Homework 1 - AutoCalib

Ashwin Varghese Kuruttukulam
UID: 115906518
Email: ashwinvk@terpmail.umd.edu

***Abstract*—In this project we estimate the calibration parameters of a camera. A series of images taken of a checkerboard of known square size is used as input. The project follows the pipeline described in [1] to get an initial estimate of the calibration parameters and then refine it through maximum likelihood inference.**

I. Method

Firstly, the corner pixels of the checkerboard in each input image has to identified. This was achieved using the opencv function ***findChessboardCorners***. An interesting observation was that each corner was uniquely identified in each image with respect to the axis provided on the checkerboard.

Following this, the homography transformation the image plane and model plane is estimated. The position of the corners in the model plane is calculated from the actual square size of the checkerboard. This can be done, by rearranging the corresponding pixels in the equation given below:

$$\begin{bmatrix} x_1x'_1 & x_1y'_1 & x_1 & y_1x'_1 & y_1y'_1 & y_1 & x'_1 & y'_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_mx'_m & x_my'_m & x_m & y_mx'_m & y_my'_m & y_m & x'_m & y'_m & 1 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{21} \\ f_{31} \\ f_{12} \\ f_{22} \\ f_{32} \\ f_{13} \\ f_{23} \\ f_{33} \end{bmatrix} = 0$$

Fig. 1: Homography estimation

And then obtaining the last column on the V matrix obtained SVD decomposition on the left matrix. This same approach is implemented in the opencv function ***findHomography***

As mentioned in [1], these homography matrices are used to form the matrix V as shown below:

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

Fig. 2: V matrix

In order to make sure that the estimated homography matrices were correct, we transformed the raw image using this homography and checked whether it looks like a rectangle as shown below.

The null space solution of the the matrix V is then found using SVD decomposition as mentioned before, to obtain matrix B. The calibration matrix is then calculated from B using the equation below:

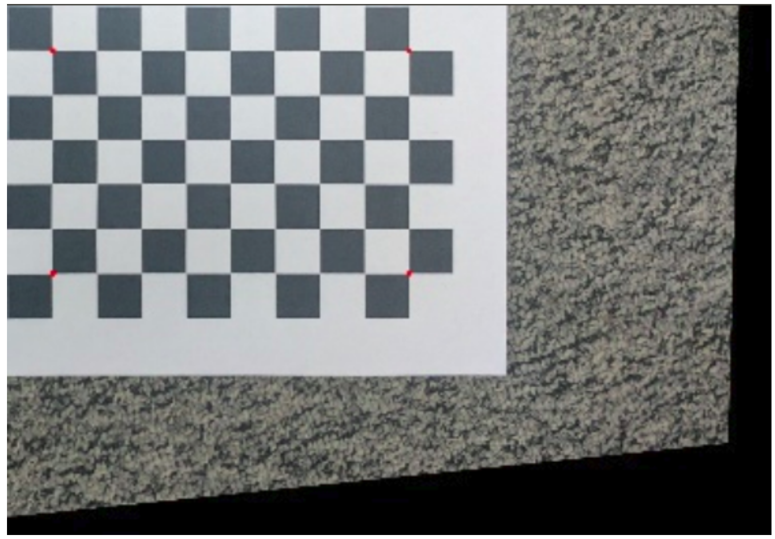

Fig. 3: Reprojected raw image

As mentioned in [1], these homography matrices are used to form the matrix V as shown below:

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

Fig. 4: Calibration matrix calculation

The calibration matrix is given by:

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

Fig. 5: Calibration matrix

The extrinsic matrix can then be calculated using the below equation: This is the rotation and transformation between the frame at the pinhole camera and the frame on the physical checkerboard board.

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

Fig. 6: Extrinsic calibration

The maximum likelihood estimation was performed using the scipy library function, ***optimize.least_squares*** In this process, the following function was minimized

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\|\mathbf{m}_{ij}-\hat{\mathbf{m}}(\mathbf{A},\mathbf{R}_i,\mathbf{t}_i,\mathtt{M}_j)\|^2\,,$$

Fig. 7: Minimization function

In order to get a performance estimate, the RMS error in reprojection was calculated.

Using this calibration matrix and distortion values, we distorted the image using the opencv function ***cv2.undistort***

II. Results

```
The intial estimate of the calibration matrix:
[[2.03475109e+03 4.92810405e-01 7.72703920e+02]
 [0.00000000e+00 2.01790464e+03 1.36090928e+03]
 [0.00000000e+00 0.00000000e+00 1.00000000e+00]]
```

Fig. 8: Initial intrinsic calibration matrix estimate

```
The maximum likelihood intrisic calibration matrix
[[2.06611250e+03 0.00000000e+00 7.62040481e+02]
 [0.00000000e+00 2.04908357e+03 1.38075816e+03]
 [0.00000000e+00 0.00000000e+00 1.00000000e+00]]
```

Fig. 9: Maximum likelihood intrinsic calibration matrix

References

[1] Zhengyou Zhang. 2000. A Flexible New Technique for Camera Calibration. IEEE Trans. Pattern Anal. Mach. Intell. 22, 11 (November 2000), 1330–1334. DOI:https://doi.org/10.1109/34.888718

```
The distortion parameters
[[ 0.05795539]
 [-0.25490134]]
```

Fig. 10: Maximum likelihood distortion parameters

```
('RMS Error', 2.9490720265565757)
```

Fig. 11: RMS error

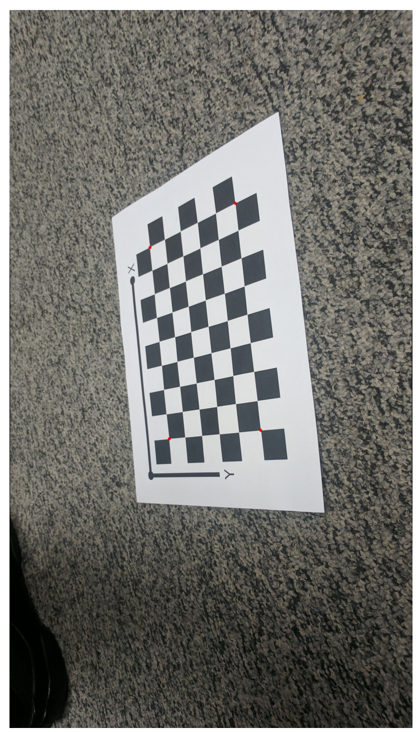

Fig. 12: Undistorted image 0 with corner

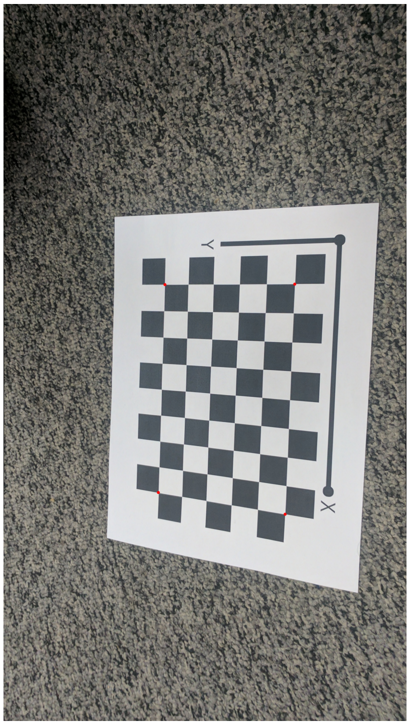

Fig. 13: Undistorted image 1 with corner

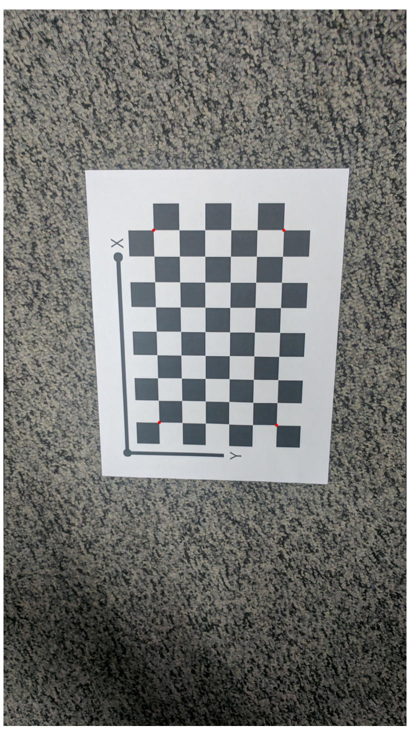

Fig. 14: Undistorted image 2 with corner

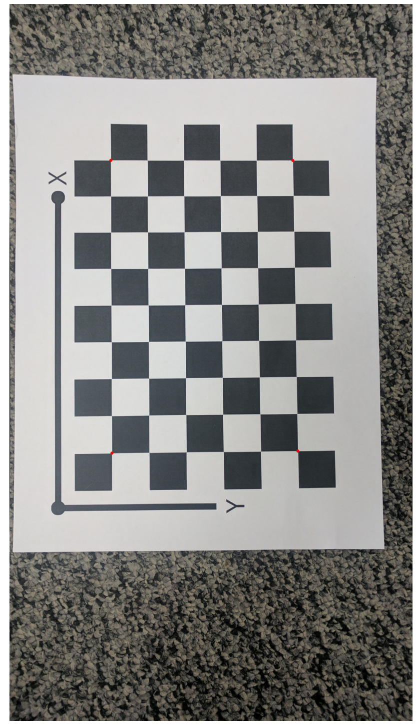

Fig. 15: Undistorted image 3 with corner

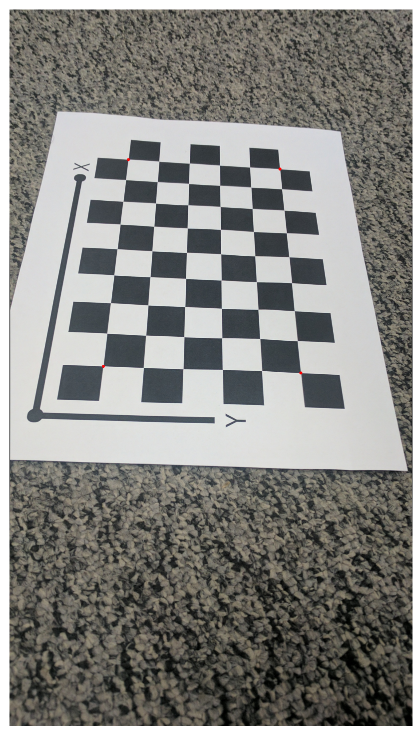

Fig. 16: Undistorted image 4 with corner