Assignment 5 - 3D Geometry

IMAGES. All images may be found in the DATASETS folder `https://drive.google.com/drive/folders/1fUo6TQmvkCPWg9MVaEQTyx3tOZOyTEkG?usp=share_link`.

1. **Independent Reading.** Read Chapters 11 and 12 from Szeliski's book.
2. **Independent Reading.** NeRF: https://arxiv.org/pdf/2003.08934
3. **(4 pts) Homography Estimation.** You will be using the Checkerboard Images for this problem. Take a pair of images and estimate the Homography matrix H between them. You can use the `findChessboardCorners` function in OpenCV to extract the corner points. Make sure there is an one-to-one correspondence between the corner points in the two images. Report the homography matrix for image pair 1 and 9. Specifically, you need to estimate H, where $P_9 = HP_1$, where P_1 and P_9 are the extracted points in image 1 and 9, respectively.
4. **(4 pts) Fundamental Matrix Estimation.** In this problem, you will be using the stereo image pairs `viprectification_deskLeft.png` and `viprectification_deskRight.png`, provided in the DATASETS folder. Load the images and detect the key-points i.e. corner points using any predefined corner detector in Python (refer to Assignment 1). Next you will obtain the feature descriptors of the corners using the ResNet50 model in Assignment 3. Use 8×8 patches around each of the corner points to extract the features. If required, make sure to pad or resize the patches before inputting them to the model. Using the matched points, employ the 8-point algorithm to obtain the fundamental matrix. Use random subsets of 8 points to estimate a set of fundamental matrices. Then, for each fundamental matrix, compute the mean error on the rest of matched points using the Sampson distance. Report the fundamental matrix, which produced the least error and also report the mean error. **You should not use built-in functions to estimate the fundamental matrix**. Sampson distance is defined as: $\frac{(x^T F x')^2}{||F^T x||_2^2 + ||Fx'||_2^2}$, where x and x' are the 2D points on the image plane in the image pair.
5. **(6 pts) NeRF.** For this problem, you will infer a pretrained NeRF model to render novel views of a given object. A starter code along with pretrained weights and test data have been provided. You'll need to fill into the blanks as asked in the starter code. Upon completion, the code will save novel views of a given object. Display only a couple of the rendered images in your report.

Submission Protocol. For all problems, you must submit a Jupyter notebook i.e., `.ipynb` file. You are recommended to use Colab. You should add comments to your codes to make them reader-friendly. All codes must be uploaded in separate folders named after the problem number. Keep all the images necessary to run a code in the same folder as the code while you are submitting. The zip file to be uploaded must have your name.