CMSC733: AutoCalib

Abhiskek Kathpal
M.Eng., Robotics
University of Maryland, College Park
Email: akathpal@terpmail.umd.edu
USING 3 LATE DAYS

***Abstract*—The task of this homework is to implement Camera Calibration technique proposed by Zhang's paper. 13 images of Checkerboard pattern (9x6 - excluding boundary) is used for this calibration. Using equations from Zhang's paper, intrinsic and extrinsic parameters are estimated. The parameters are optimized using a non-linear geometric error minimization. The results obtained from the above approach is analysed and re-projection error and final K matrix are shown.**

I. Camera Calibration

The pipeline used for this homework is given in Zhang's paper. It involves the following steps:
1. Corner Detection of Checkerboard pattern
2. Estimate Intrinsic Camera Matrix
3. Estimate Camera Extrinsics
4. Approximate distortion coefficient
5. Non-Linear Geometric error minimization
These steps are described in detail in the next sections.

A. Checkerboard Corner Detection

The checkerboard pattern used for this camera calibration technique is 9x6 excluding the borders and is shown in Figure 1. OpenCV inbuilt function [**cv2.findChessboardCorners**] is used for finding the corners for checkerboard pattern. The output is shown in Figure 2.

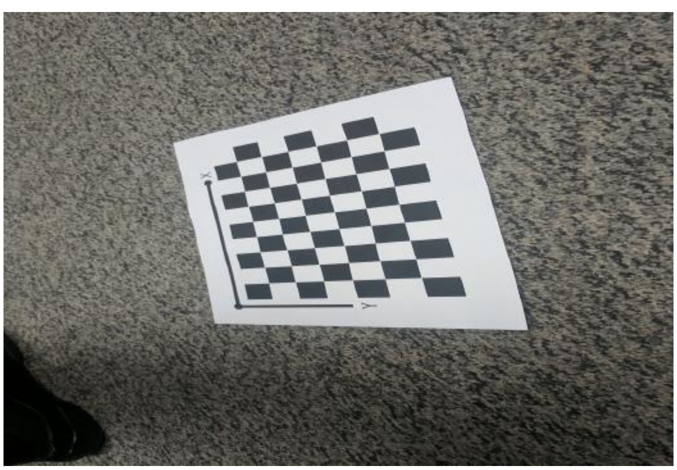


Fig. 1. Input checkerboard pattern

The object coordinates in 3d are defined by taking Z-coordinate as zero and the axis is assigned to one of the corner at edge. The points are assigned using meshgrid of 9x6 and multiplying the points with 21.5mm. Although length of the box does not matter.

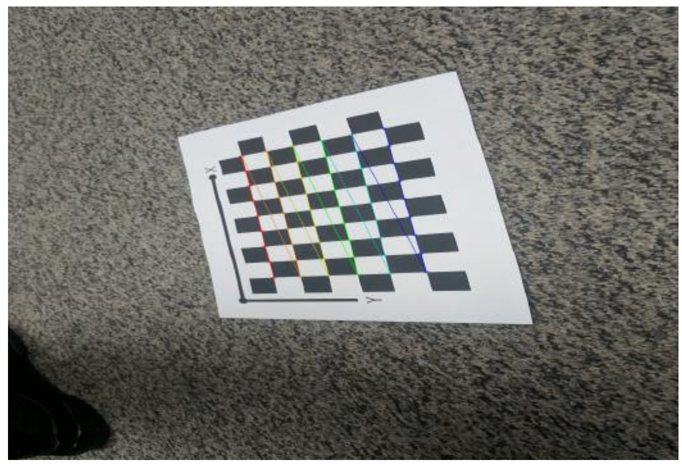


Fig. 2. Corner Detection

B. Estimating Camera Intrinsic Matrix

The relationship between a 3D point and its image projection is given by://

$$s\widetilde{\mathbf{m}} = \mathbf{K}[\mathbf{R} \quad \mathbf{t}]\widetilde{M} \tag{1}$$

In the above equation, K is camera matrix and it is given by://

$$\mathbf{K} = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

Our goal is to find these parameters. The initial estimation of these parameters are computed using a closed form solution from section 3.1 of Zhang's paper.//

$$\mathrm{B} = \mathrm{K}^{-T}\mathrm{K}^{-1} \equiv \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{12} & B_{22} & B_{23} \\ B_{13} & B_{23} & B_{33} \end{bmatrix} \tag{3}$$

$$\begin{bmatrix} \frac{1}{\alpha^2} & -\frac{\gamma}{\alpha^2\beta} & \frac{v_0\gamma-u_0\beta}{\alpha^2\beta} \\ -\frac{\gamma}{\alpha^2\beta} & \frac{\gamma^2}{\alpha^2\beta^2}+\frac{1}{\beta^2} & -\frac{\gamma(v_0\gamma-u_0\beta)}{\alpha^2\beta^2}-\frac{v_0}{\beta^2} \\ \frac{v_0\gamma-u_0\beta}{\alpha^2\beta} & -\frac{\gamma(v_0\gamma-u_0\beta)}{\alpha^2\beta^2}-\frac{v_0}{\beta^2} & \frac{(v_0\gamma-u_0\beta)^2}{\alpha^2\beta^2}+\frac{v_0^2}{\beta^2}+1 \end{bmatrix} \tag{4}$$

$$\mathbf{b} = [B_{11}, B_{12}, B_{22}, B_{13}, B_{23}, B_{33}]^T \tag{5}$$

$$\mathbf{h}_i^T \mathbf{B} \mathbf{h}_j = \mathbf{v}_{ij}^T \mathbf{b} \tag{6}$$

$$\begin{aligned} \mathbf{v}_{ij} = [& h_{i1}h_{j1}, h_{i1}h_{j2} + h_{i2}h_{j1}, h_{i2}h_{j2} \\ & h_{i3}h_{j1} + h_{i1}h_{j3}, h_{i3}h_{j2} + h_{i2}h_{j3}, h_{i3}h_{j3}]^T \end{aligned} \tag{7}$$

$$\begin{bmatrix} \mathbf{v}_{12}^T \\ (\mathbf{v}_{11} - \mathbf{v}_{22})^T \end{bmatrix} \mathbf{b} = \mathbf{0} \tag{8}$$

$$\begin{aligned} v_0 &= (B_{12}B_{13} - B_{11}B_{23}) / \left(B_{11}B_{22} - B_{12}^2\right) \\ \lambda &= B_{33} - \left[B_{13}^2 + v_0 (B_{12}B_{13} - B_{11}B_{23})\right] / B_{11} \\ \alpha &= \sqrt{\lambda / B_{11}} \\ \beta &= \sqrt{\lambda B_{11} / \left(B_{11}B_{22} - B_{12}^2\right)} \\ \gamma &= -B_{12}\alpha^2\beta/\lambda \\ u_0 &= \gamma v_0/\beta - B_{13}\alpha^2/\lambda \end{aligned} \tag{9}$$

Initial estimation of camera paramters using the above approach is given by:

$$\begin{pmatrix} 2.04201e+03 & -4.4197 & 7.7597e+02 \\ 0 & 2.0274e+03 & 1.3412e+03 \\ 0 & 0 & 1 \end{pmatrix} \tag{10}$$

C. Estimate Camera extrinsics

After computing the homographies using opencv inbuilt function and intrinsic camera parameters, we can estimate the extrinsic parameters i.e. Rotation and translation vectors. These are computed using the following equations with h1,h2,h3 are columns of homography matrix for respective image:

$$\begin{aligned} \mathbf{r}_1 &= \lambda \mathbf{A}^{-1} \mathbf{h}_1 \\ \mathbf{r}_2 &= \lambda \mathbf{A}^{-1} \mathbf{h}_2 \\ \mathbf{r}_3 &= \mathbf{r}_1 \times \mathbf{r}_2 \\ \mathbf{t} &= \lambda \mathbf{A}^{-1} \mathbf{h}_3 \end{aligned} \tag{11}$$

Using the above equations, mean re-projection error is computed and is equal to **0.93339**

D. Optimization and computing distortion coefficient kc

The next step after computing the extrinsic and intrinsic parameters is to do optimization. Intially, value of radial distortion is (0,0). To minimize the reprojection error, scipy.optimize is used. Reprojection error is given by following equation: //

$$\sum_{i=1}^{n} \sum_{j=1}^{m} \left\| \mathbf{m}_{ij} - \hat{\mathbf{m}}\left(\mathbf{K}, \mathbf{R}_i, \mathbf{t}_i, \mathrm{M}_j\right) \right\|^2 \tag{12}$$

After the optimization, the distortion coefficients come out to be: k1 = **0.04909** k2 = **-0.3353**

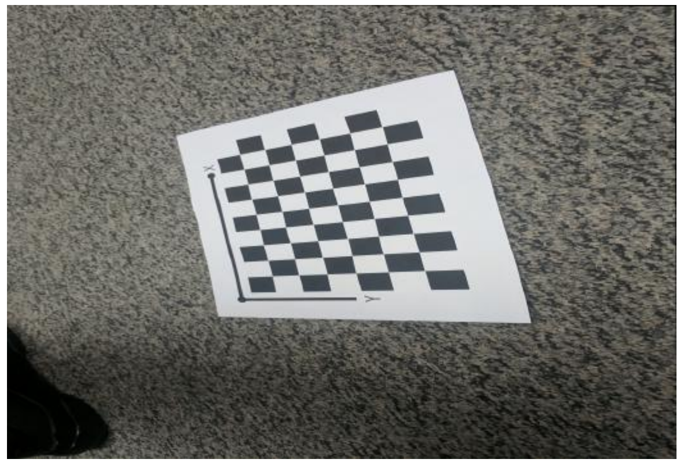
E. Results

After optimization,The final camera matrix and radial distortion coefficients are given below:

$$\begin{pmatrix} 2.24928e+03 & -1.0001e+03 & 5.57418e+02 \\ 0 & 2.332e+03 & 1.39937e+03 \\ 0 & 0 & 1 \end{pmatrix} \tag{13}$$

k1 = **0.04909** k2 = **-0.3353**

Using the above parameters, the corner points are reprojected , both the corner points and reprojected points are shown in the given figure.The output after rectification is shown below as you can see there is very less difference in the reprojected points and original corner points.



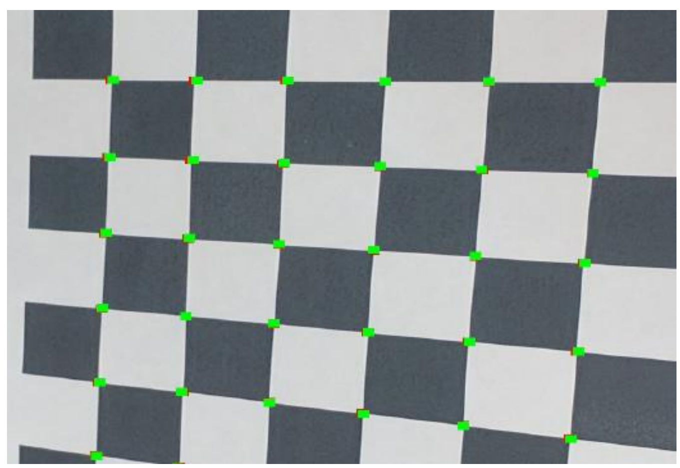


Fig. 3. Re-projected Points

The mean reprojection error after optimization comes out to be: **0.884955**

II. Conclusion

In this report, procedure for calibrating a camera using checkerboard images is discussed and described in detail. The final re-projection error and K intrinsic camera matrix is reported. Non-Linear geometric error minimization is also performed to get the optimal parameters.

REFERENCES

[1] Project Description: https://cmsc733.github.io/2019/hw/hw1/

[2] Zhang, Zhengyou. ”A flexible new technique for camera calibration.” IEEE Transactions on pattern analysis and machine intelligence 22 (2000).

[3] https://kushalvyas.github.io/calib.html