Computer Vision - Homework 1 - AutoCalib

Muhammad Sultan
Worcester Polytechnic Institute
Worcester, MA, USA
Robotics Engineering
Email: msultan@wpi.edu

***Abstract*—In this homework, we implement the paper, "A Flexible New Technique for Camera Calibration", by Zhengyou Zhang. It involves determining the intrinsic and extrinsic camera parameters, and distortion coefficients.**

I. Introduction

We are given thirteen images of a chessboard pattern from different angles, for the calibration task. The length and width of each square in the chess board is given to be 21.5 mm. Our goal is to estimate the intrinsic and extrinsic camera parameters, and radial distortion coefficients using these images. The steps involved in this process are:

1) Estimation of Intrinsic Parameters
2) Estimation of Extrinsic Parameters
3) Non-linear Geometric Error Minimization

II. Estimation of Intrinsic Parameters

The intrinsic parameters of a camera are shown in matrix A, given below.

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

In the matrix, α and β are scaling factors in u and v axes of the image, γ is the parameter that describes the skewness of the images axes, and $(u_0,\, v_0)$ are coordinates of the Principal Point.

To obtain this matrix, we first need the homographies between the images and the model plane. This is done by using the $cv2.findChessboardCorners$ function to detect pixel coordinates of the corners in the chessboard pattern. These coordinates are then used along with the corresponding model plane coordinates to compute the homographies. These homography matrices are a combination of both the intrinsic and extrinsic parameters, and hence need to be decomposed to obtain those parameters. Assuming that the model plane lies on Z = 0, the relation between the model plane and the pixel image is shown below.

$$\mathbf{s} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = A \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{t} \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

$$H = A \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{t} \end{bmatrix}$$

Since r_1 and r_2, are orthonormal, we get the following two equations:

$$\mathbf{h}_1^T \mathbf{A}^{-T} \mathbf{A}^{-1} \mathbf{h}_2 = 0$$

$$\mathbf{h}_1^T \mathbf{A}^{-T} \mathbf{A}^{-1} \mathbf{h}_1 = \mathbf{h}_2^T \mathbf{A}^{-T} \mathbf{A}^{-1} \mathbf{h}_2$$

These two equations are the constraints on the intrinsic parameters, for a given homography. Using these constraints, we can get the closed-form solution to compute the intrinsic parameters. The solution is presented as follows:

$$\mathbf{B} = \mathbf{A}^{-T} \mathbf{A}^{-1} = \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \\ B_{31} & B_{32} & B_{33} \end{bmatrix}$$

B is a symmetric matrix defined by a 6D vector **b**, where $\mathbf{b} = [B_{11}, B_{12}, B_{13}, B_{21}, B_{22}, B_{23}, B_{31}, B_{32}, B_{33}]$. This vector is obtained by solving the equation:

$$\mathbf{V}\mathbf{b} = 0$$

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

$\mathbf{v}_{11}$ and $\mathbf{v}_{12}$ can be obtained using the following equation:

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

Using Singular Value Decomposition, we calculate **b**, and compute the intrinsic parameters in the following way.

$$v_0 = \frac{B_{12}B_{13} - B_{11}B_{23}}{B_{11}B_{22} - B_{12}^2}$$

$$\lambda = B_{33} - \frac{B_{13}^2 + v_0(B_{12}B_{13} - B_{11}B_{23})}{B_{11}}$$

$$\alpha = \sqrt{\frac{\lambda}{B_{11}}}$$

$$\beta = \sqrt{\frac{\lambda B_{11}}{B_{11}B_{22} - B_{12}^2}}$$

$$\gamma = -\frac{B_{12}\alpha^2\beta}{\lambda}$$

$$u_0 = \frac{\gamma v_0}{\beta} - \frac{B_{13}\alpha^2}{\lambda}$$

III. Estimation of Extrinsic Parameters

Having calculated the intrinsic parameters, and in turn matrix **A**, the next step is to use matrix **A** and the homograhpies to compute extrinsic parameters. The following equations are used:

$$r_1 = \lambda A^{-1} h_1$$

$$r_2 = \lambda A^{-1} h_2$$

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

$$\lambda = 1/||A^{-1} h_1|| = 1/||A^{-1} h_2||$$

Hence, we have extrinsic parameters as well.

IV. Non-linear Geometric Error Minimization

The next step is to minimize the following **re-projection error** to get optimal values of intrinsic and extrinsic parameters, as well as the distortion coefficients k_1 and k_2.

$$i = \sum_{i=1}^{n} \sum_{j=1}^{m} ||m_{ij} - \hat{m}(A, k_1, k_2, R_i, t_i, M_j)||$$

Here, m_{ij} are the corners obtained using $cv2.findChessboardCorners$, and $\hat{m}(A, k_1, k_2, R_i, t_i, M_j)$ are the re-projected points from the world frame to the pixel image frame.

$$\hat{u} = u + (u - u_0)[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2]$$

$$\hat{v} = v + (v - u_0)[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2]$$

Here, $m_{ij} = (u, v)$ are ideal (distortion-free pixel coordinates), while $\hat{m}(A, k_1, k_2, R_i, t_i, M_j) = (\hat{u}, \hat{v})$ are the corresponding real observed image coordinates.

To solve this optimization problem, we use all the parameters we calculated before as initial guesses. The initial guess for both k_1 and k_2 is taken as 0.

V. Results

The intrinsic matrix **A** before and after optimization is shown below, followed by the optimal distortion coefficients.

Before optimization:

$$\mathbf{A} = \begin{bmatrix} 2024.6825 & -1.0464 & 773.1396 \\ 0.0000 & 2016.7016 & 1386.5039 \\ 0.0000 & 0.0000 & 1 \end{bmatrix}$$

After optimization:

$$\mathbf{A}_{optim} = \begin{bmatrix} 2024.6793 & -1.0466 & 773.1449 \\ 0.0000 & 2016.6956 & 1386.5173 \\ 0.0000 & 0.0000 & 1 \end{bmatrix}$$

$$\mathbf{k}_{optim} = [0.010757, -0.068246]$$

The average re-projection error before optimization was **0.7647953**, while after optimization it was **0.7526295**. These errors are calculated averaging out all the errors of each corner in each image.

Figures 1 to 12 show a comparison of 6 of the thirteen images, to show the original image and the undistorted image, with the re-projected corners drawn on the undistorted image. As can be seen there is no observable difference between the two images since the camera distortion is negligible owing to very small values of k_1 and k_2.

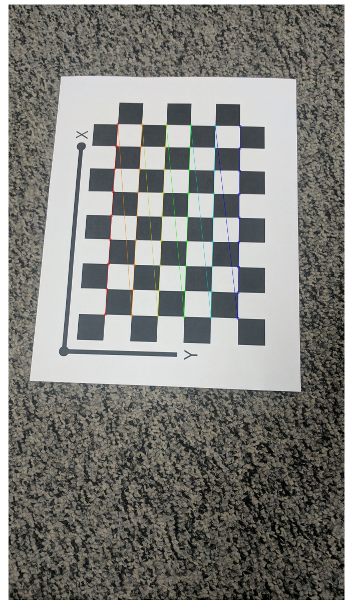

Fig. 1: Image 0: Original Image with detected corners

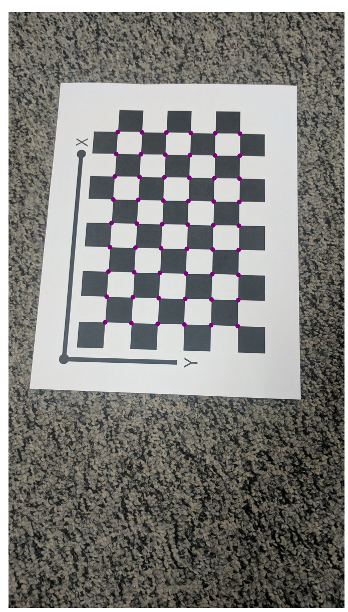

Fig. 2: Image 0: Undistorted image with re-projected corners

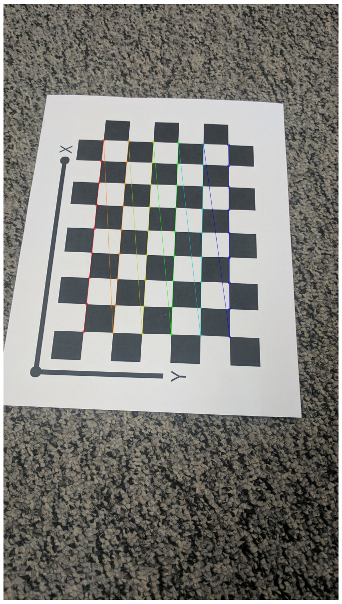

Fig. 3: Image 1: Original Image with detected corners

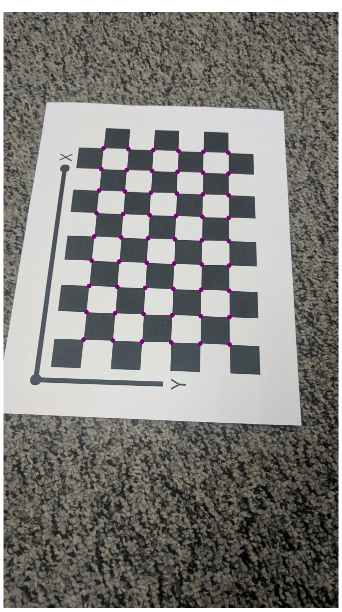

Fig. 4: Image 1: Undistorted image with re-projected corners

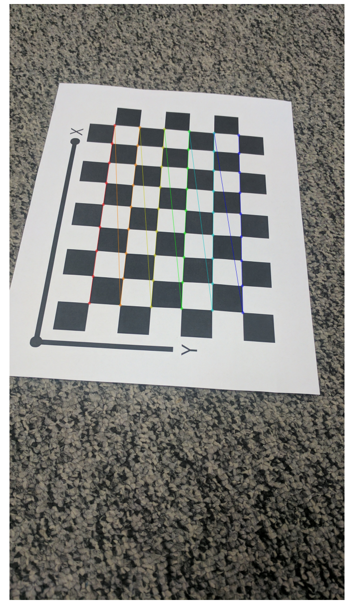

Fig. 5: Image 2: Original Image with detected corners

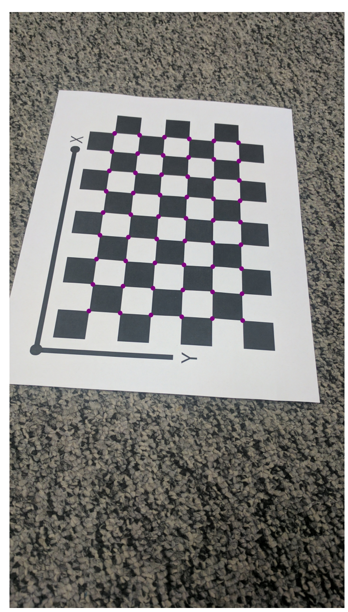

Fig. 6: Image 2: Undistorted image with re-projected corners

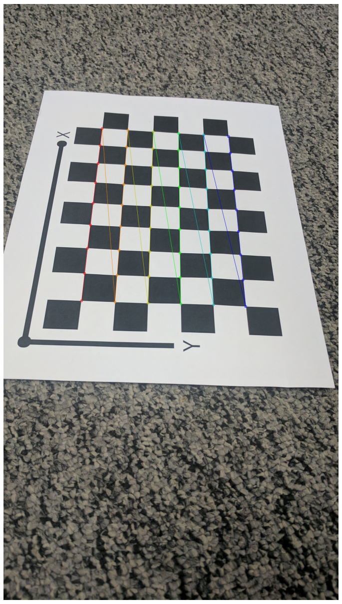

Fig. 7: Image 3: Original Image with detected corners

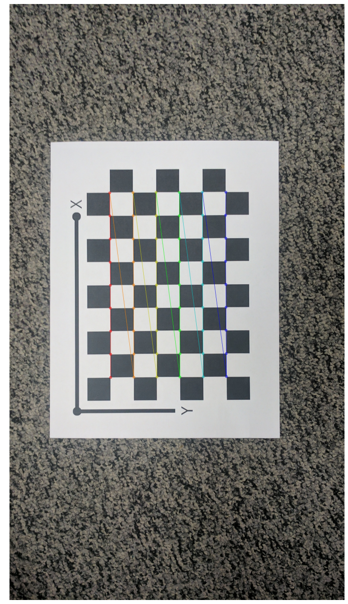

Fig. 9: Image 4: Original Image with detected corners

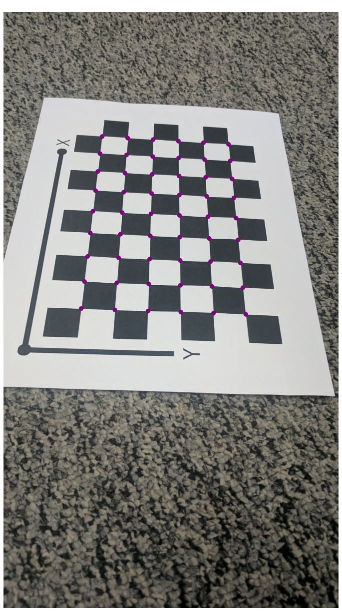

Fig. 8: Image 3: Undistorted image with re-projected corners

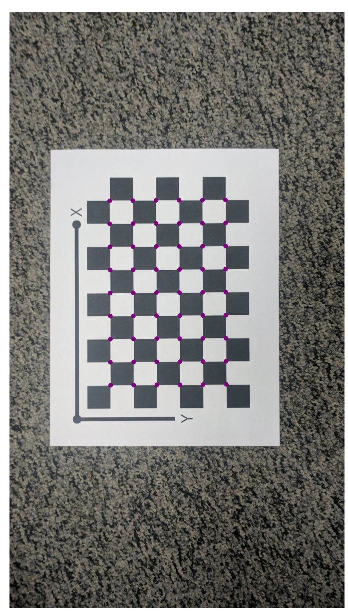

Fig. 10: Image 4: Undistorted image with re-projected corners

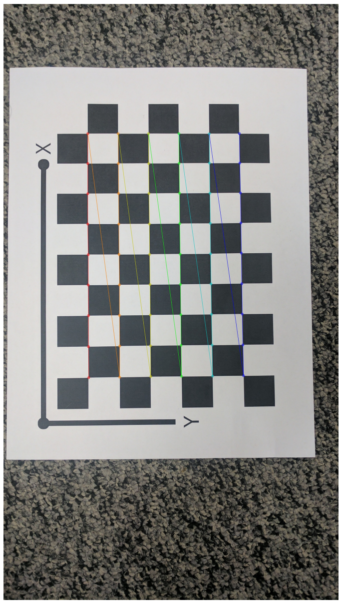

Fig. 11: Image 5: Original Image with detected corners

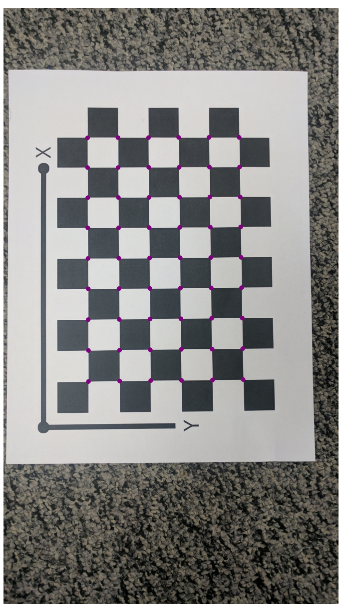

Fig. 12: Image 5: Undistorted image with re-projected corners