

# 鱼眼镜头标定基本原理、实现及测试报告

齐勇

2019 年 9 月 3 日更新

## 前言

在 VSLAM 中，经常会使用鱼眼镜头或者广角镜头。本文主要分为理论部分与鱼眼镜头标定实践两部分，其中理论部分，主要参考《A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses》，作者为 Juho Kannala 和 Sami S. Brandt，写于 2006 年，同时这篇文章的作者在 2004 年也写了一篇与鱼眼镜头标定相关的论文《A Generic Camera Calibration Method for Fish-Eye Lenses》，同样值得参考。

## 一 理论部分

对于大多数具有窄角甚至广角镜头的传统相机，针孔相机模型伴随着镜头畸变是一个比较不错的近似。但是针孔模型不太适用于鱼眼镜头。鱼眼镜头一般被设计用来覆盖相机前方的整个半球形视场，视角非常大，约  $180^\circ$ 。然而，不可能通过透视投影在有限的图像平面上投射半球形视场。因而，鱼眼镜头服从某些其他投影模型。这就是为什么鱼眼镜头本身固有的畸变不应该仅被认为是与针孔模型的偏差。

很多作者对鱼眼镜头的适合的可能模型进行了研究。许多方法的思想是将原始鱼眼镜头成像为遵循针孔模型，畸变参数是通过变换后强制按直线来估计的，但问题是这些方法并不能完

全校准。最近，鱼眼镜头相机的第一种自动校准方法也出现了，Claus 和 Fitzgibbon[1] 提出了一种畸变模型，它同样允许相机运动和镜头几何的同时线性估计，而 Thirthala 和 Pollefeys[2]使用径向一维摄像机的多视图几何来估计非参数相机模型。此外，Barreto 和 Daniilidis 最近的工作引入了径向基本矩阵来纠正广角镜头的失真[3]。然而，这些方法的重点更多的是在于自动校准技术，而不是真实镜头的精确建模。

由于透视投影模型不适用于鱼眼镜头，我们使用更灵活的径向对称投影模型。

① 针孔相机的透视投影：

$$r = f \tan q \quad (\text{i. 透视投影}) \quad (1)$$

其中， $q$ 是主轴与入射光线之间的角度， $r$ 是图像点与主点之间的距离， $f$ 是焦距。鱼眼镜头通常设计为遵循以下投影之一：

$$\textcircled{2} \quad r = 2f \tan(q/2) \quad (\text{ii. 立体投影}) \quad (2)$$

$$\textcircled{3} \quad r = fq \quad (\text{iii. 等距投影}) \quad (3)$$

$$\textcircled{4} \quad r = 2f \sin(q/2) \quad (\text{iv. 等角投影}) \quad (4)$$

$$\textcircled{5} \quad r = f \sin(q) \quad (\text{v. 正交投影}) \quad (5)$$

也许最常见的模型是等距离投影。

不同的投影行为如图 1 (a) 所示，针孔相机与鱼眼相机之间的差异如图 1 (b) 所示。

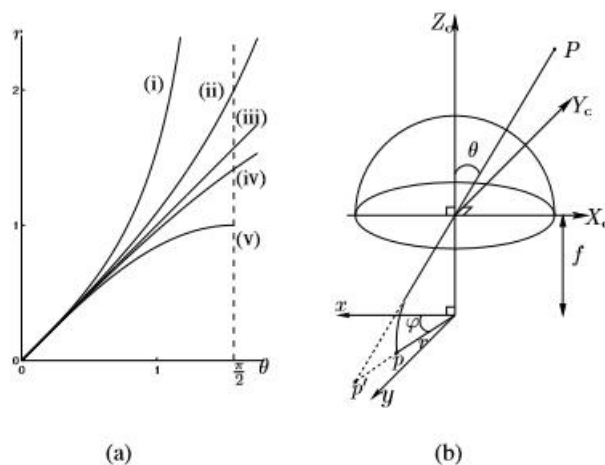


Fig. 1. (a) Projections (1)-(5) with  $f = 1$ . (b) Fish-eye camera model.  
The image of the point  $P$  is  $p$  whereas it would be  $p'$  by a pinhole camera.

然而，真实镜头并不完全遵循设计的投影模型。从自动校准的角度来看，如果我们只有一个适合不同类型镜头的型号，那么它也会很有用。因此，我们考虑一般形式的估计。

$$r(q) = k_1 q + k_2 q^3 + k_3 q^5 + k_4 q^7 + k_5 q^9 + \dots \quad (6)$$

不失一般性，甚至权重都被取消了。这是因为我们可以将  $r$  作为奇函数扩展到负数一侧，而奇数幂跨越连续奇函数集。对于计算，我们需要确定 (6) 中的各项。

为了使得本文不那么枯燥，关于鱼镜头标定的理论部分我们仅介绍到这里，如果感兴趣的读者，可以去仔细研读论文《A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses》。

## 二 实践部分

在上述论文作者的主页，作者提供的 Matlab 标定代码：[Camera Calibration Toolbox for Generic Lenses](http://www.ee.oulu.fi/~jkannala/calibration/calibration.html)：<http://www.ee.oulu.fi/~jkannala/calibration/calibration.html>。

此处，笔者简单为大家介绍一下 OpenCV 中关于鱼眼标定的流程。

首先，可以参考 OpenCV Documentation 中提供的鱼眼镜头标定相关函数：

[https://docs.opencv.org/3.2.0/db/d58/group\\_\\_calib3d\\_\\_fisheye.html](https://docs.opencv.org/3.2.0/db/d58/group__calib3d__fisheye.html)

具体鱼眼镜头标定步骤如下：

本次实验，使用小觅相机（标准版）进行实验，小觅相机的镜头视角较大，大约在  $140^{\circ}$  左

右。小觅相机实物如下图 2.1 所示。



图 2.1 小觅相机（标准版）实物图

Step-1: 制作标定板。可以是圆，也可以是棋盘格，当然也可以是圆环。此处我们选择  $4 \times 11$  的 OpenCV 提供的圆形标定板。为了使得标定结果趋于收敛，在工作距离附近，在不同姿态下，采集近 20~30 张标定图片，如下图 2.3 所示。



图 2.3 标定板采集样图

Step-2: 进行形态学处理。此处是由于在采集图片过程中, 难免会在图片中引入环境光带来的噪声。

Step-3: 检测每一张图片中的圆心角点坐标,并排序。( OpenCV 中 findCirclesGrid 函数同时解决了这个问题 )。

Step-4: 进行标定。( cv::fisheye:calibrate )

Step-5: 计算重投影误差。在 Step-3 中, cv::fisheye::calibrate 返回值为总重投影误差, 当然也可以进一步计算 x 和 y 方向的重投影误差值。

Step-6: ( 此步非必须 ) 进行去畸变, 进一步验证标定参数的正确性。

组合使用函数:

- 1) `fisheye::estimateNewCameraMatrixForUndistortRectify`
- 2) `cv::fisheye::initUndistortRectifyMap`

### 3) remap

即可获得畸变矫正后的效果图。

经过鱼眼标定后，使用标定好的参数进行去畸变后效果图（使用函数 `void estimateNewCameraMatrixForUndistortRectify(InputArray K, InputArray D, const Size &image_size, InputArray R, OutputArray P, double balance = 0.0, const Size& new_size = Size(), double fov_scale = 1.0` 进行估计新的相机矩阵用于畸变矫正或者校正时，图 2.4 为参数 *balance* 设为 0 时的畸变矫正效果图，图 5 为参数 *balance* 设为 1 时的效果图）。



图 2.4 鱼眼模型标定后，畸变矫正，balance=0 时

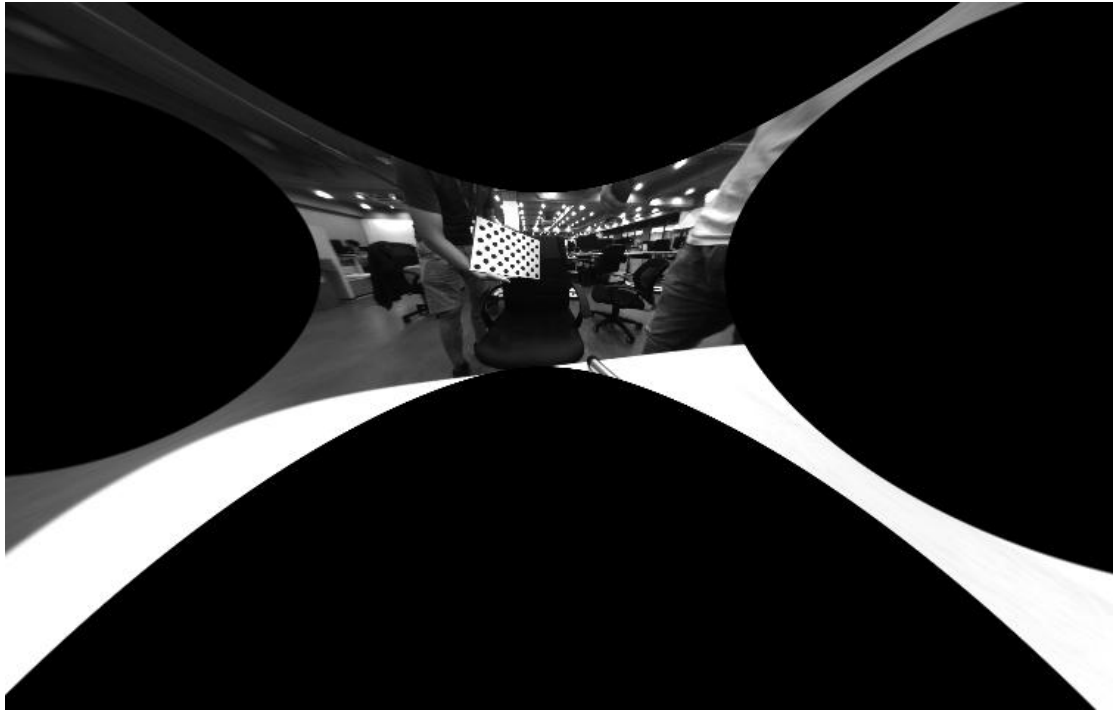


图 2.5 鱼眼模型标定后，畸变矫正效果图， $\text{balance}=1.0$  时

如果选用**针孔模型**来进行标定，去畸变后的效果图如下所示。

使用针孔模型进行标定后，同样对图 2.3 进行去畸变效果图如下（图 2.6 为  $\text{balance}=0$ ，图 2.7 为  $\text{balance}=1$  时）：



图 2.6 针孔模型标定后，畸变矫正效果图， $\text{balance}=0$  时



图 2.7 针孔模型标定后，畸变矫正效果图，balance=1 时

由图可见，使用针孔模型去畸变后，整个画幅的图像的畸变不符合「枕形」或者「桶形」，因而使用针孔模型来进行标定鱼眼镜头是不太适合的。

### 三 鱼眼标定测试报告

首先根据小觅官网提供的方法，获得本相机出厂时的标定参数如下：

```
width: 752
height: 480
in_left:
  # coeffs: k2,k3,k4,k5,mu,mv,u0,v0
  coeffs: [-0.02724966413904360, 0.01387972934505288,
    -0.01443299094096242, 0.00220385605939798,
    366.68518269136535537, 366.83887460436199035,
    388.43206635118957593, 228.62321647306046657]
in_right:
  # coeffs: k2,k3,k4,k5,mu,mv,u0,v0
```



```

        coeffs: [ -0.02325249789634150, -0.00016835064552267,
                  -0.00253078389852647, -0.00027373996972984,
                  367.03822091108315817, 367.09928208615383483,
                  381.71902127274807981, 245.51155028218011012]
    ex_right_to_left:
        rotation:      [0.99998208152544887,      0.00512209901991103,
                        -0.00309850442323602,
                        -0.00513330714531291,      0.99998027147177970,
                        -0.00362019681475175,
                        0.00307990028774734,      0.00363603752124284,
                        0.99998864665823162]

        translation:   [-120.19298798324432198,    0.35272130294376047,
                        -0.43667870870355763]

```

同时，官网对于相机的标定精度要求给出了说明：  
[https://mynt-eye-s-sdk.readthedocs.io/en/latest/src/tools/calibration\\_tool.html#calibration-result](https://mynt-eye-s-sdk.readthedocs.io/en/latest/src/tools/calibration_tool.html#calibration-result)，部分截图如下：

### 3.4 Calibration result

Calibration result, It is desirable to have a reprojection error of 0.2 or less. If exceeds 1, it needs to be recalibrated.

Reprojection error, visible output after calibration completion "Final reprojection error: 0.201

Pixels", or see the calibration result file "stereo\_reprojection\_error.yaml".

## 3.1 实验一：4\*11——38.0（圆点个数为4\*11，圆心间距为38.0）

### 3.1.1 第一种方式：使用鱼眼模型进行标定

我们采集了第一组数据，如上述的实践部分，经过标定后，得到的标定结果如下：

```

cam_matrix_L: !!opencv-matrix
  rows: 3
  cols: 3
  dt: d
  data: [ 3.6692294481315821e+02, 0., 3.8849038986104165e+02, 0.,
          3.6698541196248220e+02, 2.2921988923934205e+02, 0., 0., 1. ]

```

```
cam_distortion_L: !!opencv-matrix
  rows: 4
  cols: 1
  dt: d
  data: [ -1.4475125669490854e-02, -3.8647566415345037e-02,
          5.4622331944428684e-02, -2.9602408361161195e-02 ]
totalReproError_L: 1.0556470195533366e-01
ReproError_X_L: 7.8916448153808397e-02
ReproError_Y_L: 7.0114909325375924e-02
cam_matrix_R: !!opencv-matrix
  rows: 3
  cols: 3
  dt: d
  data: [ 3.6732489384818217e+02, 0., 3.8238460185899771e+02, 0.,
          3.6741049010963320e+02, 2.4556292675533936e+02, 0., 0., 1. ]
cam_distortion_R: !!opencv-matrix
  rows: 4
  cols: 1
  dt: d
  data: [ -1.9450444212558472e-02, -1.2395283958501700e-02,
          7.8144639511264317e-03, -2.6207619181709699e-03 ]
totalReproError_R: 5.2048633009384830e-02
ReproError_X_R: 3.8473798543000523e-02
ReproError_Y_R: 3.5054629135368581e-02
system_rotation: !!opencv-matrix
  rows: 3
  cols: 3
  dt: d
  data: [ 9.9997606864553179e-01, 4.9360653196030219e-03,
          -4.8474112046838361e-03, -4.9414964848058024e-03,
          9.9998717568487516e-01, -1.1090889402455438e-03,
          4.8418745045007072e-03, 1.1330158636733230e-03,
          9.9998763618673570e-01 ]
system_translation: !!opencv-matrix
  rows: 3
  cols: 1
  dt: d
  data: [ -1.2021952827900481e+02, 5.1530796618836971e-01,
          -7.8344189782570273e-01 ]
stereo_calib_rms: 8.8466898718544767e-02
stereo_length_base: 1.2022318538091444e+02
```

### 3.1.2 第二种方式：使用针孔模型进行标定

使用上述图片，根据针孔模型进行标定，标定结果如下：

```
cam_matrix_L: !!opencv-matrix
  rows: 3
  cols: 3
  dt: d
  data: [ 3.6643548419126029e+02, 0., 3.8752360144936682e+02, 0.,
          3.6642898198342800e+02, 2.2915860550352880e+02, 0., 0., 1. ]
cam_distortion_L: !!opencv-matrix
  rows: 1
  cols: 5
  dt: d
  data: [ -3.3974321781692618e-01, 1.5270022644803255e-01,
          1.8372363110327639e-07, 2.8854672123319246e-04,
          -3.8184526268111633e-02 ]
totalReproError_L: 1.0685563958318985e-01
ReproError_X_L: 8.0143088759332029e-02
ReproError_Y_L: 7.0676821057673062e-02
cam_matrix_R: !!opencv-matrix
  rows: 3
  cols: 3
  dt: d
  data: [ 3.6623919385122952e+02, 0., 3.7714676361702709e+02, 0.,
          3.6689707995757766e+02, 2.4778166400322718e+02, 0., 0., 1. ]
cam_distortion_R: !!opencv-matrix
  rows: 1
  cols: 5
  dt: d
  data: [ -3.1667847642985580e-01, 1.1209063671770755e-01,
          -3.0445478743536047e-04, 3.6581310307071005e-04,
          -1.8928010475205509e-02 ]
totalReproError_R: 1.3074374591502053e-01
ReproError_X_R: 1.0985090576068189e-01
ReproError_Y_R: 7.0899263743269769e-02
system_rotation: !!opencv-matrix
  rows: 3
  cols: 3
  dt: d
  data: [ 9.9996714474672976e-01, 4.3623655460442315e-03,
          6.8322173498358094e-03, -4.3180796346986008e-03,
          9.9996965993648490e-01, -6.4833166496026332e-03,
```

```
-6.8602926571038093e-03, 6.4536015799938964e-03,  
9.9995564272186865e-01 ]  
system_translation: !!opencv-matrix  
  rows: 3  
  cols: 1  
  dt: d  
  data: [ -1.2010496161125528e+02, -1.5737946492064803e-01,  
          2.7008414730993269e+00 ]  
stereo_calib_rms: 1.5137056292327217e-01  
stereo_length_base: 1.2013542823247394e+02
```

## 3.2 实验二：4\*11——18.0 (圆点个数为 4\*11，圆心间距为 18.0)

### 3.2.1 第一种方式：使用鱼眼模型进行标定

在不同位姿下，采集了 25 张标定图片，进行标定。采集的图片示例如图 3.1 所示：



图 3.1 采集的图片示例

标定结果如下：

```
cam_matrix_L: !!opencv-matrix
```

```
rows: 3
cols: 3
dt: d
data: [ 3.6567330021287199e+02, 0., 3.8880351117607796e+02, 0.,
        3.6552032298453292e+02, 2.2887630967267802e+02, 0., 0., 1. ]
cam_distortion_L: !!opencv-matrix
rows: 4
cols: 1
dt: d
data: [ -2.2265342570633471e-02, 1.5851292446402130e-03,
        -1.9926975550950743e-02, 1.9182817729218298e-02 ]
totalReproError_L: 7.8606914913035578e-02
ReproError_X_L: 5.5440342825332722e-02
ReproError_Y_L: 5.5726254670081590e-02
cam_matrix_R: !!opencv-matrix
rows: 3
cols: 3
dt: d
data: [ 3.6523414060339178e+02, 0., 3.8287768871620068e+02, 0.,
        3.6519779829048923e+02, 2.4647924026150437e+02, 0., 0., 1. ]
cam_distortion_R: !!opencv-matrix
rows: 4
cols: 1
dt: d
data: [ -1.6778657974834663e-02, -2.3774698151175864e-02,
        3.0291785655232863e-02, -1.5270844122200151e-02 ]
totalReproError_R: 3.5763630698334291e-02
ReproError_X_R: 2.6604685362486669e-02
ReproError_Y_R: 2.3900376513559942e-02
system_rotation: !!opencv-matrix
rows: 3
cols: 3
dt: d
data: [ 9.9997346538498399e-01, 4.5526652033247338e-03,
        -5.6869821076299606e-03, -4.5737982859423666e-03,
        9.9998266323913698e-01, -3.7085833416907730e-03,
        5.6699995754477643e-03, 3.7344960448756387e-03,
        9.9997695205644876e-01 ]
system_translation: !!opencv-matrix
rows: 3
cols: 1
dt: d
data: [ -1.2030252514122655e+02, 4.2350907393087243e-02,
        -2.1771375057183278e+00 ]
```

stereo\_calib\_rms: 4.4250259757255647e-02  
stereo\_length\_base: 1.2032223101602467e+02

使用上述标定结果，去畸变效果图如图 3.2 所示和图 3.3 所示：



图 3.2 去畸变效果，balance=0 时



图 3.3 去畸变效果，balance=1.0 时

### 3.2.2 第二种方式：使用针孔模型进行标定

标定结果如下：

```
cam_matrix_L: !!opencv-matrix
  rows: 3
  cols: 3
  dt: d
  data: [ 3.6520530205553467e+02, 0., 3.8850067590329195e+02, 0.,
          3.6521381587435377e+02, 2.2883745067313109e+02, 0., 0., 1. ]
cam_distortion_L: !!opencv-matrix
  rows: 1
  cols: 5
  dt: d
  data: [ -3.4034151483958469e-01, 1.5380877430054415e-01,
          4.9988209073042756e-05, 2.0996153974837416e-04,
          -3.8985624839036316e-02 ]
totalReproError_L: 7.9941732487941716e-02
ReproError_X_L: 5.6713694379392963e-02
ReproError_Y_L: 5.6340371519717346e-02
cam_matrix_R: !!opencv-matrix
  rows: 3
  cols: 3
  dt: d
  data: [ 3.6638607087768571e+02, 0., 3.6989823291604279e+02, 0.,
          3.6774392507207733e+02, 2.4908416240919391e+02, 0., 0., 1. ]
cam_distortion_R: !!opencv-matrix
  rows: 1
  cols: 5
  dt: d
  data: [ -3.1840829141662680e-01, 1.1973930282670203e-01,
          -1.4200810323298889e-04, 2.1296607241371112e-03,
          -2.2442069279915523e-02 ]
totalReproError_R: 5.9297023176395236e-02
ReproError_X_R: 4.7255473023464022e-02
ReproError_Y_R: 3.5819788201470339e-02
system_rotation: !!opencv-matrix
  rows: 3
  cols: 3
  dt: d
  data: [ 9.9970921301278404e-01, 2.5529104808288904e-03,
          2.3978575133581426e-02, -2.3254625927615524e-03,
```

```
9.9995208858035589e-01, -9.5085628521734272e-03,  
-2.4001700795768667e-02, 9.4500366063280279e-03,  
9.9966725222298314e-01 ]  
system_translation: !!opencv-matrix  
  rows: 3  
  cols: 1  
  dt: d  
  data: [ -1.1742476811044021e+02, -8.4715089418540657e-01,  
          6.8691968452346206e+00 ]  
stereo_calib_rms: 1.5809631318076781e-01  
stereo_length_base: 1.1762856666527378e+02
```

使用上述的标定结果，进行去畸变，效果如下：

原图：



图 3.4 原图

去畸变效果图如下：





图 3.5 去畸变效果,  $\text{balance}=0$  时



图 3.6 去畸变效果,  $\text{balance}=1$  时

### 3.3 结论

根据以上实验结果，我们可以得出如下结论：

- 1、通过鱼眼模型与针孔模型去畸变的效果来看，小觅相机更符合鱼眼模型。
- 2、通过两组实验的重投影误差来看，使用本程序的标定精度在 0.1 像素左右，满足小觅相机的使用要求。
- 3、根据立体标定结果来看，X 方向的精度较高，Y 方向次之，Z 方向最差。

## 四 使用标定初值进行鱼眼镜头标定测试报告

由于在扫地机量产过程中，预留给相机标定的时间有限，因而，如果能够做到一次采图即可标定出相机的内外参，那将是最佳的选择了。因而，笔者做了以下的测试。

### 4.1 实验一：4\*11——42.0（圆点个数为 4\*11，圆心间距为 42.0），不使用标定初值

在做本次实验时，鉴于已经加工回来一块较大的标定板 **42-4\*11**（圆心距为 42，圆斑的行列数为 4\*11），圆在相机中的成像质量也更高（此处的成像质量以边缘梯度是否锐利作为判断标准），以下实验，将采用本标定板进行，仍然使用实验三中的同款小觅相机进行标定实验，采集了两组图片，每一组近 25 张（摆放在相机视野的不同位置，尽量覆盖到全部的相机视野）。

首先，使用第一组图片不使用标定初值，进行鱼眼镜头的标定。采集的左图（4.1）原图如图 4.1 所示。

标定结果如下：

```
serial_num_L: "123456L"
cam_matrix_L: !!opencv-matrix
  rows: 3
  cols: 3
  dt: d
  data: [ 3.6605340439727684e+02, 0., 3.8880545714475403e+02, 0.,
          3.6614293821575342e+02, 2.2970318138125370e+02, 0., 0., 1. ]
cam_distortion_L: !!opencv-matrix
  rows: 4
  cols: 1
  dt: d
  data: [ -2.4387893646601624e-02, 5.9618192152208916e-03,
          -1.1177157639323169e-02, 3.5995551933616214e-03 ]
totalReproError_L: 1.1134756038998461e-01
ReproError_X_L: 1.0299857064189263e-01
ReproError_Y_L: 4.2303352710256009e-02
serial_num_R: "123456R"
cam_matrix_R: !!opencv-matrix
  rows: 3
  cols: 3
  dt: d
  data: [ 3.6588732564170311e+02, 0., 3.8199544626644399e+02, 0.,
          3.6593550703353532e+02, 2.4625965187143674e+02, 0., 0., 1. ]
cam_distortion_R: !!opencv-matrix
  rows: 4
  cols: 1
  dt: d
  data: [ -2.1399318131873655e-02, -4.4370359976941718e-03,
          -1.0889731948937063e-04, -3.9400566846670365e-04 ]
totalReproError_R: 5.8506352660408510e-02
ReproError_X_R: 4.0044188565754688e-02
ReproError_Y_R: 4.2655084852035778e-02
right_to_left: " "
system_rotation: !!opencv-matrix
  rows: 3
  cols: 3
  dt: d
  data: [ 9.9997665877095143e-01, 4.8124051951933646e-03,
          -4.8500174764150869e-03, -4.8186417935020568e-03,
          9.9998757742035360e-01, -1.2750279362478125e-03,
          4.8438212756222896e-03, 1.2983686724397600e-03,
          9.9998742573806410e-01 ]
system_translation: !!opencv-matrix
  rows: 3
```

```
cols: 1
dt: d
data: [ -1.1950737013006474e+02, 6.1227786050099997e-01,
        -1.8049372358435560e+00 ]
stereo_calib_rms: 9.9083158572562974e-02
stereo_length_base: 1.1952256773516910e+02
```

去畸变后的效果图如图 4.2 和图 4.3 所示。



图 4.1 采集的原图



图 4.2 经过畸变矫正后， $\text{balance}=0$  时



图 4.3 经过畸变矫正后， $\text{balance}=1$  时

由标定数据（重投影误差在 0.1 左右），以及标定效果图来看，相机标定的效果好。

## 4.2 实验二：4\*11——42.0（圆点个数为 4\*11，圆心间距为 42.0），使用标定初值

在 4.1 实验中，我们已经得到一组较为理想的标定初值，我们使用 4.1 得到的标定结果作为初值，进行对采集的另一组图片（从中选择 9 组图片，尽量保证覆盖整个视野，与产线上保持一致）进行标定（实验中，仅对单目相机的内参和畸变参数赋初值，对于双目标定，不予赋初值）。

标定结果如下：

```
%YAML:1.0
---
serial_num_L: "123456L"
cam_matrix_L: !!opencv-matrix
  rows: 3
  cols: 3
  dt: d
  data: [ 3.6363490529763499e+02, 0., 3.8928700900396507e+02, 0.,
    3.6367576128727632e+02, 2.2887945821936941e+02, 0., 0., 1. ]
cam_distortion_L: !!opencv-matrix
  rows: 4
  cols: 1
  dt: d
  data: [ -2.1187193781209035e-02, -2.6783557581200107e-03,
    -1.6166404472163082e-03, 9.1909358543395506e-04 ]
totalReproError_L: 1.3357717357842309e+00
ReproError_X_L: 6.9695181383929494e-01
ReproError_Y_L: 4.6435815625864418e-01
serial_num_R: "123456R"
cam_matrix_R: !!opencv-matrix
  rows: 3
  cols: 3
  dt: d
  data: [ 3.6390252545168732e+02, 0., 3.8269912746172702e+02, 0.,
    3.6397573754390680e+02, 2.4518582323903436e+02, 0., 0., 1. ]
cam_distortion_R: !!opencv-matrix
  rows: 4
  cols: 1
  dt: d
  data: [ -3.0617779690402358e-02, 4.4597613630298176e-02,
```

```
-9.4046304155131669e-02, 5.9879492795663285e-02 ]
totalReproError_R: 1.6803551029427615e+00
ReproError_X_R: 6.2637193488680987e-01
ReproError_Y_R: 3.8396584089269109e-01
right_to_left: " "
system_rotation: !!opencv-matrix
  rows: 3
  cols: 3
  dt: d
  data: [ 9.9997285085757337e-01, 5.0382282726456652e-03,
    -5.3771557211966725e-03, -5.0410479631442986e-03,
    9.9998716329591841e-01, -5.1095872193205813e-04,
    5.3745123695608842e-03, 5.3805134973680668e-04,
    9.9998541245236894e-01 ]
system_translation: !!opencv-matrix
  rows: 3
  cols: 1
  dt: d
  data: [ -1.1957967466192325e+02, 4.5956228897438084e-01,
    -1.4570683539046720e+00 ]
stereo_calib_rms: 8.8595721862413393e-02
stereo_length_base: 1.1958943447452540e+02
```

采集的原图样图如图 4.4 所示，畸变矫正后的图片如图 4.5（balance=0）和图 4.6(balance=1.0)所示。



图 4.4 采集的原图



图 4.5 去畸变后(balance=0)



图 4.6 去畸变后 (balance=1)

由上述的标定结果可知，重投影误差 ( $>1$ ) 较之实验一的结果，反而变得更大了，这也说明精度更低了。与此组类似的实验，做了多次，结果与此类似，此处便不再展示。



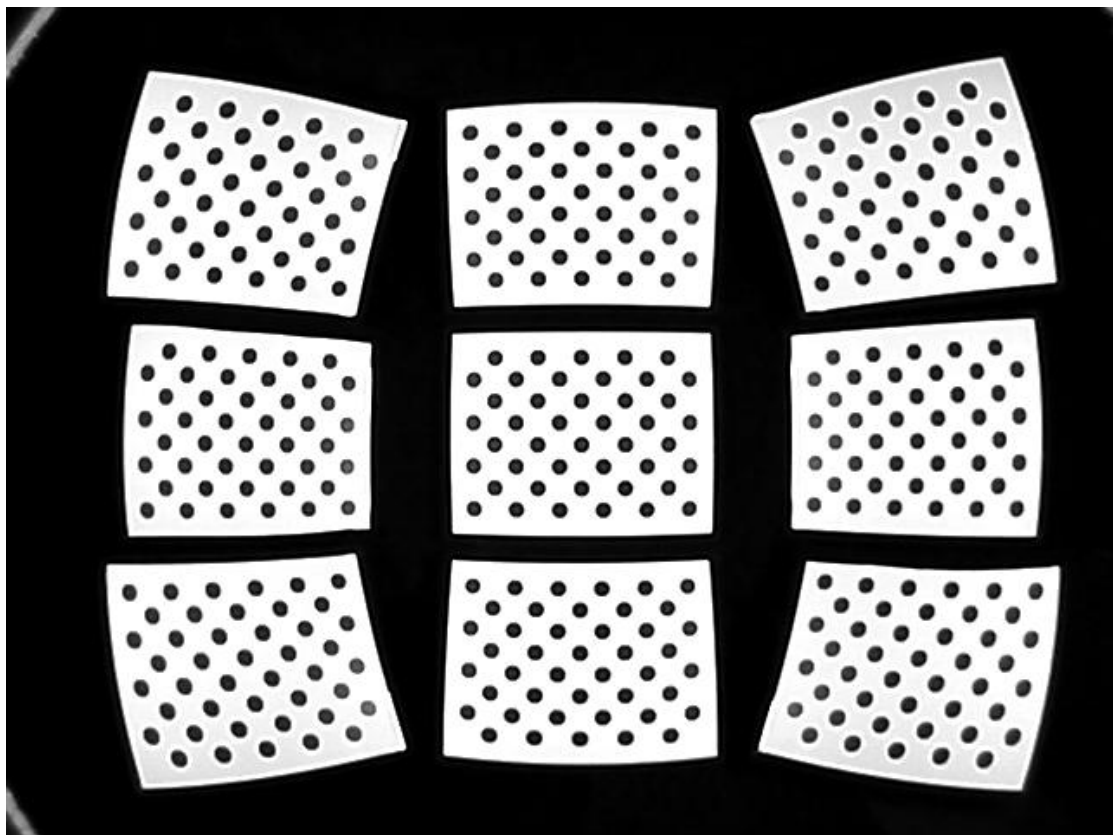
### 4.3 可能原因分析

- 1) 由于鱼眼镜头的视野较大（一般为 $>140^\circ$ ），仅使用 9 组图片而想覆盖整个相机视野有点困难，导致标定结果不能收敛。
- 2) 本身 9 组图片的角点提取难以保证精度较高（图像分辨率为  $752 \times 480$ ）。

## 四 跋

简单写下我自己对于扫地机标定流程的一些思考。

鉴于以上的实验表明，在采集图片 $>20$ 组时，标定结果（从重投影和去畸变效果）综合分析，标定结果已经较好，对于落实到扫地机上进行使用问题不大。但是，这需要多次采图（动相机或者动标定板工装）（如果一次工装仅能采集 9 张图的话）。



如果标定条件仅限单次工装的话，建议①增加相机与标定板的距离，控制在正常使用距离附

近（1m 前后），这样更大的视野里，应该可以放置更多的标定板（>12 块）；②在工装不变动的情况下，提高采图质量。

### 话外篇：

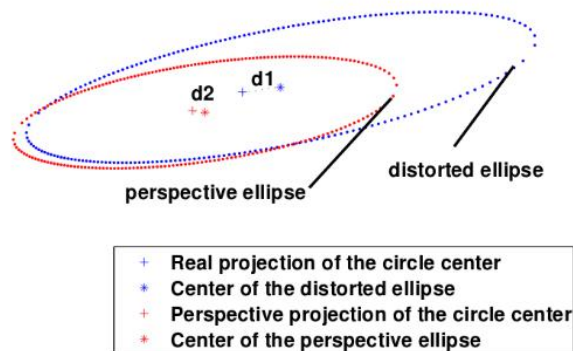
鉴于 OpenCV 中使用鱼眼模型是基于 Kannala&&Brandt 模型，作者在文章中提到，对于圆形的中心在经过透视变换后，由于较大的畸变存在，椭圆的中心已经不再对应圆形的中心。

To avoid the problem above, we propose solving the centroids of the projected circles numerically. We parameterize the interior of the circle at  $(X_0, Y_0)$  with radius  $R$  by  $\mathbf{X}(\varrho, \alpha) = (X_0 + \varrho \sin \alpha, Y_0 + \varrho \cos \alpha, 0)^\top$ . Given the camera parameters, we get the centroid  $\hat{\mathbf{m}}$  for the circle by numerically evaluating

$$\hat{\mathbf{m}} = \frac{\int_0^R \int_0^{2\pi} \hat{\mathbf{m}}(\varrho, \alpha) |\det \mathbf{J}(\varrho, \alpha)| d\alpha d\varrho}{\int_0^R \int_0^{2\pi} |\det \mathbf{J}(\varrho, \alpha)| d\alpha d\varrho}, \quad (19)$$

where  $\hat{\mathbf{m}}(\varrho, \alpha) = \mathcal{P}(\mathbf{X}(\varrho, \alpha))$  and  $\mathbf{J}(\varrho, \alpha)$  is the Jacobian of the composite function  $\mathcal{P} \circ \mathbf{X}$ . The analytical solving of the Jacobian is rather a tedious task but it can be computed by mathematical software such as Maple.

为了更好地看出椭圆构象偏差的效果，可以参考下图：



**Figure 1.** Projection of a circle in perspective model and in real camera model.

在椭圆较大时，构象偏差会较为明显。但是椭圆较小时，我们也可以忽略椭圆偏心的影响。

由于 OpenCV 圆角点提取精度以满足我们的使用要求，对于椭圆构象偏心差的矫正暂且无需作为重点进行研究。

## 参考文献：

- [1] Kannala J, Brandt S S. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses[J]. IEEE transactions on pattern analysis and machine intelligence, 2006, 28(8): 1335-1340.
- [2] Claus, D. and Fitzgibbon, A. W.: “A rational function lens distortion model for general cameras” , Proc. CVPR, pp. 213-219, 2005
- [3] Thirithala, S. and Pollefeys, M.: “Multi-view geometry of 1D radial cameras and its application to omnidirectional camera calibration” , Proc. ICCV, pp. 1539-1546, 2005.
- [4] Barreto, J. P. and Daniilidis, K.: “Fundamental matrix for cameras with radial distortion” ,Proc. ICCV, pp. 625-632, 2005.