University Jean Monnet

Faculty of Science and Technology

Computer Vision - Lab 2 Report

Author:
Allwyn Joseph
Rohil Gupta

Supervisor:
Ms. Rada Deeb

April 15, 2018

Contents

List of Figures

1 Introduction

Geometric camera calibration is also referred to as camera re-sectioning which estimates the parameters of a lens and image sensor (of an image or video camera). These parameters are used to correct the lens distortion, measure the size of an object in world units, or determine the location of the camera in the scene. These tasks are used in applications in various areas such as machine vision (to detect and measure objects), robotics, navigation systems and 3D scene reconstruction.

The camera calibration toolbox in MATLAB can be used for tasks such as corner extraction, calibration, calibration using Zhengyou Zhang's data, calibration using Bakstein and Halir's data, calibrating a stereo system and stereo image rectification and 3D stereo triangulation.

Single Camera Calibration: The Single Camera calibration involves loading calibration images, extracting image corners, running the main calibration engine, displaying the results, controlling accuracies, adding and suppressing images, un-distorting images, exporting calibration data to different formats.

Calibrate the stereo system: Calibrating a stereo system (intrinsically and extrinsically), rectifying stereo images and performing 3D stereo triangulation.

2 Objectives

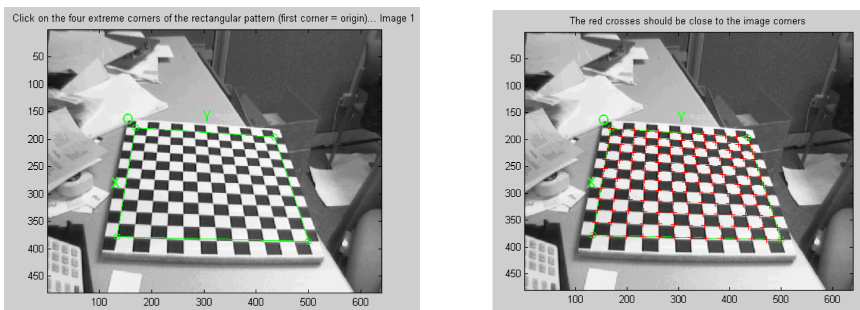
The lab session had us tasked with three main objectives:

- Task 1: Learn the usage of the calibration toolbox.
- Task 2: Calibrate a single camera.
- Task 3: Calibrate the stereo system

3 Task 1

Keeping in line with the objectives of the lab session, we first familiarized ourselves with the usage of the calibration and stereo toolbox.
The following are the steps followed to do the same:

- Step 1 - the images were loaded and read onto variables `I_1` up to `I_20`.
- Step 2 - the grid corners of the images were then extracted. This was done by strategically clicking at the four corners of an image to setup a calibration grid boundary as see in figure(1a). The program then automatically proceeds to count the number of squares in both dimensions, and shows the predicted grid corners as seen in figure(1b). The image corners are then automatically extracted, and displayed on figure 1.



(a) Boundary of the calibration grid

(b) Predicted grid corners in absence of distortion

Figure 1: After selecting the four corners.

- Step 3 - after corner extraction as seen in figure 2, the main camera calibration procedure is run. The calibration is done in two steps: first initialization (computes a closed-form solution for the calibration parameters) and nonlinear optimization (to minimizes the total re-projection error over all the calibration parameters).
- Step 4 - The calibration results then ought to be analyzed and ameliorated. From figure 3 below, we notice that the results after optimization have rather high error values. In the following steps we try and correct for the same.
- Step 5 - To correct for the errors, the corners are recomputed and calibration is conducted again. The results of which are as seen in figure 4, where we observer smaller error ranges and more robust values.

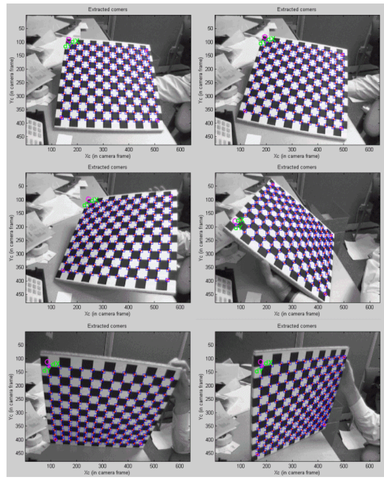
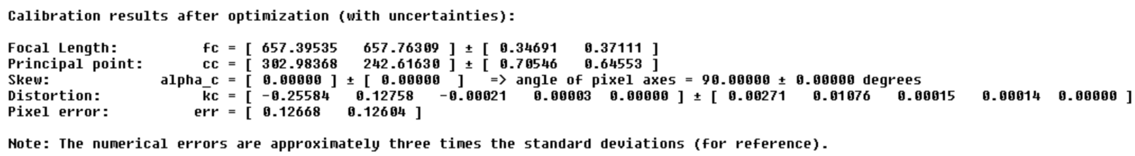


Figure 2: Results after first calibration step

```
Calibration parameters after initialization:

Focal Length:          fc = [ 671.13759   680.77186 ]
Principal point:       cc = [ 319.50000   239.50000 ]
Skew:             alpha_c = [ 0.00000 ]   => angle of pixel = 90.00000 degrees
Distortion:            kc = [ 0.00000   0.00000   0.00000   0.00000   0.00000 ]

Main calibration optimization procedure - Number of images: 20
Gradient descent iterations: 1...2...3...4...5...6...7...8...9...10...11...done
Estimation of uncertainties...done


Calibration results after optimization (with uncertainties):

Focal Length:          fc = [ 661.67001   662.82858 ] ± [ 1.17913   1.26567 ]
Principal point:       cc = [ 306.09590   240.78987 ] ± [ 2.38443   2.17481 ]
Skew:             alpha_c = [ 0.00000 ] ± [ 0.00000  ]   => angle of pixel axes = 90.00000 ± 0.00000 degrees
Distortion:            kc = [ -0.26425   0.22645   0.00020   0.00023  0.00000 ] ± [ 0.00934   0.03826   0.00052   0.00053  0.00000 ]
Pixel error:          err = [ 0.45330   0.38916 ]
```

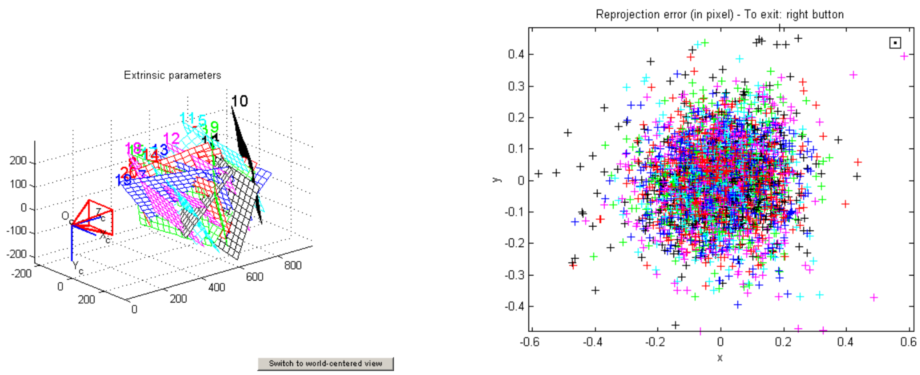
Figure 3: Results after second calibration step

- Step 6 - After having corrected for the errors we can then proceed to visualize and study the extrinsic parameters figure 5a along with the re-projection errors figure 5b. The errors in this case are lower since a second calibration step was effectuated. Once robust results are obtained the parameters should be saved as a `Calib_Results.mat` file for future stereo calibration.
- Step 7 - The above calibration steps ought to be repeated for camera the second camera too.
- Step 8 - For the stereo calibration step, the `Calib_Results.mat` file for the two cameras would be required.



```
Calibration results after optimization (with uncertainties):

Focal Length:          fc = [ 657.39535   657.76309 ] ± [ 0.34691   0.37111 ]
Principal point:       cc = [ 302.98368   242.61630 ] ± [ 0.70546   0.64553 ]
Skew:             alpha_c = [ 0.00000 ] ± [ 0.00000  ]   => angle of pixel axes = 90.00000 ± 0.00000 degrees
Distortion:            kc = [ -0.25584   0.12758   -0.00021   0.00003  0.00000 ] ± [ 0.00271   0.01076   0.00015   0.00014  0.00000 ]
Pixel error:          err = [ 0.12668   0.12604 ]

Note: The numerical errors are approximately three times the standard deviations (for reference).
```

Figure 4: Extracted Corners



(a) Extrinsic Features

(b) Re-projection error

Figure 5: Extrinsic features and Re-projection error

- Step 9 - Once the necessary files are obtained, they are loaded and a stereo calibration step is effectuated to obtain results as in figure 6. If results from stereo calibration and single camera calibration were to be compared, one would observe a decrease in uncertainties within the stereo scenario. This is attributed to the minimal set of unknown parameters over which the stereo calibration was performed.

- Step 10 - The extrinsic parameters can then we visualized, as seen in figure 7.

- Step 11 - Finally the the images used for stereo calibration are rectified such that the epipolar lines match with the horizontal scanned lines.

```
Stereo calibration parameters after optimization:


Intrinsic parameters of left camera:

Focal Length:          fc_left = [ 533.52331   533.52699 ] ± [ 0.83147   0.84055 ]
Principal point:       cc_left = [ 341.60376   235.19287 ] ± [ 1.23937   1.20470 ]
Skew:             alpha_c_left = [ 0.00000 ] ± [ 0.00000  ]   => angle of pixel axes = 90.00000 ± 0.00000 degrees
Distortion:            kc_left = [ -0.28838   0.09714   0.00109   -0.00030  0.00000 ] ± [ 0.00621   0.02155   0.00028   0.00034  0.00000 ]


Intrinsic parameters of right camera:

Focal Length:          fc_right = [ 536.81377   536.47649 ] ± [ 0.87631   0.86541 ]
Principal point:       cc_right = [ 326.28657   250.10121 ] ± [ 1.31444   1.16609 ]
Skew:             alpha_c_right = [ 0.00000 ] ± [ 0.00000  ]   => angle of pixel axes = 90.00000 ± 0.00000 degrees
Distortion:            kc_right = [ -0.28943   0.10690   -0.00059   0.00014  0.00000 ] ± [ 0.00486   0.00883   0.00022   0.00055  0.00000 ]


Extrinsic parameters (position of right camera wrt left camera):

Rotation vector:             om = [ 0.00669   0.00452  -0.00350 ] ± [ 0.00270   0.00308  0.00029 ]
Translation vector:           T = [ -99.80198   1.12443  0.05041 ] ± [ 0.14200   0.11352  0.49773 ]
```

Figure 6: Intrinsic parameters after stereo calibration

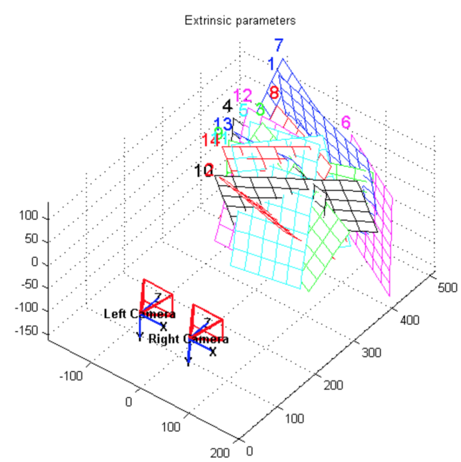


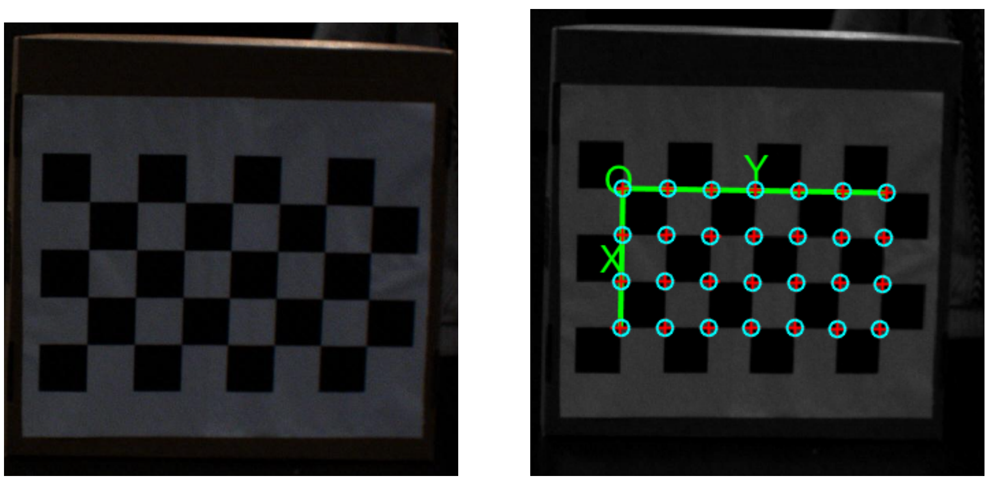
Figure 7: Extrinsic parameters after stereo calibration

4 Task 2

Once proficiency over the calibration toolbox was attained, a similar procedure was repeated over another set of images (left and right) taken by two cameras - the procedure for only the images taken by the camera has been elaborated below.

The following were calibration steps for the images taken by the right camera:

- Step 1 - the images were loaded and read onto variables `I_1` up to `I_25`.
- Step 2 - the grid corners of the images were then extracted as seen in figure 8.



(a) Undistorted image (b) Distorted image

Figure 8: An example of an undistorted and distorted image

- Step 3 - After having extracting the corners we move onto camera calibration, the results of which are seen in figure 9.
- Step 4 - We then try and reduce the uncertainties by recomputing the corners and running the calibration step once again. As one can observer from the results in figure 10, there is only a slight improve

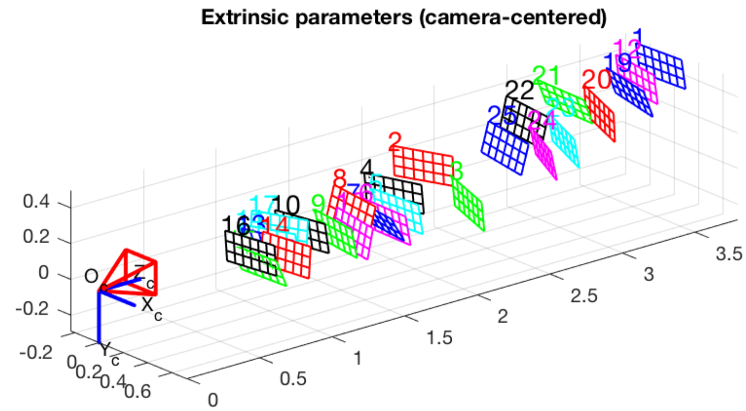
```
Calibration parameters after initialization:

Focal Length:          fc = [ 1668.71991   1668.71991 ]
Principal point:       cc = [ 811.50000   611.50000 ]
Skew:             alpha_c = [ 0.00000 ]   => angle of pixel = 90.00000 degrees
Distortion:            kc = [ 0.00000   0.00000   0.00000   0.00000   0.00000 ]

Main calibration optimization procedure - Number of images: 25
Gradient descent iterations: 1...2...3...4...5...6...7...8...9...10...11...12...13...14...15...16...17...18...19...20...21...22...23...done
Estimation of uncertainties...done


Calibration results after optimization (with uncertainties):

Focal Length:          fc = [ 1924.67329   1939.05922 ] +/- [ 31.83263   32.31420 ]
Principal point:       cc = [ 837.07923   849.56298 ] +/- [ 21.44078   16.49420 ]
Skew:             alpha_c = [ 0.00000 ] +/- [ 0.00000  ]   => angle of pixel axes = 90.00000 +/- 0.00000 degrees
Distortion:            kc = [ -0.08893   0.27001   0.02027   -0.00292  0.00000 ] +/- [ 0.02724   0.18532   0.00220   0.00282  0.00000 ]
Pixel error:          err = [ 0.27766   0.26423 ]

Note: The numerical errors are approximately three times the standard deviations (for reference).
```

Figure 9: Results after first calibration

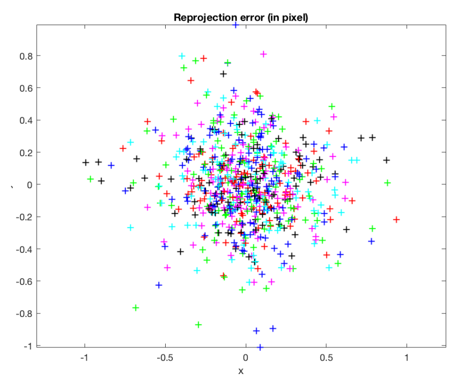
```
Main calibration optimization procedure - Number of images: 25
Gradient descent iterations: 1...2...3...4...5...6...7...8...9...10...11...12...13...14...15...done
Estimation of uncertainties...done


Calibration results after optimization (with uncertainties):

Focal Length:          fc = [ 1924.62508   1939.01049 ] +/- [ 31.81694   32.29816 ]
Principal point:       cc = [ 837.04819   849.57982 ] +/- [ 21.42984   16.48628 ]
Skew:             alpha_c = [ 0.00000 ] +/- [ 0.00000  ]   => angle of pixel axes = 90.00000 +/- 0.00000 degrees
Distortion:            kc = [ -0.08896   0.27004   0.02027   -0.00292  0.00000 ] +/- [ 0.02723   0.18519   0.00220   0.00282  0.00000 ]
Pixel error:          err = [ 0.27750   0.26415 ]

Note: The numerical errors are approximately three times the standard deviations (for reference).
```

Figure 10: Results after first calibration

- Step 5 - we then proceed to visualize the error along with the extrinsic parameters as in figure 11. Once robust results are obtained after the calibration step, the parameters are save in `Calib_Resutl_Right.mat` file.
- Step 6 - The above steps are repeated for images taken by the right camera and the parameter values are stored in a `Calib_Resutl_Left.mat` file.



(a) Extrinsic parameters



(b) Re-projection error

Figure 11: Extrinsic parameters and re-projection error

5 Task 3

Once the left and the right cameras have been calibrated we move on to stereo calibration.

- Step 1 - the `Calib_Resutl_Left.mat` and `Calib_Resutl_Right.mat` files are loaded.

- Step 2 - stereo calibration is then effectuated to deliver the results seen in figure 12.

```
Intrinsic parameters of left camera:

Focal Length:          fc_left = [ 1885.00381   1893.77120 ] ± [ 15.30702   15.84518 ]
Principal point:       cc_left = [ 828.40232   684.28415 ] ± [ 17.68853   12.36984 ]
Skew:             alpha_c_left = [ 0.00000 ] ± [ 0.00000  ]   => angle of pixel axes = 90.00000 ± 0.00000 degrees
Distortion:            kc_left = [ -0.12400   0.40111   0.00498   -0.00027  0.00000 ] ± [ 0.02015   0.12932   0.00138   0.00244  0.00000 ]


Intrinsic parameters of right camera:

Focal Length:          fc_right = [ 1892.17752   1902.29800 ] ± [ 15.63571   16.21544 ]
Principal point:       cc_right = [ 857.29234   783.94883 ] ± [ 22.53052   15.39397 ]
Skew:             alpha_c_right = [ 0.00000 ] ± [ 0.00000  ]   => angle of pixel axes = 90.00000 ± 0.00000 degrees
Distortion:            kc_right = [ -0.10139   0.32033   0.01158   0.00256  0.00000 ] ± [ 0.02823   0.20749   0.00192   0.00332  0.00000 ]


Extrinsic parameters (position of right camera wrt left camera):

Rotation vector:             om = [ -0.04989   -0.19885  0.01384 ] ± [ 0.00763   0.01336  0.00160 ]
Translation vector:           T = [ 0.46491   0.02984  0.01657 ] ± [ 0.00051   0.00041  0.00596 ]
```

Figure 12: Stereo calibration parameters after optimization

- Step 3 - the extrinsic parameters is then displayed as seen in figure 13.

- Step 4 - Finally, the the stereo images used for calibration are rectified such that the epipolar lines match with the horizontal scanned lines.

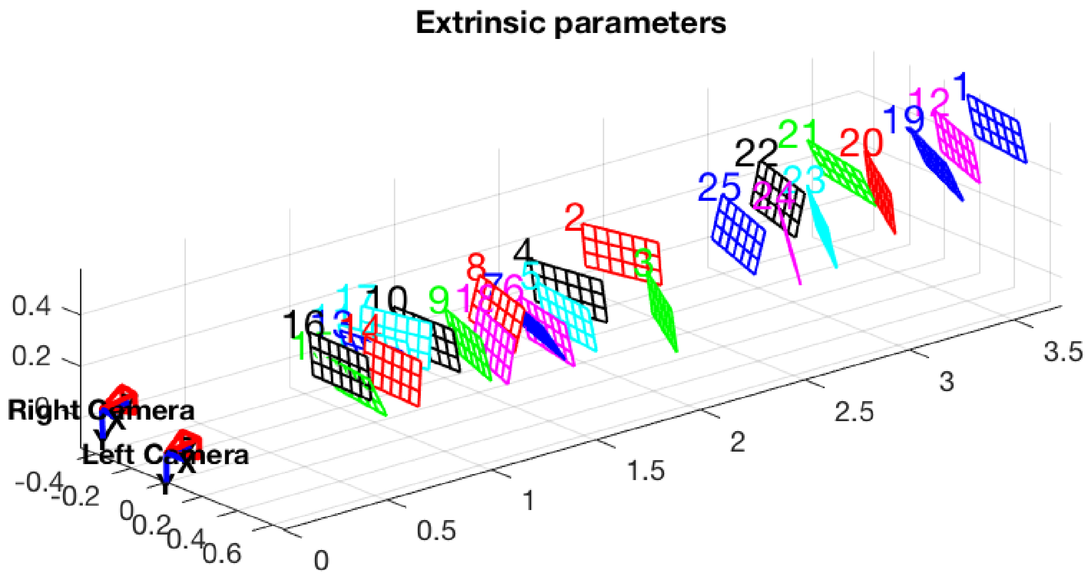


Figure 13: Stereo calibration parameters after optimization

6 Conclusion

Over the course of this lab session we got to apply the theoretical concepts surrounding camera and stereo calibration taught in class onto real world calibration tasks. Having successfully completed the defined objectives, we now have gained a more holistic understanding on this topic and are looking forward to apply this new found knowledge towards realizing the defined objectives for the following lab sessions.