ME-GY 7863 C: Special Topics in Robot Perception Assignment 1

Instructor: Dr. Chen Feng

Task 1: Sherlock’s Message (10 points)

It is October 24th, 2048. The famed detective, Sherlock Holmes, has been investigating a series of recent autonomous driving incidences related to abnormal malfunctioning of their perception systems. Sherlock traced clues for several months which inevitably lead him to his archenemy, Professor Moriarty, a notorious cracker, and his acolyte, Colonel Moran, a vicious roboticist wanted by Scotland Yard for a long time. Sherlock believes Moriarty and Moran must have been trying to use an old technique named adversarial attack to tamper with YouDrive, the driverless car system used worldwide. YouDrive is so complicated that almost no one can understand it both holistically and atomistically. To prevent Moriarty and Moran to make more harms and damages, Sherlock decided to study robot perception, the engineering field that enabled YouDrive, to better fight again the sinister gang.

This evening at around 6 pm, Dr. John Watson, Sherlock’s roommate and a doctor who is also well-known for pioneering the use of artificial intelligence in his clinical practices, came back their home at 221B Baker Street, London, and found Sherlock missing, who originally initiated the dinner tonight with John and Ms. Irene Adler, the CEO of YouDrive and the frenemy of Sherlock, to discuss about those incidents. However, he seems to have left a secret message for them on the laptop left open on purposely, as shown in Figure 1.

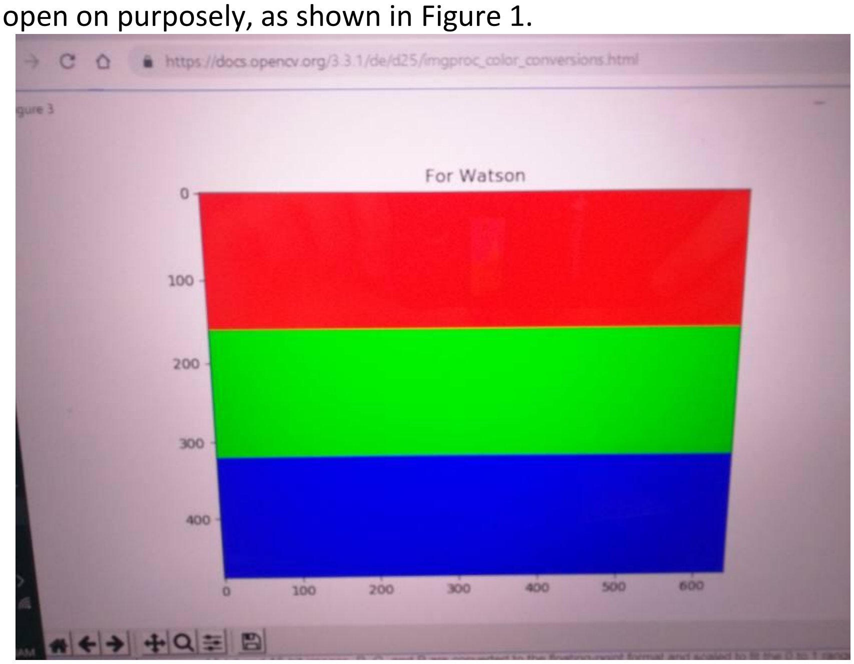

Figure 1 Dr. Watson's photo of Sherlock's laptop screen.

Since neither John nor Irene specializes in robot perception, they came to you, the CTO&CSO of YouDrive. The image has been saved as the **for_watson.png** file. Can you please write some code and decode the message from Sherlock?

Following Sherlock's message, help him to finish the following tasks:

Task 2: Vanishing Points (25 points)

Take an image with your camera so that in the image you can find 3 orthogonal directions, i.e., 3 vanishing points. Calculate, by hand or by code, those three vanishing points and use them to calibrate your camera (focal length f, and principal point position, x_0 and y_0). Document and report the process. Visualization of the process and the result is recommended.

Task 3: Camera Calibration (20 points)

Use the pyAprilTag package provided in the class, or other free packages that you may be aware of, to calibrate your camera (full K matrix, with the top two distortion parameters k_1 and k_2). Compare this calibration result with the one you obtain above and discuss the pros and cons of the two methods.

Task 4: Tag-based AR (25 points)

Use the pyAprilTag package to detect an AprilTag in an image, for which you should take a photo of a tag. Use the K matrix you obtained above, to draw a 3D cube of the same size of the tag on the image, as if this virtual cube really is on top of the tag.

Figure 2 An example result of Tag-based AR

Task 5: RANSAC Plane Fitting (20 points)

Write codes to fit at least one 3D plane using RANSAC in the point cloud stored in **record_00348.pcd**. To load the data, you may find these pages useful:
http://www.open3d.org/docs/tutorial/Basic/python_interface.html#install-open3d-from-conda
http://www.open3d.org/docs/tutorial/Basic/file_io.html#point-cloud

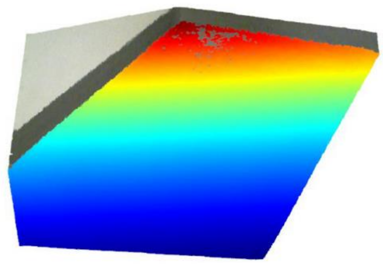

Figure 3 An example plane detection result of the record_00348.pcd