Assignment 1: Camera Calibration

Subrat Kumar Swain(2021QIZ8247)
Chandan(2021EEZ8527)

February 6, 2022

We've done the camera calibration using two methodologies. The first one is by creating a 3d space of checkerboard as specified in the assignment. In the second part, we took a sample of 25 images and calibrated the camera using them. We briefly described both the methods below.

3D Method

At first, we took the printout of the checkerboard of a square size of 2.5cm. We placed the checkerboard at the corner of the house and took the photo as shown in fig [3]. We tried to make the structure as orthogonal as possible. Then we picked some points on the checkerboard plane and note down the coordinates on the image plane using Microsoft Paint, fig [4], *e.g.* the brown point on the XZ plane has the world coordinate (5, 0, 2.5) and the corresponding image coordinate is (447, 1033).

After we got a set of point pairs(real-world coordinates and image coordinates), we normalize the points to zero mean and unit variance across each axis which resulted in $\sqrt{2}$ and $\sqrt{3}$ average distance from mean in the image and real-world coordinate respectively. Then we calculated the projection vector. The projection vector is the smallest Eigenvalue of the A matrix. Then we re-shaped it to find the projection matrix and later applied QR factorization to extract the extrinsic and the intrinsic matrices. The error after re-projection is found to be 1.5.

2D Method

At first, we took the printout of the checkerboard (8*8) of a square size of 2.5cm. We placed the checkerboard at the wall of the house and took 47 photos as similar to shown in fig [add im] with some rotation of camera and inclinations. Then we imported these images and used the findchessCorner function of the OpenCV module and the find out the inner corner points of chessboard squares and then used CornerSubPix, drawchessboardCorner function to draw the marked points in the inner corner points of the chessboard images. Then, used calibrateCamera function of OpenCV to collect camera Matrix (Intrinsic Matrix), Distortion Coefficients, Extrinsic Matrix (Rotational and Translational Matrix). and in the end, calculated reprojection error by taking into account any two images from 47 images and it comes out to be around 0.0618

Run Instructions

To run the 3D analysis: *python final.py*
To run the 2D analysis: *python calibration$_2$d.py*

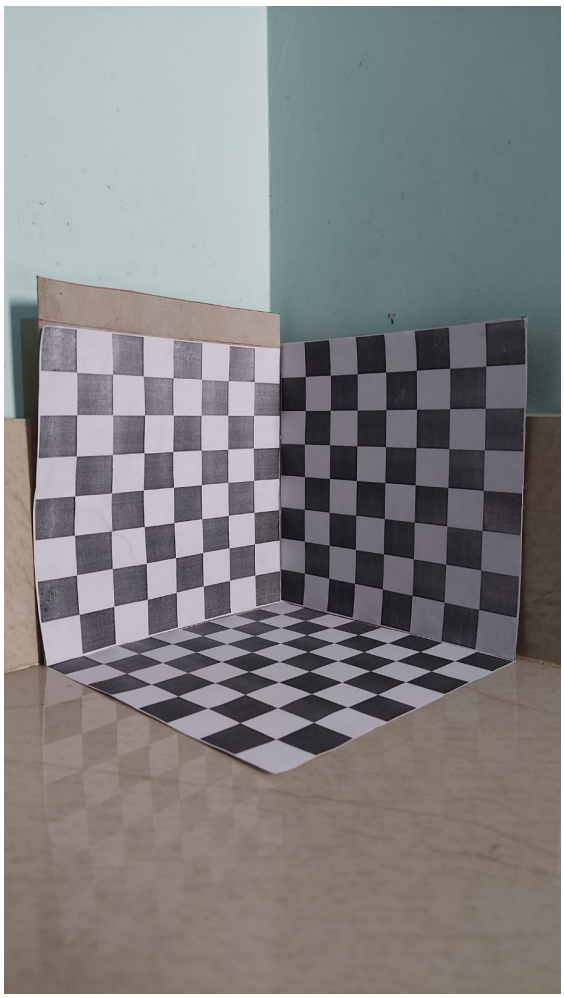

Figure 1: 3d plane of checkerboard

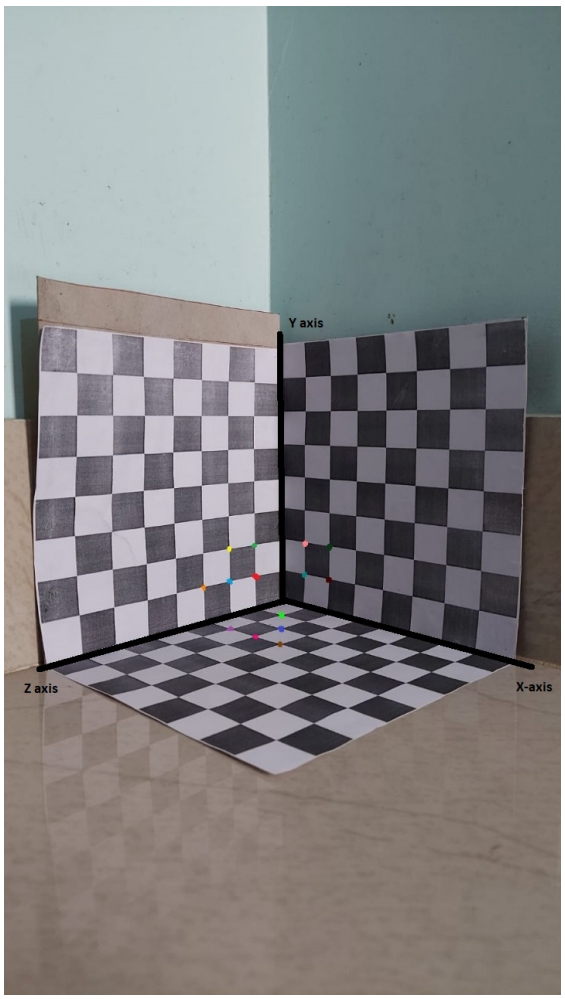

Figure 2: Marked points on the checkerboard

Figure 3: 2d plane of checkerboard

Figure 4: Marked points on the checkerboard