COL780
Assignment 1

Avani Jain
2020MT10792

Contents

1 Part 1 : Camera Calibration

1.1 Method

The method for camera calibration uses a checkerboard pattern. The pictures of the pattern are taken from a laptop screen because a sheet of paper was not getting stuck to the wall completely, hence creating error ofs a few pixels. The pictures are also taken in dark, to reduce errors from the shadows on the laptop screen.

The method used for the calibration is as written in steps :

1. **Detecting corners using Harris Corner Method**. It uses openCV method for detecting Harris Corners followed by NMS method exactly same as done in Assignment 2. On trying to use the code from Assignment 2 for corner detection, it was found that the code didn't gave exact corners. Hence, the openCV method was used. After that, NMS is used to reduce the number of corners, and get one precise pixel as a corner.

2. **Getting vanishing points in two perpendicular directions**. First, an appropriate direction is chosen in the checkerboard image and the task is to find the vanishing point in that direction. This is done by manually finding the equations of lines that are in real world parallel to each other and finding their intersection point. Finding the line equation is done manually by reading the pixel coordinates from step 1 and joining 2 such points. The step is repeated for the direction perpendicular to previously chosen one.

3. **Getting 5 such pairs of vanishing points** by repeating above steps on 5 different images. The optimisation I did at a later stage was to instead just use 3 images, by taking diagonal directions in 2 of the images as well. This could be done because the checkerboard has squares and its diagonals are perpendicular to each other.

4. **Finding the matrix W**. W is symmetric, hence has 5 parameters. So, using these 5 points in the equation, we can get the value of W as $v_2^T W v_1 = 0$. For one pair,

$$\begin{pmatrix} x_1 & y_1 & 1 \end{pmatrix} \begin{pmatrix} w_{11} & w_{12} & w_{13} \\ w_{12} & w_{22} & w_{23} \\ w_{13} & w_{23} & w_1 \end{pmatrix} \begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix} = 0$$

$$\Rightarrow \begin{pmatrix} x_1x_2 & x_1y_2 + x_2y_1 & x_1 + x_2 & y_1y_2 & y_1 + y_2 \end{pmatrix} \begin{pmatrix} w_{11} \\ w_{12} \\ w_{13} \\ w_{22} \\ w_{23} \end{pmatrix} = -1$$

$$\Rightarrow A \begin{pmatrix} w_{11} \\ w_{12} \\ w_{13} \\ w_{22} \\ w_{23} \end{pmatrix} = \begin{pmatrix} -1 \\ -1 \\ -1 \\ -1 \\ -1 \end{pmatrix}$$

   Taking, $\begin{pmatrix} x_1x_2 & x_1y_2 + x_2y_1 & x_1 + x_2 & y_1y_2 & y_1 + y_2 \end{pmatrix} = a_i$ we make a matrix A_{5x5}, with its rows as $\{a_0, a_1, a_2, a_3.a_4\}$ from 5 of the pairs obtained in step 3. After this, we solve this system of equations to get W.

5. **Finding the camera parameters** using cholesky decomposition on W^{-1}. We know that $K^{-T}K^{-1} = W \Rightarrow W^{-1} = KK^T$ and from here getting the camera parameters u_x, u_y, s, f_x, f_y.

1.2 Results

The matrix K, obtained is as shown,

$$K = \begin{pmatrix} 1.74822648 & 0.30040889 & 0.63326311 \\ 0 & 1.64155325 & 0.34782767 \\ 0 & 0 & 1 \end{pmatrix}$$

Also, I scaled the x, y pixels by a factor of 1000. Hence the values of parameters are : $u_x = 633, u_y = 348, f_x = 1.75, f_y = 1.64, s = 0.3$, which is close to what we should have obtained. The camera centre lies almost at the centre of the image, skew is close to zero and the focal lengths in both the directions is almost the same.

2 Part 2 : Insertion of artificial objects

2.1 Method

$$x_{3\times1} = K_{3\times3}[R \;\; t]_{3\times4}X_{4\times1}$$

In this equation, now since we know the matrix K, we want to find the matrices R, t for each image. For this, the method used is that of point correspondences by fixing the left topmost corner of the checkerboard to be the origin in world coordinates and then mapping the coordinates with respective corners in the image.

For solving, the matrix equation to get the parameters, the following rotation matrix construction is used. And two points, will be required to get the 6(3+3) parameters, as each point gives 3 equations and 6 equations are required to determine the parameters completely.

2.2 Results

Refer to this link. Additionally, all the code files as well as the Jupyter notebook, along with the requirements.txt file, have been included in the main folder.

$$a^2 + b^2 + c^2 + d^2 = 1.$$

When the rotation is applied, a point at position $\overrightarrow{x}$ rotates to its new position

$$\vec{x}' = \begin{pmatrix} a^2 + b^2 - c^2 - d^2 & 2(bc - ad) & 2(bd + ac) \\ 2(bc + ad) & a^2 + c^2 - b^2 - d^2 & 2(cd - ab) \\ 2(bd - ac) & 2(cd + ab) & a^2 + d^2 - b^2 - c^2 \end{pmatrix} \vec{x}.$$

Figure 1: Rotation matrix

3 Links of Generated Outputs

Click on this link for the outputs

There are two folders for each dataset - **output**, **final**. The number following them in the name is for the respective dataset as given in the input.
output folder contains the initially detected harris corners after doing part 1 of the assignment, the corner pixels are surrounded by green rectangles with their location written in the image.
final folder contains the files obtained after inserting the artificial object.

All the code files as well as the Jupyter notebook, along with the requirements.txt file, have been included in the main folder.