RBE549 Computer Vision : Homework 1

Thabsheer Jafer Machingal
Robotics Engineering (M.S))
tmachingal@wpi.edu
Subbmitted on: October 10 2022

I. Introduction

Homework1: Autocalib is to implement a camera calibration algorithm as described in Zhang's paper[1]. The task is to estimate intrinsic and extrinsic parameters of the camera. For calibration, we are given 13 images of a checker board printed on a4 paper and shot using Google Pixel XL, with focus locked on a specific target for all the 13 images.

II. Pipeline

A. Data

The calibration target used can be found online at [2]. The calibration target. This was printed on an A4 paper and the size of each square was 21.5mm. The Y axis has odd number of squares and X axis has even number of squares. And the outer squares and neglected as a common practice.

B. Detect Corners

Initial step is to detect corners of the squares in the board. We will detect 54 squares and its corners (6x9). As suggested, I used inbuilt functions in openCV to implement this part.

C. Estimate homography

Second step is to find the homography between the world coordinates XY and the corners detected in the image.

D. Initial Parameter Estimation

With the results from above two steps, the idea is to obtain a good initial estimate of the parameters and feed it to a Non-linear optimizer and try and reduce the error thus find a true or precise estimate of the intrinsic and extrinsic parameters.

1) Intrinsic parameters: To find the Camera intrinsic parameters, solve the following equation(1).

$$A^T A = B \tag{1}$$

Here A is the Camera instrinsic matrix and B is given as Fig.1 in Zhang's paper[1].

A can be decomposed into:

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

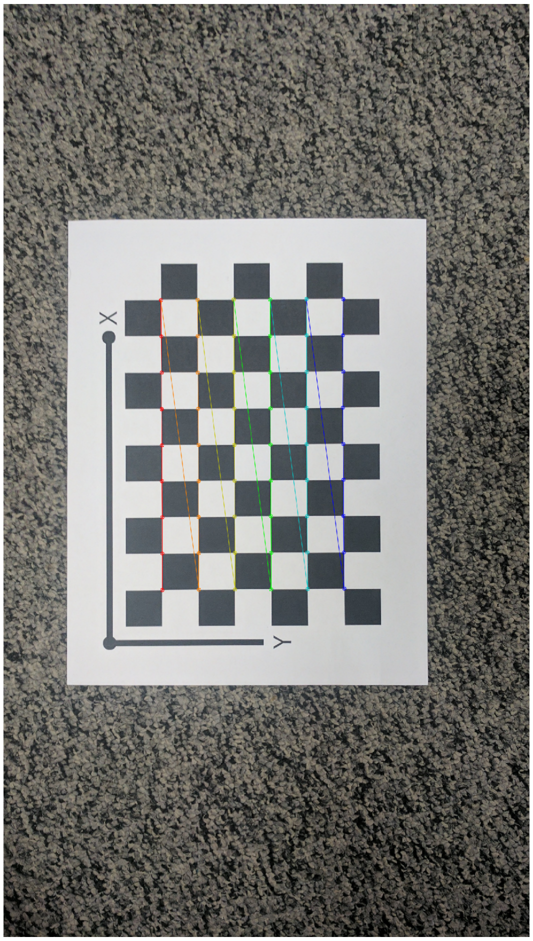


Fig. 1. Detected corners in the chessboard

$$\mathbf{B} = \mathbf{A}^{-T}\mathbf{A}^{-1} \equiv \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{12} & B_{22} & B_{23} \\ B_{13} & B_{23} & B_{33} \end{bmatrix}$$
$$= \begin{bmatrix} \frac{1}{\alpha^2} & -\frac{\gamma}{\alpha^2\beta} & \frac{v_0\gamma-u_0\beta}{\alpha^2\beta} \\ -\frac{\gamma}{\alpha^2\beta} & \frac{\gamma^2}{\alpha^2\beta^2}+\frac{1}{\beta^2} & -\frac{\gamma(v_0\gamma-u_0\beta)}{\alpha^2\beta^2}-\frac{v_0}{\beta^2} \\ \frac{v_0\gamma-u_0\beta}{\alpha^2\beta} & -\frac{\gamma(v_0\gamma-u_0\beta)}{\alpha^2\beta^2}-\frac{v_0}{\beta^2} & \frac{(v_0\gamma-u_0\beta)^2}{\alpha^2\beta^2}+\frac{v_0^2}{\beta^2}+1 \end{bmatrix}.$$

Fig. 2. Elements of B matrix

2) Extrinsic parameters: With the initial estimate of the intrinsic parameters we can estimate the extrinsic parameters that is rotational andtranslational vectors.

$$r1 = \lambda A^{-1} h_1 \tag{3}$$

$$r2 = \lambda A^{-1} h_2 \tag{4}$$

$$r3 = r1 X r3 \tag{5}$$

$$t = \lambda A^{-1} h_3 \tag{6}$$

Here, $\lambda = \frac{1}{||A^{-1}h_1||} = \frac{1}{||A^{-1}h_2||}$

3) Distortion Coefficients: Distortion coefficients are K_C = [k1,k2], which is initialized as [0,0]

E. Nonlinear geometric optimization

Using initial guess of each parameters we can optimize them. For this section I used least square error method, which is inbuilt in scipy library. Thus we can get a good estimate of each parameters, K,K_C and extrinsic parameters.

III. Results

A. Initial Parameter Estimation

1) Camera Intrinsic matrix (K): The initial guess of the Camera Intrinsic matrix, K is found to be:

$$\begin{bmatrix} 2.03475X10^3 & 4.93135X10^{-1} & 7.727044X10^2 \\ 0 & 2.01790X10^3 & 1.36090X10^3 \\ 0 & 0 & 1 \end{bmatrix}$$

B. Optimized Parameters

1) Camera Intrinsic matrix (K):

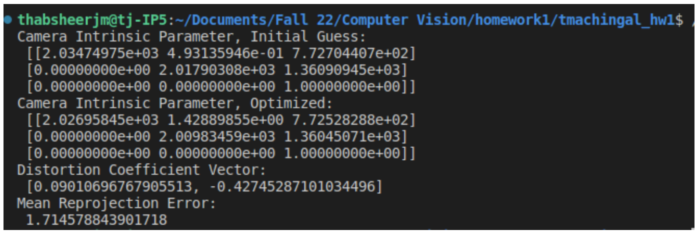
$$\begin{bmatrix} 2.02695X10^3 & 1.4288985 & 7.72528X10^2 \\ 0 & 2.00983X10^3 & 1.36045X10^3 \\ 0 & 0 & 1 \end{bmatrix}$$

2) Distortion coefficient vector:

$$K_C = \begin{bmatrix} 0.0901069676 & -0.427452871 \end{bmatrix}$$

3) Distortion coefficient vector:

$$error = 1.7457884$$



```
• thabsheerjm@tj-IP5:~/Documents/Fall 22/Computer Vision/homework1/tmachingal_hw1$
 Camera Intrinsic Parameter, Initial Guess:
  [[2.03474975e+03 4.93135946e-01 7.72704407e+02]
  [0.00000000e+00 2.01790308e+03 1.36090945e+03]
  [0.00000000e+00 0.00000000e+00 1.00000000e+00]]
 Camera Intrinsic Parameter, Optimized:
  [[2.02695845e+03 1.42889855e+00 7.72528288e+02]
  [0.00000000e+00 2.00983459e+03 1.36045071e+03]
  [0.00000000e+00 0.00000000e+00 1.00000000e+00]]
 Distortion Coefficient Vector:
  [0.09010696767905513, -0.42745287101034496]
 Mean Reprojection Error:
  1.714578843901718
```

Fig. 3. Screenshot of the results (Camera Parameters)

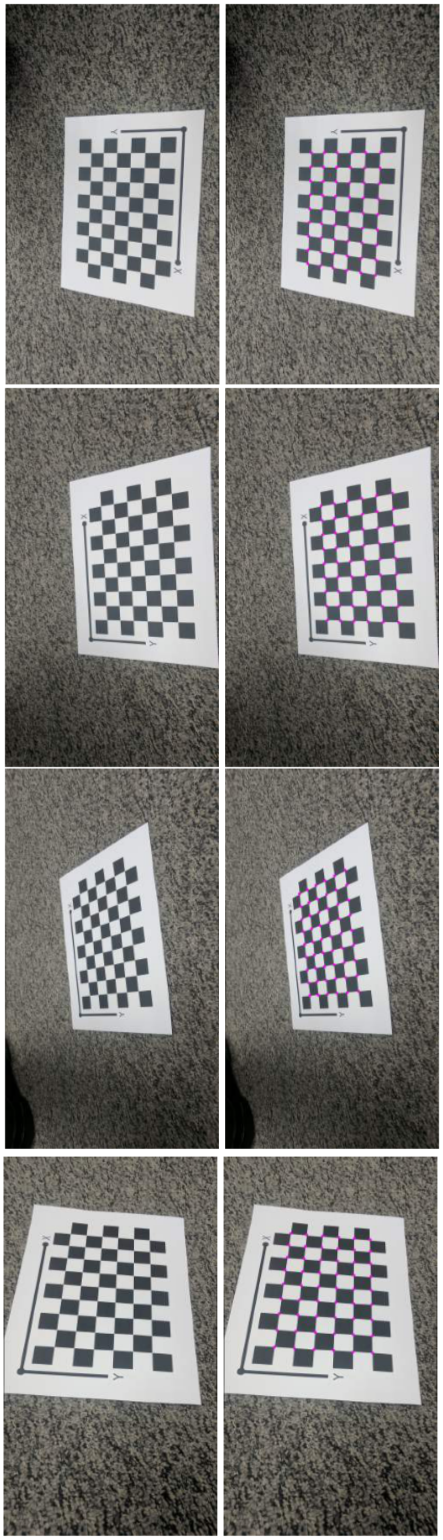


Fig. 4. Corrected image and corners projected on the image side by side

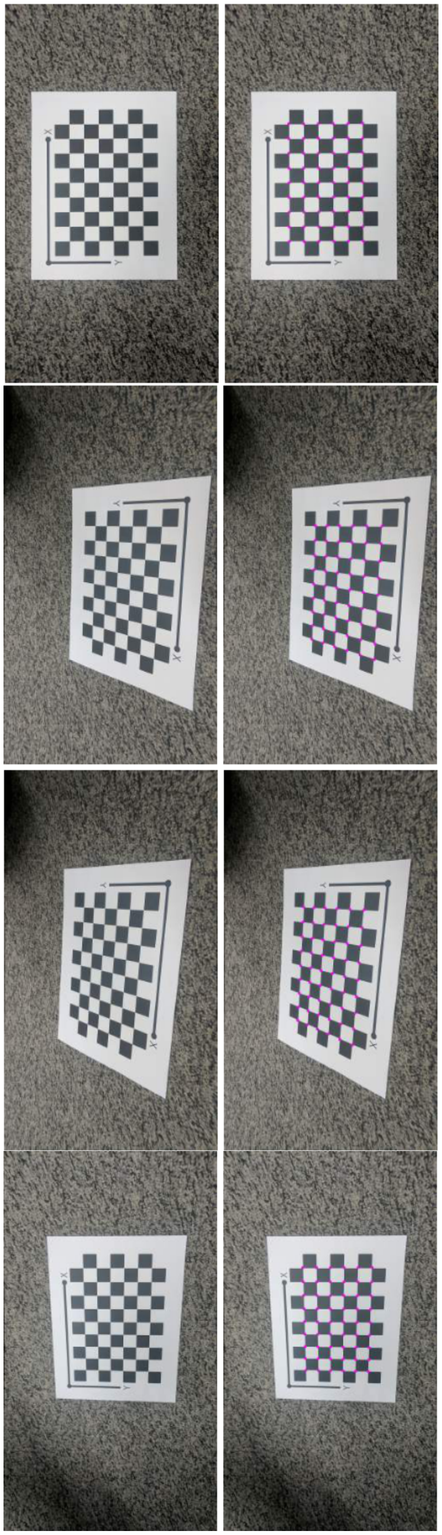


Fig. 5. Corrected image and corners projected on the image side by side

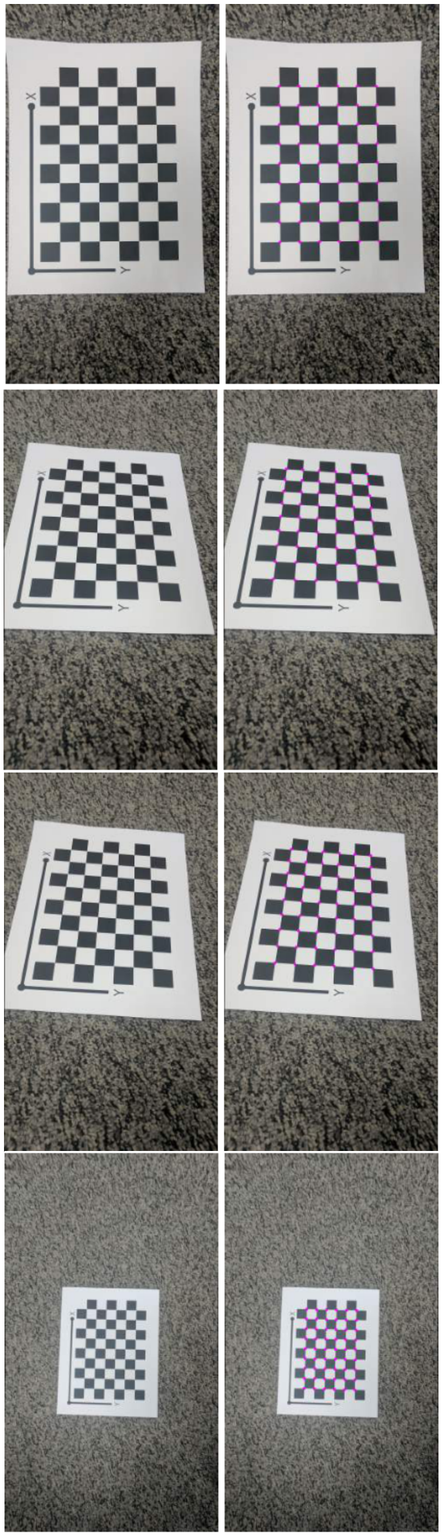


Fig. 6. Corrected image and corners projected on the image side by side

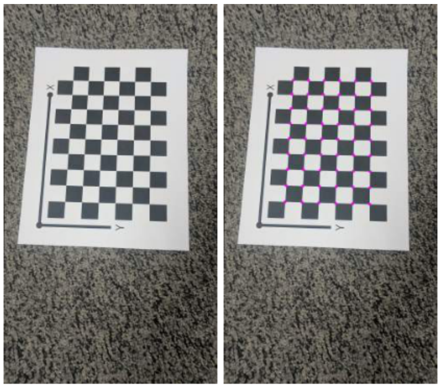


Fig. 7. Corrected image and corners projected on the image side by side

C. Corrected image and corners projected on first five sample images

Fig.4 to Fig.7 shows the reults of the calibration pipeline

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] pdf, 2019, https://raw.githubusercontent.com/cmsc733/cmsc733.github.io/master/assets/2019/hw1/checkerboardPattern.pdf.