RBE/CS549 Computer Vision Homework 1 - AutoCalib

Sreejani Chattterjee
Email: schatterjee@wpi.edu

Abstract

In this report, we will discuss the calculation of camera calibration parameters from scratch, in step by step fashion. Camera calibration is the estimation of camera intrinsics, extrinsics and distortion parameters. Focal length and principal points comprises the intrinsic parameters, whereas distortion parameters are coefficients of distortion. Camera extrinsics is the position and orientation of the camera frame with respect to the world do-ordinates system. Camera calibration is a crucial technique in the field of Computer Vision. In this homework we will implement the most widely used camera calibration technique proposed by Zhengyou Zhang.

Index Terms-Camera Calibration, Intrinsic camera parameters, Distortion

1 INTRODUCTION

We need two kinds of transformation when we are trying to do a transformation between a real world point and the corresponding point captured in the image space. The first transformation is between the world point and the point in the image plane frame also known as Camera Extrinsics. The second is the transformation of the image plane frame to image pixel co-ordinate system, also known as Camera Intrinsics. The following matrix is the Intrinsic camera matrix.

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

Here $(\mathrm{u0}, \mathrm{v0})$ is the optical center or the principal point in pixel coordinates. (α and β), are scale factors in u and v axis and s is the skew coefficient in case the image plane frame and sensor is not perfectly aligned.

In Zhang's method, first we need to assume that the model plane is located on Z = 0. Which means that the third column of Rt matrix i.e. r3 becomes redundant so we can remove it. The following equation is the overall transformation of the world point to image pixel co-ordinates

$$
s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{A} \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{r}_3 & \mathbf{t} \end{bmatrix} \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix} \tag{2}
$$

2 Initial Parameters Estimation

We need data to start with our camera calibration process. For this homework we were given 13 images of checkerboard, taken from a Google Pixel XL smartphone with focus locked. The actual size of checkerboard square side is 21.5 mm. The checkerboard is of (7×10) size. But as a common practice we will neglect the corner columns and rows and work with size of (6×9) checkerboard.

3 Solving for approximate K or camera intrinsic matrix

The following steps are to come up with an initial guess of the intrinsic parameters.

3.1 Find Checkerboard Corners:

We started with finding the corners of the checkerboard in world co-ordinates as well as pixel coordinates. Used cv2.findChessboardCorners() function to find 54 corners in every image. And then calculated the corresponding real world co-ordinates with the given size of the square. Every corner found on the checker board will give us an equation of the form as shown in Eq. 2. Each image has 54 corners. Fig. 1 are the original corners detected in the 13 images.

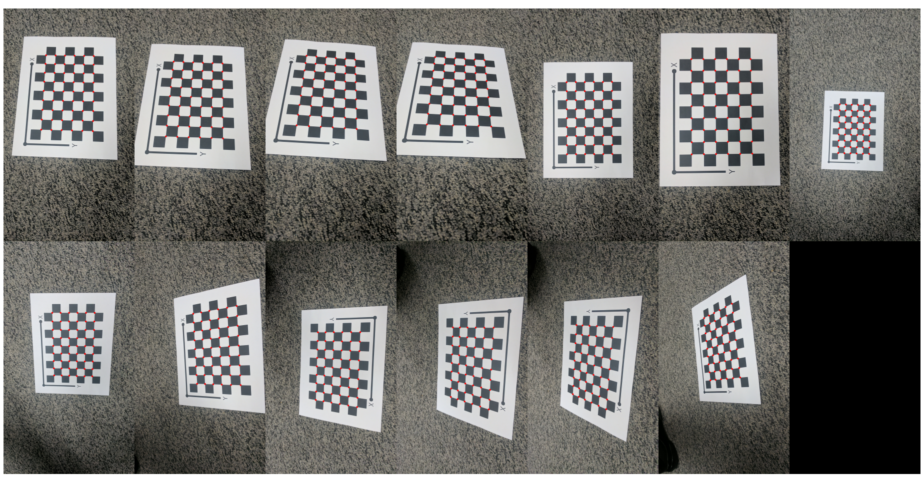

Fig. 1: 54 Corners found in the Each Image.

3.2 Find Homography Matrix:

The next step is to calculate the overall transformation matrix or Homography matrix (consists of both A and Rt matrix) for each image. Eq.2 is the Homography equation.

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \underset{3\times 3}{\mathrm{H}} \begin{bmatrix} X_i \\ Y_i \\ 1 \end{bmatrix} \tag{3}$$

To calculate the homography, we need to implement Singular value Decomposition (SVD) after rearranging the problem into $Ax = 0$ form. We can get our solution by solving the below set of equations.

$$\begin{aligned} \boldsymbol{a}_{x_i}^{\top} \boldsymbol{h} &= 0 \\ \boldsymbol{a}_{y_i}^{\top} \boldsymbol{h} &= 0 \\ \boldsymbol{a}_{x_i}^{\top} &= (-X_i, -Y_i, -1, 0, 0, 0, x_i X_i, x_i Y_i, x_i) \\ \boldsymbol{a}_{y_i}^{\top} &= (0, 0, 0, -X_i, -Y_i, -1, y_i X_i, y_i Y_i, y_i) \end{aligned}$$

Notice that the vectorized homography matrix h has 8 unknowns in so a minimum of eight equations are reuired to solve all unknowns. For 8 equations we need 4 corner points with known world and image co-ordinates. After SVD, we need to normalize all the terms such that the last element of the h matrix becomes 1 .

3.3 Decomposition of Homography Matrix:

After obtaining Homography matrix, we need to decompose it into two individual matrices namely. These matrices are the camera intrinsic and extrinsic matrix. We need to form a new homogeneous linear system to solve for the matrix A. To formulate such a system, we need exploit the constraints of the Rt matrix. We know that r1 and r2 in Rt matrix are orthonormal vectors that means their dot product is zero and their norm is equal to 1 . Using the above constraints we will get the below written equations and solve the same.

$$\begin{aligned} &\mathbf{h}_1^T \mathbf{A}^{-T} \mathbf{A}^{-1} \mathbf{h}_2 = 0 \\ &\mathbf{h}_1^T \mathbf{A}^{-T} \mathbf{A}^{-1} \mathbf{h}_1 = \mathbf{h}_2^T \mathbf{A}^{-T} \mathbf{A}^{-1} \mathbf{h}_2. \end{aligned}$$

Now, We define the B matrix as follows:

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

It should be noted that B is a symmetric and positive definite matrix and hence it has only 6 Degrees of Freedom. We need to define a linear homogenous system as follows:

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

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

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

There are 6 unknowns hence 3 different homography matrices are needed to compute the above equations. For that 3 different images of the checker board in three different planes are required.

3.4 Initial K matrix:

Once, we get the B matrix it is easy to get the K matrix by Cholesky's decomposition or we can just refer to Appendix B in Zhang's paper to find the individual elements of the matrix and construct the K matrix from those.

Fig. 2 shows the initial estimated camera intrinsic matrix (K). We will further optimize the matrix in the later sections.

```
Camera K matrix before optimization:-
K =  [[ 2.46363273e+03 -5.61941998e-01  7.62798541e+02]
 [ 0.00000000e+00  2.44450584e+03  1.35164446e+03]
 [ 0.00000000e+00  0.00000000e+00  1.00000000e+00]]
```

Fig. 2: K Matrix before optimization

4 Approximating Camera Extrinsic Matrix

The camera extrinsic parameters can be estimated by the following equations. The computed Rt matrix was not able to satisfy the properties of a Rotation matrix possibly due to the noise in the data.

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

5 Approximating Distortion Co-efficients k

We assumed that we have minimum camera distortion so we can take $[0, 0]$T as our initial guess for the distortion coefficients. Please note that we are considering only the first two terms for calculating distortion and hence two parameters. For better results we can use more parameters.

Later when we take distortion in consideration we use below equations to calculate the pixel co-ordinates. Please note that here we have assumed $\gamma = 0$ i.e. there is no skewness in the image frame and the sensor.

$$\breve{u} = u + (u - u_0)\left[k_1\left(x^2 + y^2\right) + k_2\left(x^2 + y^2\right)^2\right]$$
$$\breve{v} = v + (v - v_0)\left[k_1\left(x^2 + y^2\right) + k_2\left(x^2 + y^2\right)^2\right].$$

6 Optimization

So far we have estimated the required Intrinsic, Extrinsic and Distortion parameters. The solution for Intrinsic parameters is obtained by minimizing an algebraic distance which has no physical connotation. Hence we will do a Maximum Likelihood Estimation inference to get the exact solution. We need to update the distortion co-efficients too. The following equation shows how we can do a Maximum Likelihood Estimation on the entire set of parameters at one go.

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\left\|\mathbf{m}_{ij} - \breve{\mathbf{m}}\left(\mathbf{A}, k_1, k_2, \mathbf{R}_i, \mathbf{t}_i, \mathrm{M}_j\right)\right\|^2$$

Here, the least square distances between actual corner pixel co-ordinates and the projected co-ordinates (from estimated parameters) are minimized using scipy.optimize.least_squares function.

7 Results

Fig. 3 shows the distortion coefficients after optimization, Fig. 4 shows the final K matrix and Fig. 5 shows the Reprojection error all post optimization.

```
Optimized Distortion Coefficients:
[k1, k2] =  [ 0.01292523 -0.12347749  0.          0.          0.        ]
```

Fig. 3: Optimized Distortion coefficients

```
Optimized Camera intrinsics (K matrix):-
K =  [[ 2.46362635e+03 -5.62999602e-01  7.62808559e+02]
 [ 0.00000000e+00  2.44449505e+03  1.35166029e+03]
 [ 0.00000000e+00  0.00000000e+00  1.00000000e+00]]
```

Fig. 4: Optimized K matrix

```
Mean Error calculation and Reprojection error:-
Reprojection Error:  0.7532661004965734
```

Fig. 5: Re-projection Error

Fig. 6. shows reprojected corners on undistorted image for all 13 images.

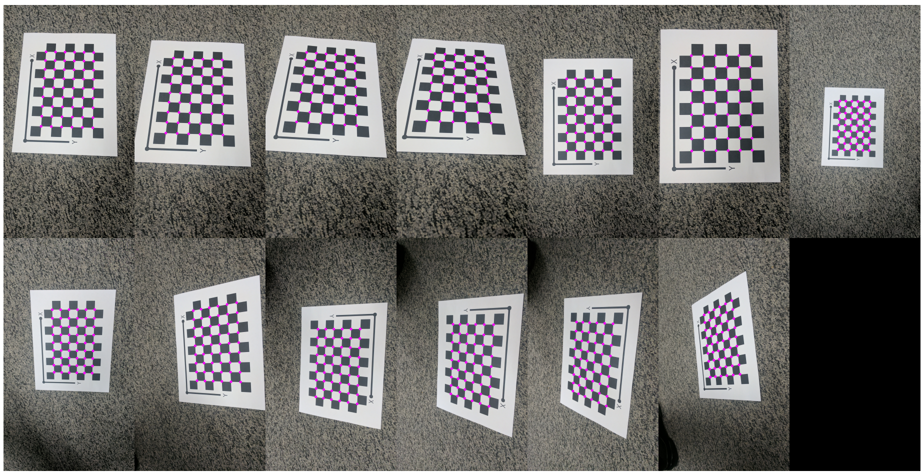

Fig. 6: Reprojected corners on Undistorted Images.

8 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. https://docs.opencv.org/4.x/dc/dbb/tutorial_py_calibration.html