

01 基础知识（回顾）

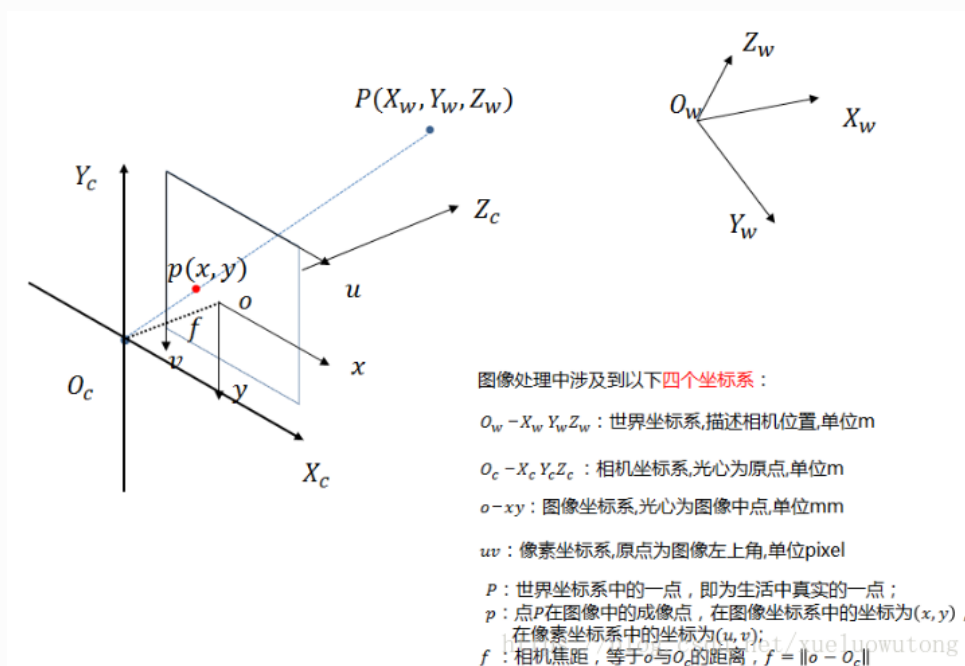
02 OpenCV实践

- 1) 准备标定图片
- 2) 检测角点，并亚像素优化
- 3) 调用张正友标定法
- 4) 纠正图像

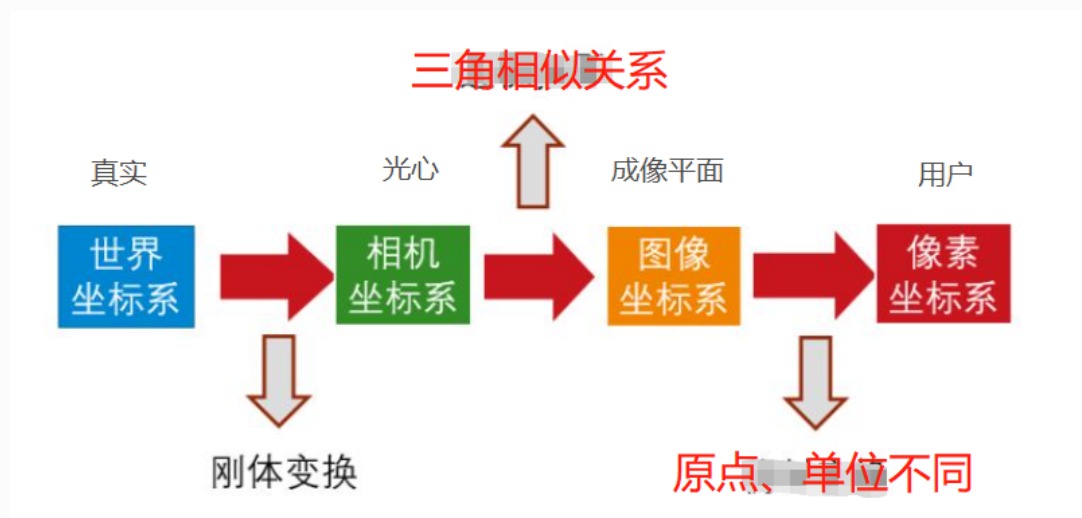
03 注意事项

01 基础知识（回顾）

相机模型如下（如有不明白，请查看之前的推送），主要包含四个坐标系：



四个坐标系的转换关系如下：



而我们主要关心的是从世界坐标系到像素坐标系的变换，它们的公式如下：

$$Z_c \underbrace{\begin{bmatrix} u \\ v \\ 1 \end{bmatrix}}_{\text{像素坐标系}} = \underbrace{\begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}}_{M1: \text{内参}} \underbrace{\begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \end{bmatrix}}_{M2: \text{外参}} \underbrace{\begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix}}_{\text{世界坐标系}} \quad (1)$$

而相机成像过程中存在着径向畸变和切向畸变，合并后：

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right)}_{\text{径向畸变}} \begin{bmatrix} x \\ y \end{bmatrix} + \underbrace{\begin{bmatrix} 2p_1 xy + p_2 (r^2 + 2x^2) \\ 2p_2 xy + p_1 (r^2 + 2y^2) \end{bmatrix}}_{\text{切向畸变}} \quad (2)$$

为了之后三维重建算法的精确性，我们必须对这些畸变进行矫正。我们可以通过张正友标定法得到相机的**畸变系数**，除此之外还有它的**内参**。获得了这两个参数，我们就能对相机的整个成像过程建立较为准确的映射关系（至少是单向的），尽管具体的矫正实现比较复杂，但好在OpenCV很多过程都帮我们封装好了。

02 OpenCV实践

下面就开始实践，我们来实现一个基于OpenCV的实践，具体的步骤如下：

1. 准备标定图片（不同位置、角度、姿态下拍摄，至少需要3张，以10~20张为宜）
2. 对每一张标定图片，提取角点信息，并进行亚像素角点优化（绘制相应图像）
3. 调用张正友标定方法，对相机进标定，也就是计算内参和畸变系数
4. 对标定结果进行评价（重投影误差）
5. 利用标定结果（内参、畸变系数），对任何输入的图像进行矫正

更多的请直接看代码，相信在明白了理论之后，阅读我们的代码应该非常容易（尽量命名做到望文知义、封装做到了面对对象）。关键的代码如下：

1) 准备标定图片

用于标定的图片如下，我们需要知道它的包括三个信息，某行/列的内角点数量，以及每个棋盘格的大小：

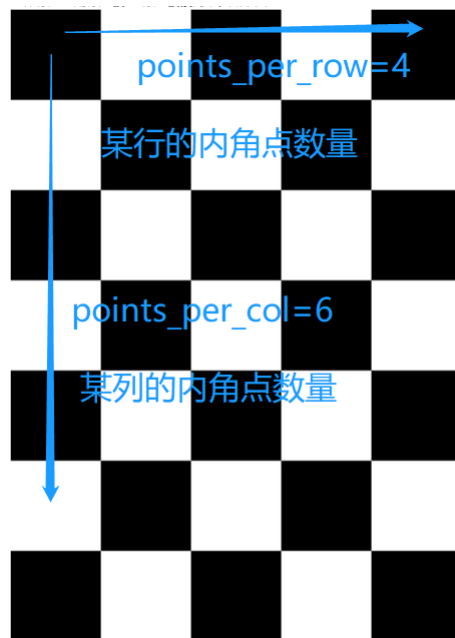


图1 标定板图像

在我们的代码中，即以下函数：

```
1 // 读取所有标定图片到calib_images <vector> 中
2 Utils::list_files(calib_folder, calib_images);
```

2) 检测角点，并亚像素优化

```
1 // 粗检测
2 bool found = cv::findChessboardCorners(img, patten_size, corners_pixel);
3 // 迭代标准
4 cv::TermCriteria criteria = cv::TermCriteria(CV_TERMCRIT_EPS + CV_TERMCRIT_ITER,
5 30, 0.1);
6 // 迭代优化
7 cv::cornerSubPix(img_gray, corners_pixel, cv::Size(5, 5), cv::Size(-1, -1),
8 criteria);
```

3) 调用张正友标定法

获取相机内参矩阵、畸变矩阵：

```
1 calib_pro_error = cv::calibrateCamera(
2     checkboard.corners_world_all, // 世界坐标系点, vector<vector<cv::Point3f> >
3     checkboard.corners_pixel_all, // 图像坐标系点(对应的),
4     vector<vector<cv::Point2f> >
5     checkboard.calib_images_size, // 标定图像尺寸大小
6     cameraMatrix, // 相机内参矩阵
7     distCoeffs, // 畸变矩阵
8     _rVecs, // 旋转向量
9     _tVecs // 平移矩阵
10 );
```

4) 纠正图像

利用得到的内参矩阵、畸变系数对图像进行矫正：

```
1 cv::Mat img_rectify = img.clone();  
2 cv::undistort(img, img_rectify, cameraMatrix, distCoeffs);
```

03 注意事项

拍摄图片的有以下注意事项：

1. 用于标定的标定板制作精度以一定要高，否则影响精度；
2. 棋盘格与相机之间的夹角要小于 45° ，否则误差较大；
3. 应使视场大小与标定板大小相当，或者稍微大一点，并且保证成像清晰、曝光适当；
4. 标定板所处离相机的距离，应与平时相机的工作距离相近；