

视觉SLAM十四讲笔记

第五讲 - 相机与图像

相机模型

针孔相机模型

- 相机内参矩阵 \mathbf{K} :

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \frac{1}{Z} \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \stackrel{\text{def}}{=} \frac{1}{Z} \mathbf{K} \mathbf{P}$$

- 相机外参矩阵 \mathbf{T} :

$$Z \mathbf{P}_{uv} = Z \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{K} (\mathbf{R} \mathbf{P}_w + \mathbf{t}) = \mathbf{K} \mathbf{T} \mathbf{P}_w$$

- 归一化坐标:

$$(\mathbf{R} \mathbf{P}_w + \mathbf{t}) = \underbrace{[X, Y, Z]^T}_{\text{相机坐标}} \rightarrow \underbrace{[X/Z, Y/Z, 1]^T}_{\text{归一化坐标}}.$$

畸变模型

- 径向畸变

- 桶形畸变
- 枕形畸变

-

$$\begin{aligned} x_{\text{distorted}} &= x (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \\ y_{\text{distorted}} &= y (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \end{aligned}$$

- 切向畸变

-

$$\begin{aligned} x_{\text{distorted}} &= x + 2p_1 xy + p_2 (r^2 + 2x^2) \\ y_{\text{distorted}} &= y + p_1 (r^2 + 2y^2) + 2p_2 xy \end{aligned}$$

- 通过5个畸变系数, 将归一化坐标 $[x, y]$ 转换为像素坐标:

- $$\begin{cases} x_{\text{distorted}} = x (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1 xy + p_2 (r^2 + 2x^2) \\ y_{\text{distorted}} = y (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + p_1 (r^2 + 2y^2) + 2p_2 xy \end{cases}$$

- $$\begin{cases} u = f_x x_{\text{distorted}} + c_x \\ v = f_y y_{\text{distorted}} + c_y \end{cases}$$

双目相机模型

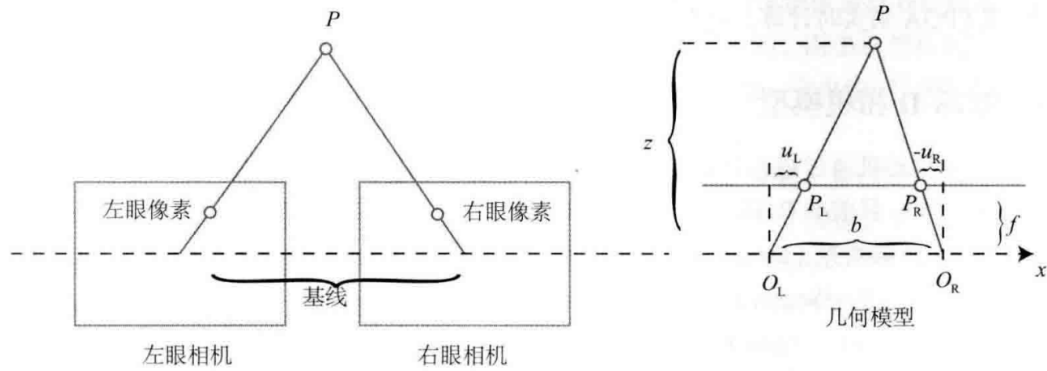


图 5-6 双目相机的成像模型。 O_L, O_R 为左右光圈中心，方框为成像平面， f 为焦距。 u_L 和 u_R 为成像平面的坐标。请注意，按照图中坐标定义， u_R 应该是负数，所以图中标出的距离为 $-u_R$

$$\frac{z-f}{z} = \frac{b-u_L+u_R}{b}$$
$$z = \frac{fb}{d}, \quad d \stackrel{\text{def}}{=} u_L - u_R$$

RGB-D 相机模型

- 红外结构光 Structured light
 - Kinect Gen1
 - Intel Realsense
- 飞行时间 ToF Time-of-Flight
 - Kinect Gen2
 - ToF cam

CPP Demo

- imageBasics
 - imageBasics
 - 本程序介绍了 OpenCV 的一些基本操作
 - undistortImage
 - 本程序通过 OpenCV 对图像进行去畸变处理
- stereo
 - 本程序通过一对左右相机图像，生成 SGMB 视差图，并变换到 3D 空间，通过点云显示。
- rgbd
 - 本程序通过将5副彩色图对应的深度图结合他们的pose，计算并拼接点云，合成一个场景点云。