Homework 1: AutoCalib

[Using 1 late day]
Miheer Diwan

MS Robotics Engineering
Worcester Polytechnic Institute
msdiwan@wpi.edu

***Abstract*—This report describes the procedure for the automatic calibration of cameras based on Zhengyou Zhang's work [1].**

I. Introduction

For performing camera calibration, we start by estimating the intrinsic and extrinsic parameters of the camera. The calibration was performed using a checkerboard pattern of size 10×7 with a square size of 21.5 mm. 13 images of the checkerboard pattern taken from different angles were used.

II. Computing the Camera Calibration Matrix

The camera calibration matrix is given by,

$$K = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

The estimated camera calibration matrix is,

$$K = \begin{bmatrix} 2052.78 & -0.369 & 763.06 \\ 0.0 & 2036.63 & 1352.61 \\ 0.0 & 0.0 & 1.00 \end{bmatrix}$$

The camera is modeled using the equation,

$$m = K \begin{bmatrix} r_1 & r_2 & t \end{bmatrix} M$$

Here, M = [X Y Z] is the 3D object point and m = [x y] is the 2D image point. We assume the Z coordinate to be zero as the points lie on the same plane in the image. We first detect the corners in the images using the *cv2.findChessboardCorners* function.We then calculate the homography (H) between the detected points and the object points. The homography matrix can be computed as shown in [2][3]. However, I used the *cv2.findHomography* function from OpenCV. Although, this adds noise to the computed homography matrix. The next step is to compute the V matrix with the help of h_1,h_2, and h_3. Where h_i are the column vectors of H. We then define the equation $V \times b = 0$ and solve for b using DLT/SVD. With the help of b, we can now determine the parameters of the camera calibration matrix. Using K we estimate the R and t for each image and also find the projection error using non-linear optimization.

The projection error (e) = 20.413

III. Estimating Radial Distortion

The radial distortion was calculated by alternation as mentioned in [1].

$$\begin{bmatrix} k1 \\ k2 \end{bmatrix} = \begin{bmatrix} 0.020 \\ -0.27 \end{bmatrix}$$

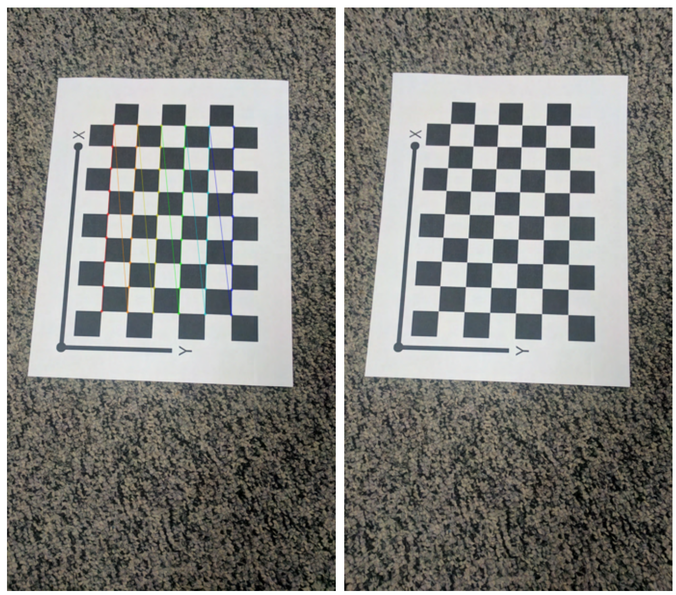

Fig. 1: Image 1 and Distortion Corrected Image 1

References

[1] Z. Zhang, "A flexible new technique for camera calibration," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 11, pp. 1330-1334, Nov. 2000, doi: 10.1109/34.888718.
[2] C. Stachniss, "Camera Calibration using Zhang's Method," on YouTube, April 2020, url: https://youtu.be/-9He7Nu3u8s.
[3] C. Stachniss, "Direct Linear Transform for Camera Calibration and Localization," on YouTube, April 2020, url: https://youtu.be/3NcQbZu6xt8.

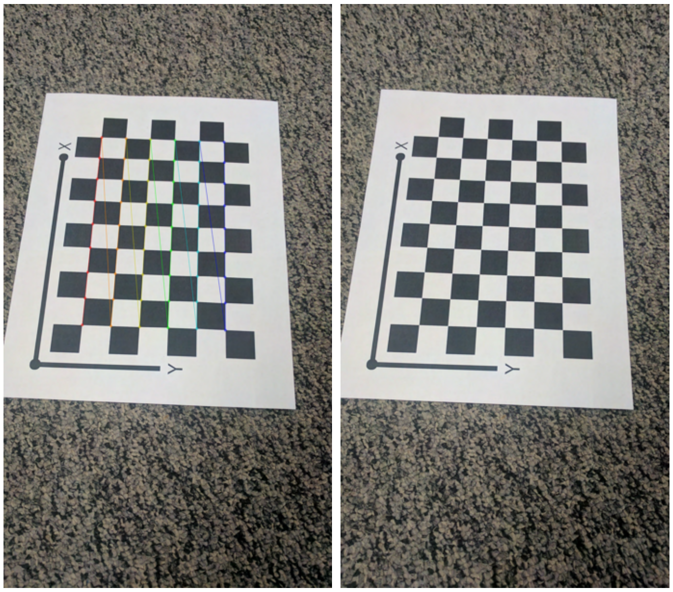

Fig. 2: Image 2 and Distortion Corrected Image 2

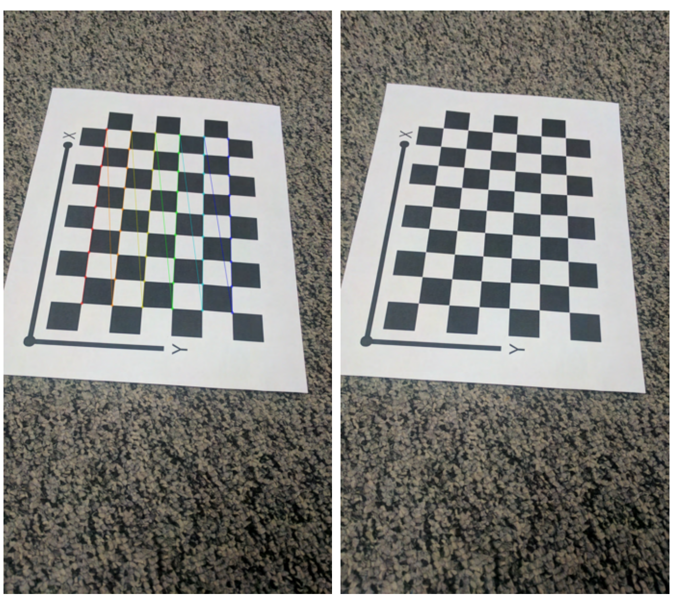

Fig. 3: Image 3 and Distortion Corrected Image 3

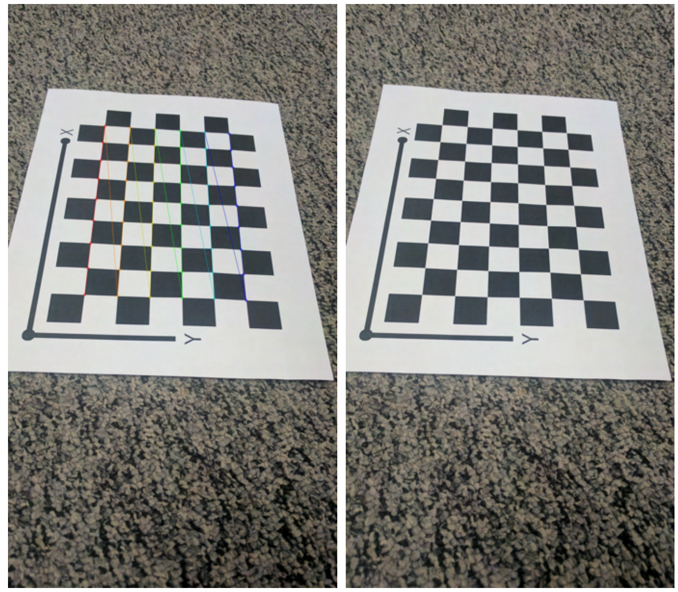

Fig. 4: Image 4 and Distortion Corrected Image 4

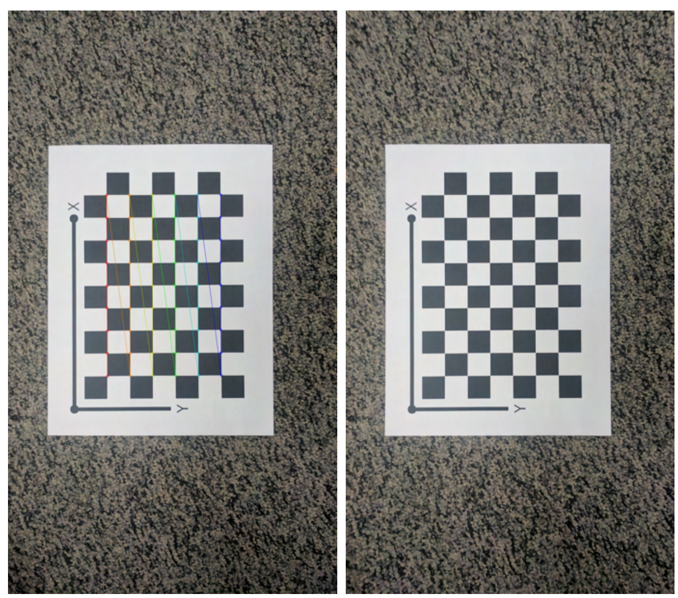

Fig. 5: Image 5 and Distortion Corrected Image 5

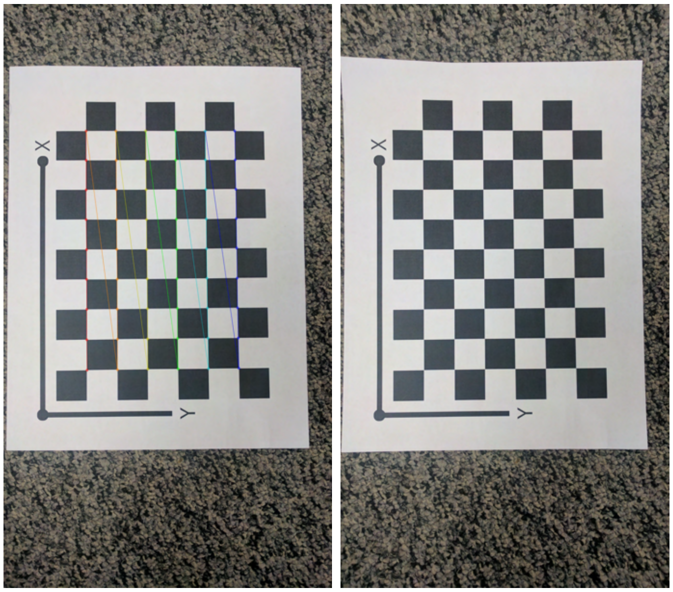

Fig. 6: Image 6 and Distortion Corrected Image 6

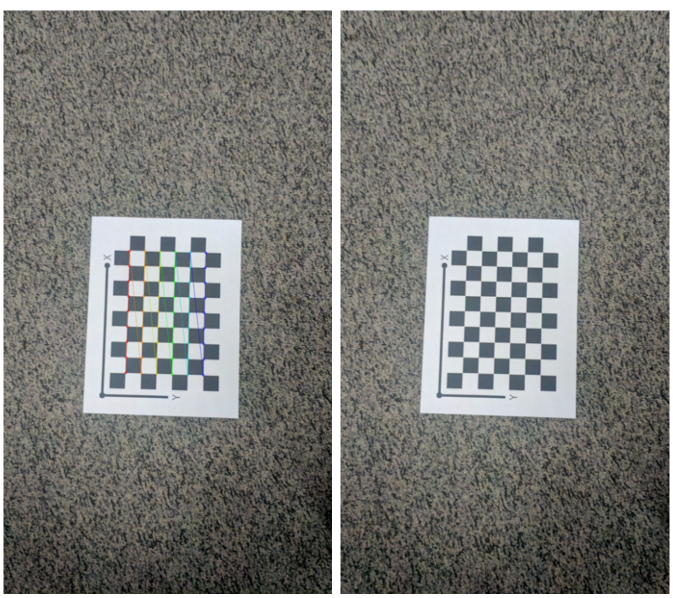

Fig. 7: Image 7 and Distortion Corrected Image 7

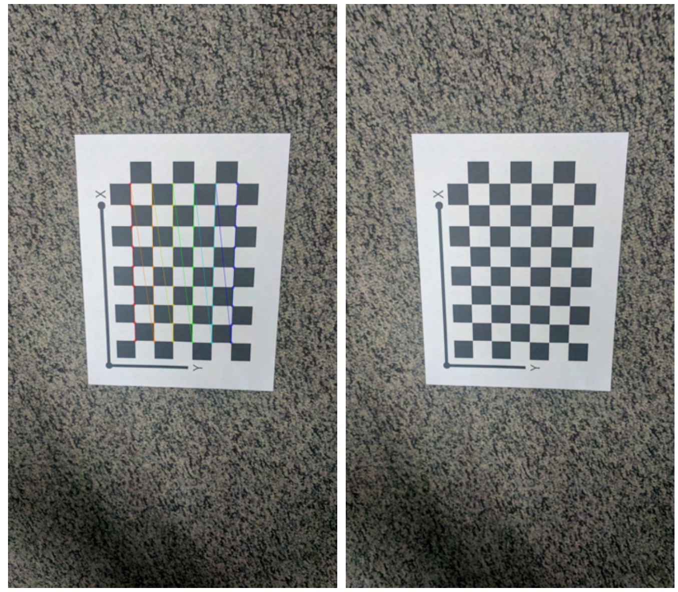

Fig. 8: Image 8 and Distortion Corrected Image 8

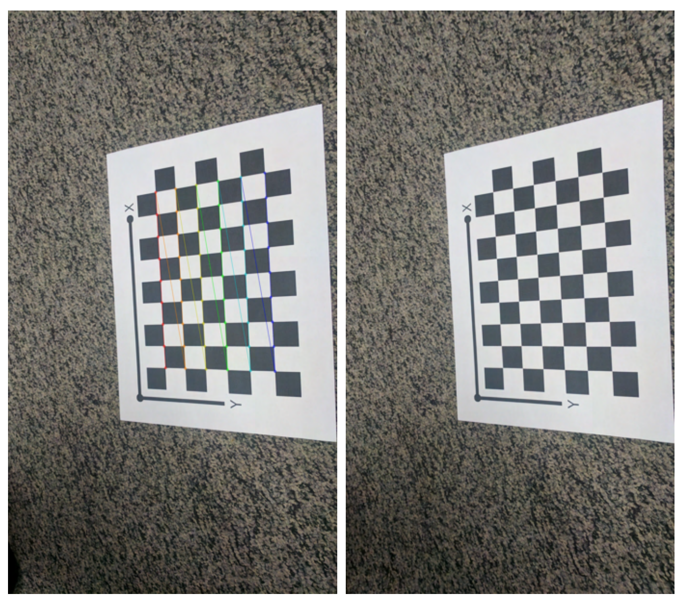

Fig. 9: Image 9 and Distortion Corrected Image 9

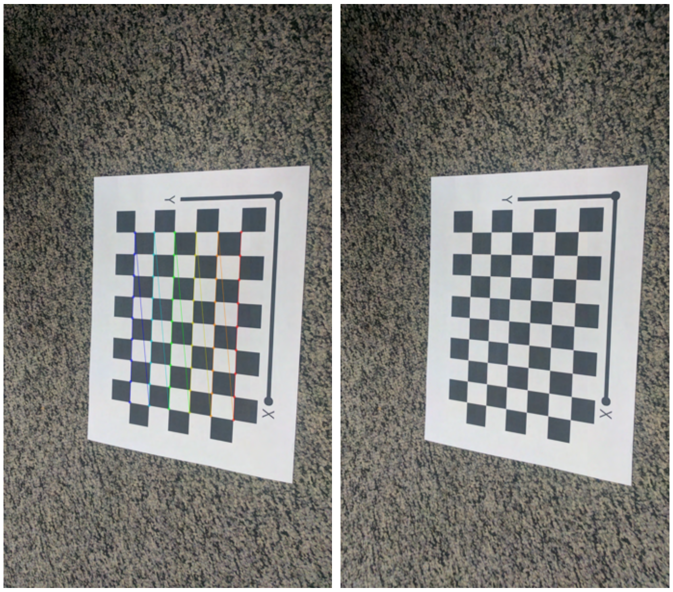

Fig. 10: Image 10 and Distortion Corrected Image 10

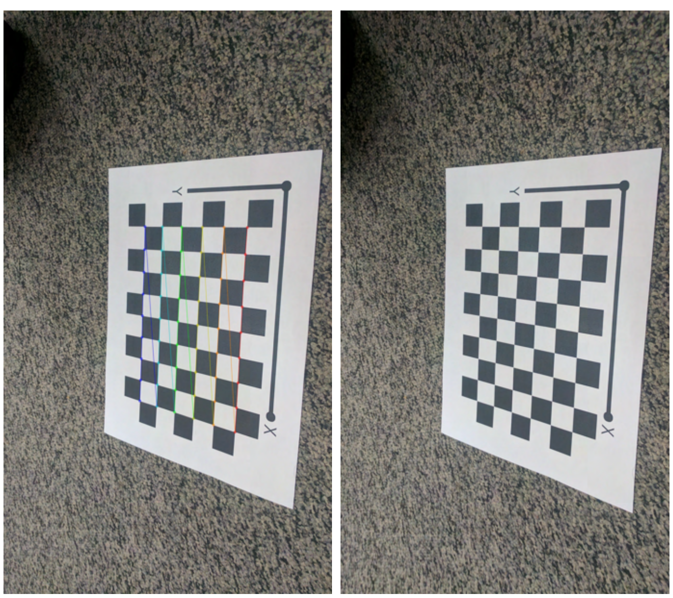

Fig. 11: Image 11 and Distortion Corrected Image 11

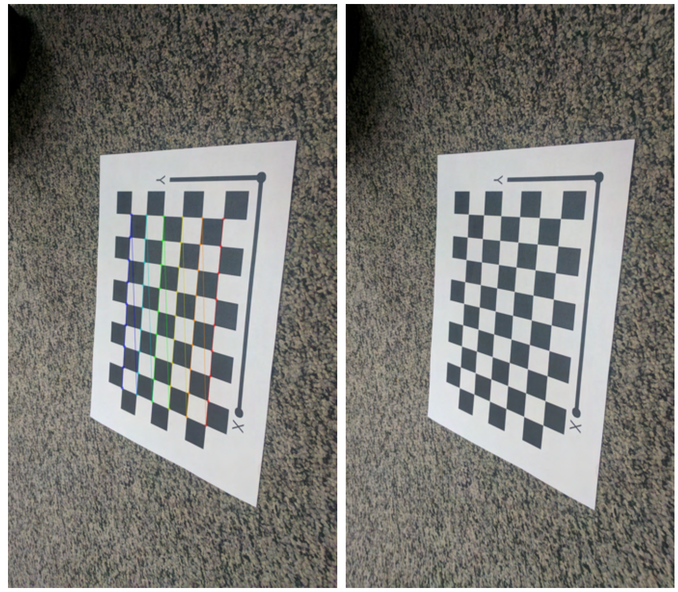

Fig. 12: Image 12 and Distortion Corrected Image 12

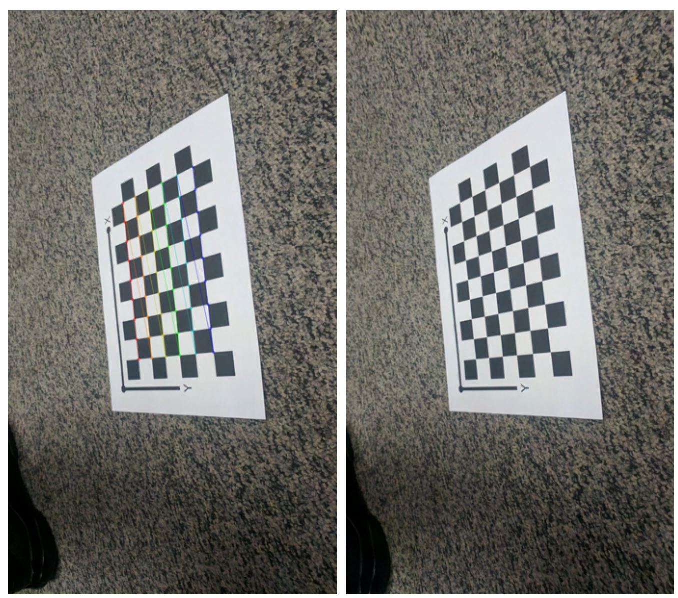

Fig. 13: Image 13 and Distortion Corrected Image 13