HW1: AutoCalib

Dhiraj Kumar Rouniyar
Email: dkrouniyar@wpi.edu

***Abstract*—The aim is to estimate the camera intrinsic, extrinsic parameters and radial distortion parameters using Zhengyou Zhang's method. The Zhang's paper relies on a calibration target (checkerboard in our case) to estimate camera intrinsic parameters. The calibration target used is a checkerboard.**

I. Data preparation and Initial parameter estimation

A. Introduction

For any research in computer vision where 3D geometry is involved, camera calibration is an important task. The camera calibration matrix K is given by the following matrix:

$$\begin{bmatrix} f_x & 0 & c_x \\ a & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

where f_x, f_y are focal lengths or scale factors, c_x, c_y are principle points, skew is zero and radial distortions is k_1, k_2. The steps are followed as follows:
1. First we approximate the (K) camera intrinsic matrix.
2. Using that (K), we estimate camera's extrinsic parameters Rotation(R) and Translation(t).
3. Then Non linear optimization is performed to minimize the geometric error.

B. Data preparation and Initial parameter estimation

In this work, the checkerboard taken as a calibration target has 9 inner rows and 6 inner columns and the squares pattern has size of 21.5 mm. The pictures was clicked on Google Pixel XL phone camera with locked focus and printed on A4 paper. The sample of calibration target is shown in Fig 1.

Firstly, the pixel coordinates of each chessboard image corners are found using cv2.findChessboardCorners function. And considering (9,6) patternsize corresponding to the inner corners a total of 54 corner points are found for each image. Then, we define a 3D coordinate system for the detected 54 points. Here, the Z coordinate is kept 0 since the picture is in 2D plane. The relationship between the image coordinate and the 3D world coordinate simplifies to:

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \begin{bmatrix} r_1 & r_2 & r_3 & t \end{bmatrix} \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix} \tag{1}$$

Where s is a scale factor, r_i is the i^{th} rotation column vector and t is the translation vector. Based on the given square size of 21.5 mm, the coordinate system is as follows:

$$[0, 0, 0], [21.5, 0, 0], [0, 21.5, 0],$$

$$[21.5, 21.5, 0], ...[172, 0, 0]...,$$

$$[172, 21.5, 0], ...[0, 107.5, 0], ...[172, 107.5, 0]$$

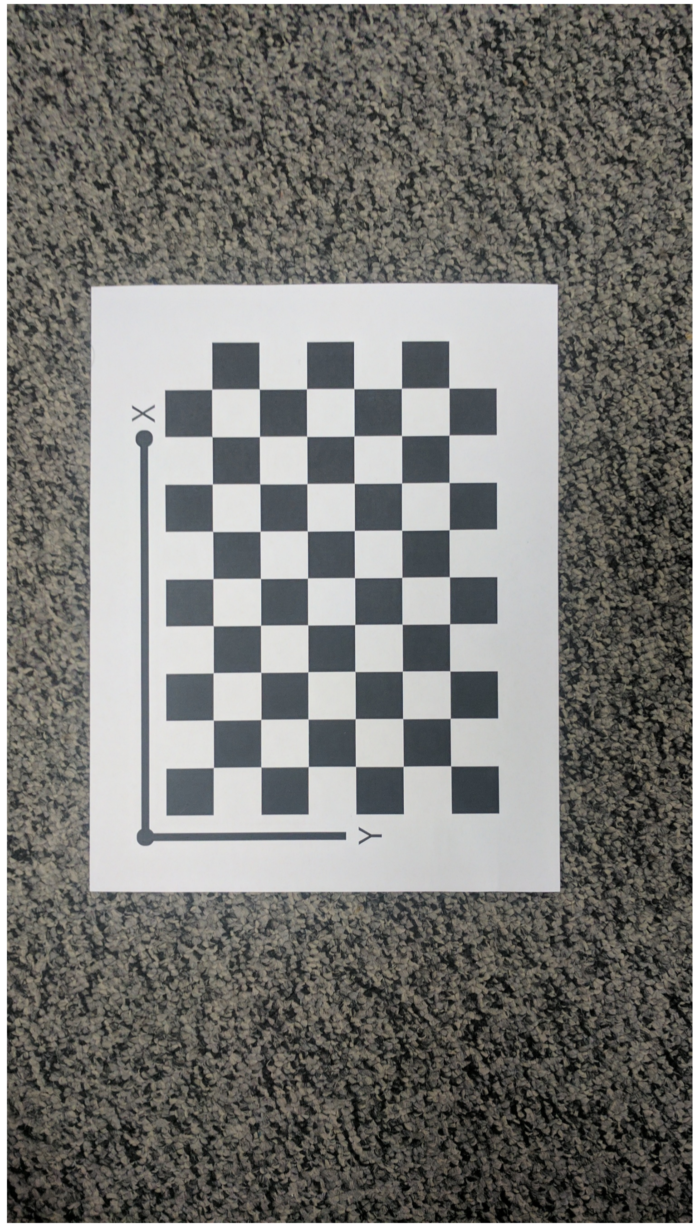

Fig. 1. Calibration target

Here, we get the mapping of 3D coordinates and 2D pixel coordinates. Then, we find the homography between the 3D and 2D correspondence using direct linear transform. Each corresponding pair can be used to create 2x9 matrix as shown:

$$p_i = \begin{bmatrix} -x_i & -y_i & -1 & 0 & 0 & 0 & x_i x_i' & y_i x_i' & x_i' \\ 0 & 0 & 0 & -x_i & -y_i & -1 & x_i y_i' & y_i y_i' & y_i' \end{bmatrix}$$

Multiple such points here 8 is stacked to get P. Then we solve PH=0 to get H,
Then with the singular value decomposition (SVD) of P is done. Once homography is obtained the matrix V is updated:

$$V_{ij} = \begin{bmatrix} h_{i1}h_{j1}, h_{i1}h_{j2} + h_{i2}h_{j1}, h_{i2}h_{j2}, \\ h_{i3}h_{j1} + h_{i1}h_{j3}, h_{i3}h_{j2} + h_{i2}h_{j3}, h_{i3}h_{j3} \end{bmatrix}^T$$

where i^{th} column og the homography matrix is

$$h_i = \begin{bmatrix} h_{i1} & h_{i2} & h_{i3} \end{bmatrix}^T$$

Then we represent 2 homogeneous equation in b and for n images, we stack n such equations and the system is solved for b using SVD of V. Then after we get b we solve the intrinsic patameters of the camera matrix using the below:

$$v_0 = (B_{12}B_{13} - B_{11}B_{23})/(B_{11}B_{22} - B_{12}^2)$$
$$\lambda = B_{33} - [B_{13}^2 + v_0(B_{12}B_{13} - B_{11}B_{23})]/B_{11}$$
$$\alpha = \sqrt{\lambda/B_{11}}$$
$$\beta = \sqrt{\lambda B_{11}/(B_{11}B_{22} - B_{12}^2)}$$
$$\gamma = -B_{12}\alpha^2\beta/\lambda$$
$$u_0 = \gamma v_0/\beta - B_{13}\alpha^2/\lambda$$

The output is obtained as below:

```
u = 751.628803047355
v = 1338.634696458544
lamda = -0.6401102612852156
alpha = 2060.7377160788924
beta = 2045.6533993021096
gamma = -4.240086367836726
```

Fig. 2. Calibration target

The initial estimation of Calibration matrix:

$$\begin{bmatrix} 2.06 & -4.24 & 7.51 \\ 0 & 2.04 & 1.33 \\ 0 & 0 & 1 \end{bmatrix}$$

The extrinsic parameters for each image can be computed using the following equations given below:

$$r_1 = \lambda A^{-1} h_1$$
$$r_2 = \lambda A^{-1} h_2$$
$$r_3 = r_1 \times r_2$$
$$t = \lambda A^{-1} h_3$$

II. Non-Linear geometric error minimization

Using equation [1] the error is calculated for each pixel coordinates of each image and the error is obtained by taking the twelve norm of the resulting vector which is same as executing below over all points for all images.

$$\sum_{i=1}^{n}\sum_{j=1}^{m} |m_{ij} - \hat{m}(A, R, t_i, M_j)|^2,$$

The errors after all points and on images is summed up and is divided by (54*11) for mean reprojection error which is **1.249**. The error can be decreased we need to target the error with radial distortion which is then summed up over all points of images to get the mean reprojection error and is divided by (54*11) as done previously. This error minimization is done by scipy.optimize.least_squares. The output for calibration matrix after optimization is:

$$\begin{bmatrix} 2.96 & -1.81 & 9.27 \\ 0 & 1.04 & 1.39 \\ 0 & 0 & 1 \end{bmatrix}$$

The output for the reprojected points plotted on the original images are attached here where red denoted reprojected points and blue denotes original corners.

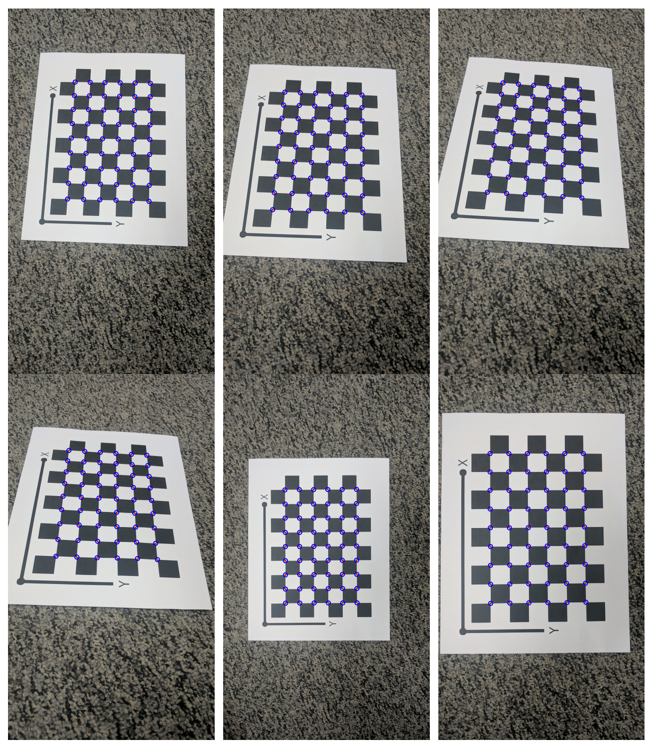

Fig. 3. Output images with corners and reprojected points

Please refer output folder for rest of the images.

III. Conclusion

The technique specified in Zhang's paper was successfully studied and implemented to calibrate a monocular camera with a reprojection error of **1.294**.