CMSC 733 Homework 1 Report
Auto Calibration

Darshan Shah
M. Eng Robotics
University of Maryland - College Park, Maryland 20740
Email: dshah003@umd.edu

I. Introduction

Camera calibration is a very important process in the field of Computer vision and photogrammetry. It is a process of estimating a camera's intrinsic and extrinsic parameters. The intrinsic parameters deal with the camera's internal characteristics such as focal length, skew, distortion, and image center. Extrinsic parameters describe its position and orientation in the world. Knowing intrinsic parameters is an essential first step for 3D computer vision, as it allows you to estimate the scene's structure in Euclidean space and removes lens distortion, which degrades accuracy.

The early methods of camera calibration uses DLT to solve the projection matrix and assumes the control points do not lie in the same plane. This method was quite tedious and had it's shortcomings. This implementation however, is based on the more recent and widely used method proposed by Zhengyou Zhang in his paper "A flexible new technique for camera calibration" [1]. This method makes use of a pictures of a checkerboard pattern of known dimensions each having different views.

The process of estimating intrinsic parameters involves the following steps: a. Establishing correspondences between image points and world points. b. for each established correspondences, compute the associated homography matrix. c. Calculate the intrinsic parameters $\alpha, \beta, \gamma, u_c, v_c$ from the estimated homographies. and finally d. Update the parameters using LM-Optimizer and further refine all the parameters.

II. Camera Calibration

A. Calibration Data

We use the checkerboard as shown in figure 1 for our project. A dataset of images of this checkerboard in 13 different views is used.

B. Point Correspondences

The corners of the checkerboard pattern object in world frame is known to us, hence we manually create the list of world points by multiplying integral multiples of square pattern size. To find the checkerboard corner points, each image is read and the corner points are estimated using the opencv's `cv2.findChessboardCorners` function. The corners are saved in a list for further use.

The figure 2 shows the detected points in the checkerboard.

C. Estimate Homography

The next step in the algorithm is to estimate homographies for each of the given views.s Before we do that, the corresponding points need to be normalized around their mean. This makes sure that we get a finite solution in the next steps of estimating the Homographies using Direct Linear Transformation.

A homography matrix is the transformation matrix which converts points from one coordinate space to another. $m = HM$ where m is the object points, H is the homography matrix and M is the image points. Section 2.2 from [1] describes in detail the computation of Homography matrix. To obtain the homography matrix, we rearrange and create the system of equations such that $Ax = 0$ one of the solutions of the equation is $x = 0$ however, that is not of much use to us. Hence we try to approximate Ax 0 we use Null space vector of A such that $||Ax||^2 -> min$. The solution of this system is computed using SVD (Singular Value Decomposition).

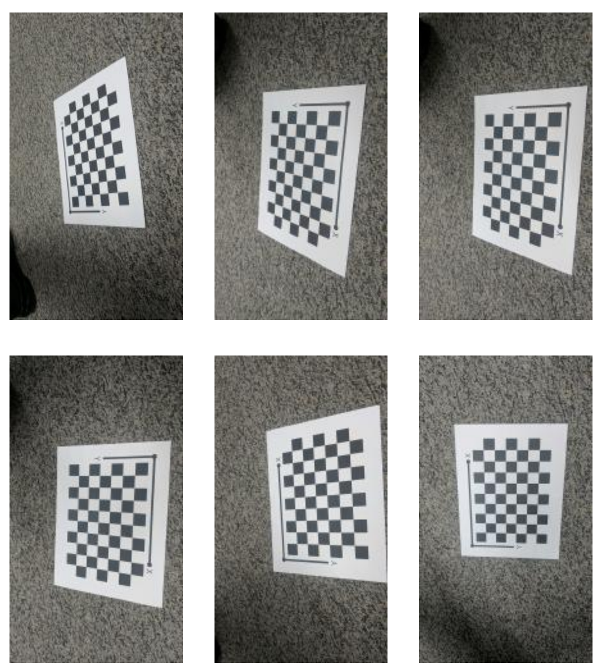

Fig. 1. Calibration images

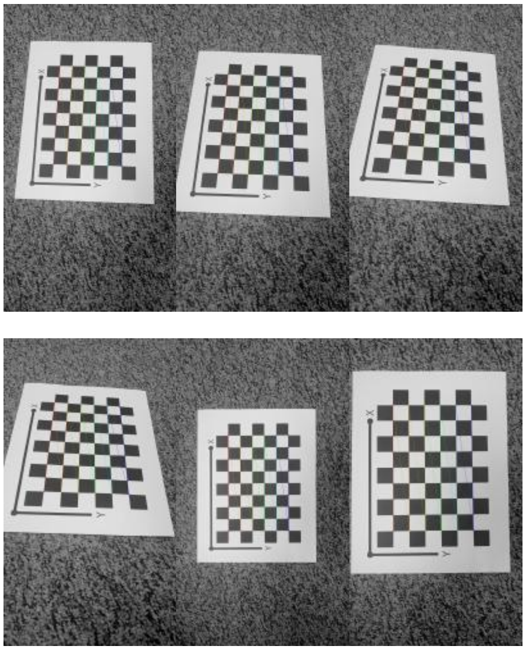

Fig. 2. Point correspondences

D. Refining Homographies

To further refine homography obtained for each view, we use Levenberg Marquadt Optimizer. which is a non linear optimization technique. The `optimize` function from the `scipy` library is used to implement LM-optimizer.

E. Extract Intrinsic parameters

The homography matrix is given by

$$H = A.[R|t]$$

where A is the matrix having all the intrinsic parameters. Also referring to section 2.2 and equation 2 from Zhengyou's paper,

$$H = A[r_1 r_2 t]$$

To extract the intrinsic parameters from the obtained homography matrix, we exploit the constraints that (a) r_1 and r_2 are orthonormal (b) dot product of r_1 and r_2 is zero.

Following the steps in section 3.1 of [1], we get $V.b = 0$ which again is of the form Ax = 0. and this solution is again obtained computing the SVD of V. SVD of V gives us b and hence B. Once we compute B, getting the intrinsic parameters is very straightforward. (Refer Appendix B of [1]). Finally putting all together, we get the intrinsic parameter matrix A.

F. Extract Extrinsic Parameters

Once A is known, the extrinsic parameters for each images are calculated as per Section 3.1 from [1]
$r_1 = \lambda A^{-1} h_1$
$r_2 = \lambda A^{-1} h_2$
$r_3 = r_1 * r_2$

Fig. 3. View 1

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

Each of these rotation matrices and translation matrix are appended to their respective List and finally printed out.

III. Result

The resultant Intrinsic parameters for the given dataset assuming the square pattern size as Unity, we get the following intrinsic parameters:
α = 2057.5956
β = 2044.57
γ = -0.6683
u_c = 764.017
v_c = 1360.7609

The matrix A is obtained as follows:

2.05759563e+03	-6.68390012e-01	7.64017398e+02
0.00000000e+00	2.04457744e+03	1.36076094e+03
0.00000000e+00	0.00000000e+00	1.00000000e+00

By setting the square size as 21.5 mm, we get the following as Matrix A:

$$A = \begin{array}{ccc} 2.05096497e+03 & -1.65537119e+00 & 7.63008462e+02 \\ 0.0 & 2.03572688e+03 & 1.35894496e+03 \\ 0.0 & 0.0 & 1.0 \end{array}$$

The Rotation matrix for the first view, as shown in fig 3 is given as,

$$R = \begin{array}{ccc} -8.24880393e-05 & -2.29274826e-03 & 2.52790732e-06 \\ 2.49732834e-03 & -1.41733142e-04 & -1.78237950e-07 \\ 1.13925584e-04 & 1.00577895e-03 & 5.73743649e-06 \end{array}$$

The Translation Matrix for first view is

$$t = \begin{array}{c} 0.16020379 \\ -\ 0.15174719 \\ 1. \end{array}$$

References

[1] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(11):1330–1334, November 2000.