

# 作业二 相机校正

——518021910971 裴奕博

## 实验原理

- 根据上课的推导，我们得到在相机成像时，物点坐标（三维）、像点坐标（二维）和相机内参的关系为

$$P' = \begin{bmatrix} x' \\ y' \\ z \end{bmatrix} = \begin{bmatrix} \alpha & 0 & c_x & 0 \\ 0 & \beta & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & 0 & c_x & 0 \\ 0 & \beta & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} P = K [I \ 0] P \quad (1)$$

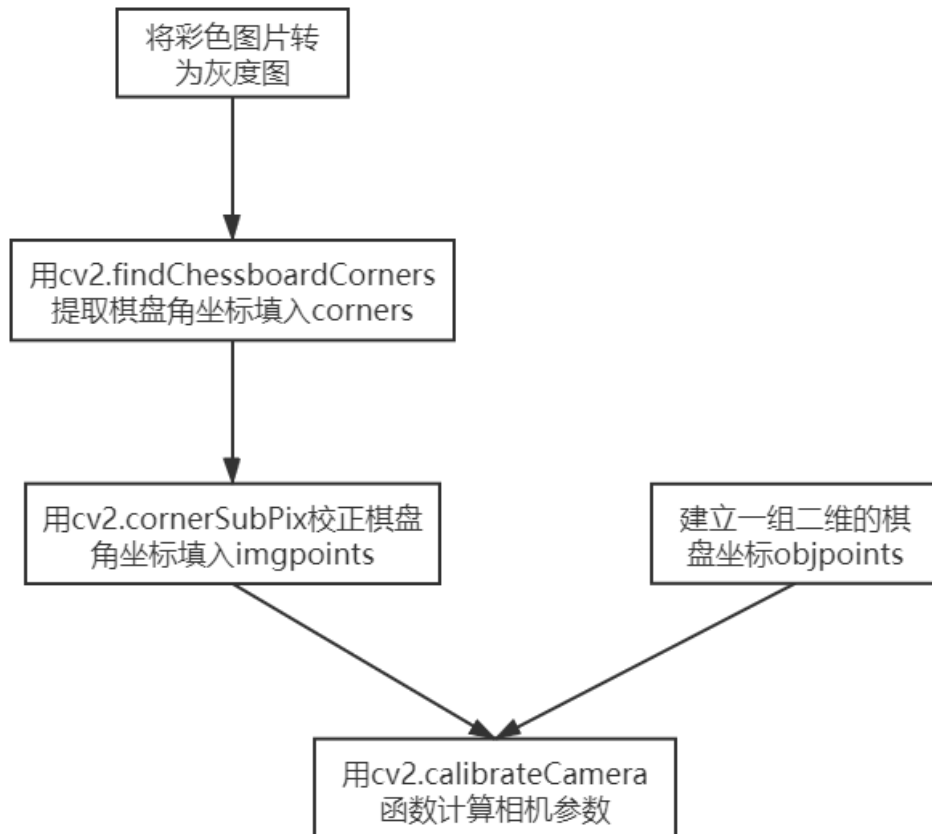
- 其中， $P'$ 为像点坐标， $P$ 为物点坐标， $K$ 为相机的内参矩阵。
- 因此，我们可以通过取足够多组的 $P$ 和 $P'$ 来求出 $K$ 中的 $\alpha, \beta, c_x, c_y$ 四个参数。

## OpenCV-Python代码实现

```
1 import numpy as np
2 import cv2
3 import glob
4 import matplotlib.pyplot as plt
5
6
7 # termination criteria
8 criteria = (cv2.TERM_CRITERIA_EPS + cv2.TERM_CRITERIA_MAX_ITER, 30, 0.001)
9 # prepare object points, like (0,0,0), (1,0,0), (2,0,0) ....,(6,5,0)
10 objp = np.zeros((7*10,3), np.float32)
11 objp[:, :2] = np.mgrid[:7,:10].T.reshape(-1,2)
12
13 # Arrays to store object points and image points from all the images.
14 objpoints = [] # 3d point in real world space
15 imgpoints = [] # 2d points in image plane.
16
17 images = glob.glob('data/*.jpg')
18
19 for fname in images:
20     img = cv2.imread(fname)
21     gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
22     #plt.imshow(gray, cmap='gray')
23     # Find the chess board corners
24     ret, corners = cv2.findChessboardCorners(gray, (7,10), None)
25
26     # If found, add object points, image points (after refining them)
27     if ret == True:
28         objpoints.append(objp)
29         corners2 = cv2.cornerSubPix(gray, corners, (11,11), (-1,-1), criteria)
30         imgpoints.append(corners2)
31
32     # Draw and display the corners
33     img = cv2.drawChessboardCorners(img, (7,10), corners2, ret)
34     cv2.imshow('img', img)
35     cv2.waitKey(500)
```

```
36 cv2.destroyAllWindows()
37
38 ret, mtx, dist, rvecs, tvecs = cv2.calibrateCamera(objpoints, imgpoints,
39 gray.shape[::-1],None,None)
40 printa(mtx)
41 # Code Reference:https://opencv-python-
42 tutorials.readthedocs.io/en/latest/py_tutorials/py_calib3d/py_calibration/py_
43 calibration.html
```

## 工作流程



## 计算结果

```
MI
ret, mtx, dist, rvecs, tvecs = cv2.calibrateCamera(objpoints, imgpoints,
gray.shape[::-1],None,None)
mtx

array([[458.87333421,  0., 177.76747824],
       [ 0., 473.80246366, 155.26872519],
       [ 0.,  0., 1.]])
```