Johns Hopkins University
Computer Aided Medical Procedures
Prof. Nassir Navab, Prof. Ehsan Azimi
Sing Chun Lee, Alejandro Martin-Gomez
Wei-Lun Huang, Yihao Liu

EN.601.454/654
Augmented Reality, Spring 2022
Assignment 2
Due: Feb 16th (Wed) 2022, 8:30am

Assignment 2: Camera Models and Camera Calibration

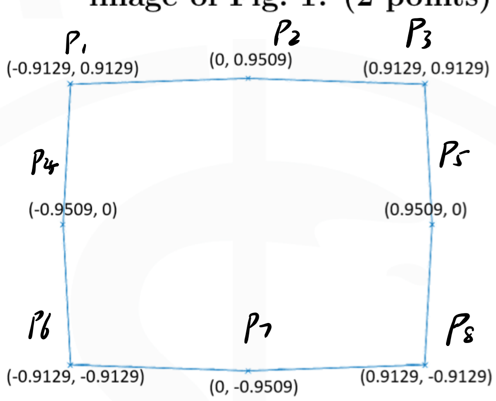
1 (Written) Camera Distortion [5 points + 1 point *(UG Optional)*]

1.1 What is the goal of undistortion? Why should we undistort an image before processing it? (1 point)

Goal: remove the radial & tangential distortion created by lenses.

Reason: Because all image processing procedure is a linear transformation. If we don't undistort the image so that it's in a rectangular view, it will end up being in a mess.

1.2 Given $f(r) = 1 + 0.057r^2 + 0.00014r^4$, compute and sketch the undistorted image of Fig. 1. (2 points)



Mark the points as shown.

$r_1 = \|\vec{P_1}\| = 1.291.$ $\quad$ $r_2 = \|\vec{P_2}\| = 0.951.$

$r_3 = \|\vec{P_3}\| = 1.291.$ $\quad$ $r_4 = \|\vec{P_4}\| = 0.951.$

$r_5 = \| \vec{P_5} \| = 0.951.$

$r_7 = \| \vec{P_7} \| = 0.951.$

$\therefore \; f_1 = f(r_1) = 1.0954.$

$f_3 = f(r_3) = 1.0954.$

$f_5 = f(r_5) = 1.0517.$

$f_7 = f(r_7) = 1.0517.$

$r_6 = \| \vec{P_6} \| = 1.291.$

$r_8 = \| \vec{P_8} \| = 1.291.$

$f_2 = f(r_2) = 1.0517.$

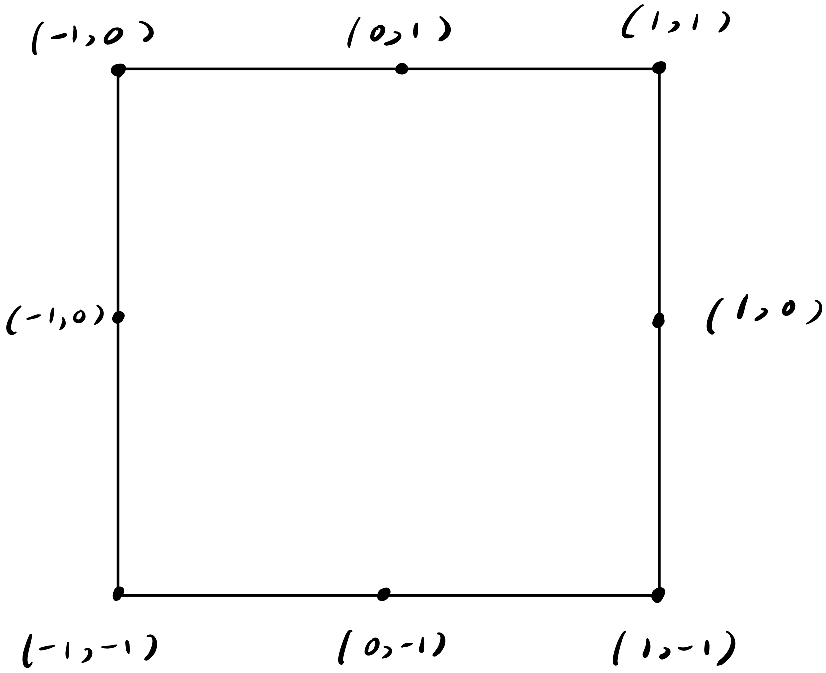
$f_4 = f(r_4) = 1.0517.$

$f_6 = f(r_6) = 1.0954.$

$f_8 = f(r_8) = 1.0954.$

$f_1 \cdot \vec{P_1} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}, \quad f_2 \cdot \vec{P_2} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad f_3 \cdot \vec{P_3} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad f_4 \vec{P_4} = \begin{pmatrix} -1 \\ 0 \end{pmatrix},$

$f_5 \cdot \vec{P_5} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad f_6 \cdot \vec{P_6} = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \quad f_7 \cdot \vec{P_7} = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \quad f_8 \cdot \vec{P_8} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$



1.3 A fisheye camera is a camera that can capture images with a field of view of more than 180^o. (More details on *Wikipedia*). We are also interested in undistorted images from fisheye camera. Will Eq. 1 work for modeling the distortion? Why or why not? (2 points)

It won't work. Because the tangential distortion of a fisheye camera cannot be ignored. Eq. 1 only contains radial distortion.

1.4 *(UG Optional)* Typically, we use $f(r) = 1 + k_1 r^2 + k_2 r^4 + k_3 r^6$ to model the radial distortion. Note that only even power terms are used in the polynomial. Why do we not use the odd order terms? (1 point)

Because the domain for this function is always positive ($r > 0$).

Augmented Reality - Assignment 2 Report

1-IntelRGB

1. Calibration Result

Root Mean Square Error:

0.011045491071220302

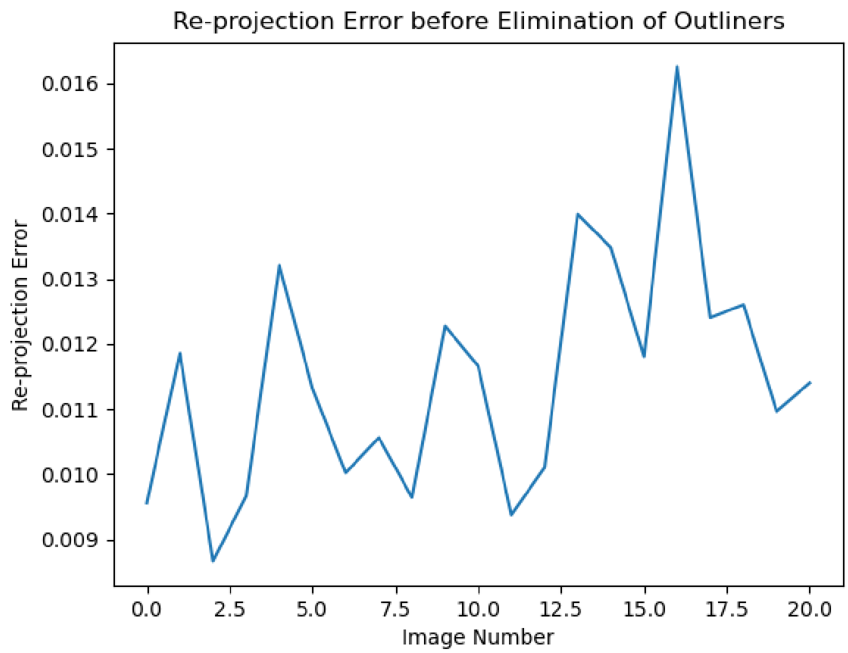
Camera Matrix:

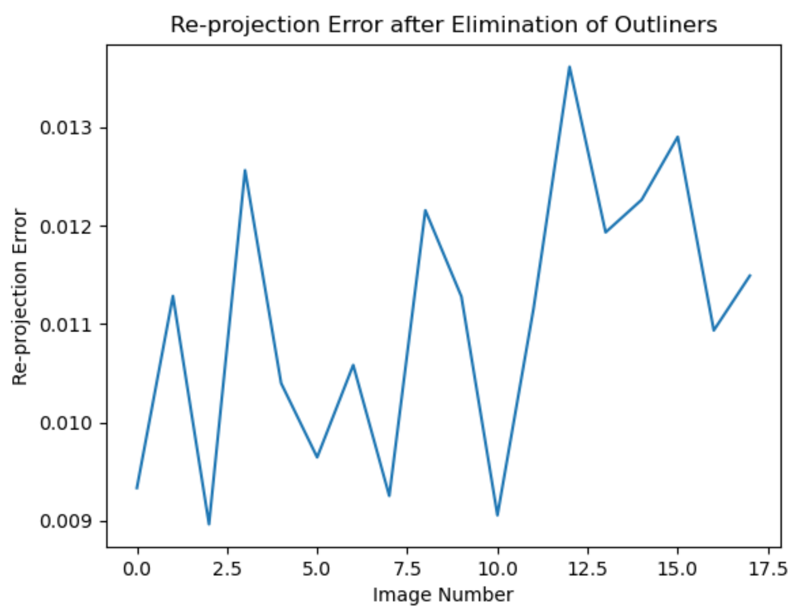
```
[[615.7360059    0.         317.73052346]
 [  0.         616.32042915 249.36521834]
 [  0.           0.           1.        ]]
```

Distortion Parameters:

$[\ 0.13751346\ \ -0.69570955\ \ 0.0043054\ \ -0.00150517\ \ 0.17927672]^T$

2. Reprojection Error





3. Methods to reduce the overall errors / exclude the outliers / poor images

   A. OpenCV has a function *cv.findChessboardCorners()*. If it does not recognize the given chessboard pattern, it will return false (the first return value). This makes sure that **poor images** that cannot be recognized is abandoned.

B. To **exclude the outliners** and **reduce the overall errors**, a calibration is performed first with all the images involved. Then, the reprojection errors are calculated for each image. Then, the root mean error of this calibration is calculated based on those reprojection errors. I get rid of those images which has a relative error greater than 20%, where the relative error is calculated by

$$|rms\text{-}rep| / rms *100\%$$

Then do the calibration one more time to obtain the final version of calibration results.

2-Intel

1. Calibration Result

Root Mean Square Error:

0.16909332507604674

Camera Matrix:

K1

```
[[475.55624522   0.         321.82614246]
 [  0.         475.69168999 235.34915325]
 [  0.           0.           1.        ]]
```

K2

```
[[615.19725189   0.         317.71265004]
 [  0.         615.78529678 249.18332197]
 [  0.           0.           1.        ]]
```

Distortion Parameters:

D1

[-0.14269781 0.13005809 0.00053476 0.00125527 -0.40158586]^T

D2

[0.13902456 -0.70904186 0.00435479 -0.00155607 0.21570048]^T

Other parameters (See outputs for more information):

Q

[[1. 0. 0. -344.81966782]

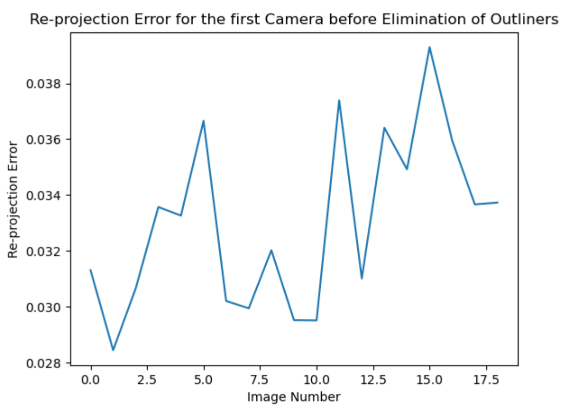
[0. 1. 0. -257.04434395]

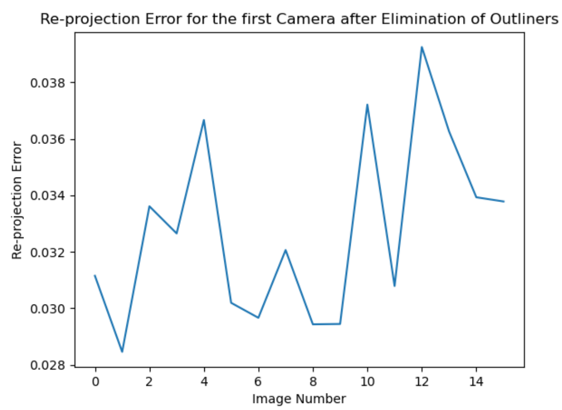
[0. 0. 0. 508.53094563]

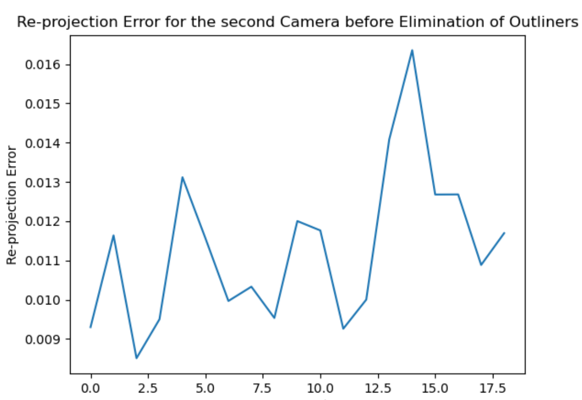
[0. 0. -41.84623047 0.]]

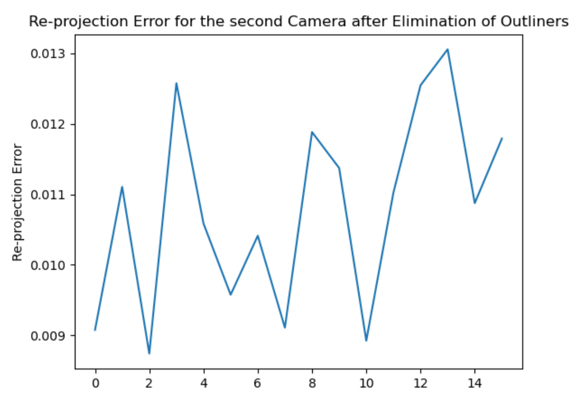
where Q matrix is required to get depth map from disparity map.

2. Reprojection Error









3. Methods to reduce the overall errors / exclude the outliers / poor images

   A. The same as the first dataset.
   B. To **exclude the outliners** and **reduce the overall errors**, a calibration is first performed for both cameras with all the images. Then I get rid of those pairs of images that yields a relative error greater than 20% for either camera. For example, if an image gives a relative error that greater than 20% for Camera 1, then that pair of images will be deleted, even if the corresponding image for Camera 2 does not give a such big relative error. The relative error is again calculated by

      $$|rms-rep| / rms *100\%$$

      Then do the calibrations one more time to obtain the final version of calibration results.