get some extra credit (+20%) for their implementations. If you have ROS installed on your laptop, OpenCV should be already there. If not, you will need to install it. Try to use OpenCV 2.4!

DELIVERABLES:
- ***.zip*** archive with your code and other files
- **.csv** file with detected corners for part 1
- Camera calibration file for part 2

1. Chessboard (15 pts)

For this part you need to detect the chessboard corners. The starter code is given to you in *detect.py* - it will open an image and show it on the screen, but you will need to do all the processing. Modify the *process* method of the *CbDetect* class. Surely, there is a ready-made function in opencv (*findChessboardCorners*) which can detect chessboard corners for you, but you should write your own one! You will need to use pyopencv and numpy (opencv and eigen for C++) and you will need to detect chessboard corners but remove all false positives (many of them are on the carpet). You will have some sample chessboard pictures in the *Imgs* folder. To run the starter script, run:

./detect.py -n Imgs/IMG_20170209_042634.jpg



Your code (given an input image) should show it with corners detected. Also, it should generate a *.csv* file with each detected corner on a new line.

The (probably) easiest way to go will be to use the Harris corner detector and then add some constraints - you know how many corners you need to detect (the chessboard is fully visible all the time), and you know that points tend to be located on one line. You can also detect the white background of the chessboard to remove the carpet!

Some C++ OpenCV tutorials: link, link

Python tutorials: link

2. Calibration (5 pts)

Try a real camera calibration in ROS! Follow this tutorial and get a camera calibration file for you own web camera! Do not forget to put this file in your .zip archive.