# ORBSLAM3 系列代码讲解

## 地图点 专题一

主 讲 人：魏宏宇

公 众 号：3D视觉工坊

# 主要内容

1 3D空间点的几何形成原理

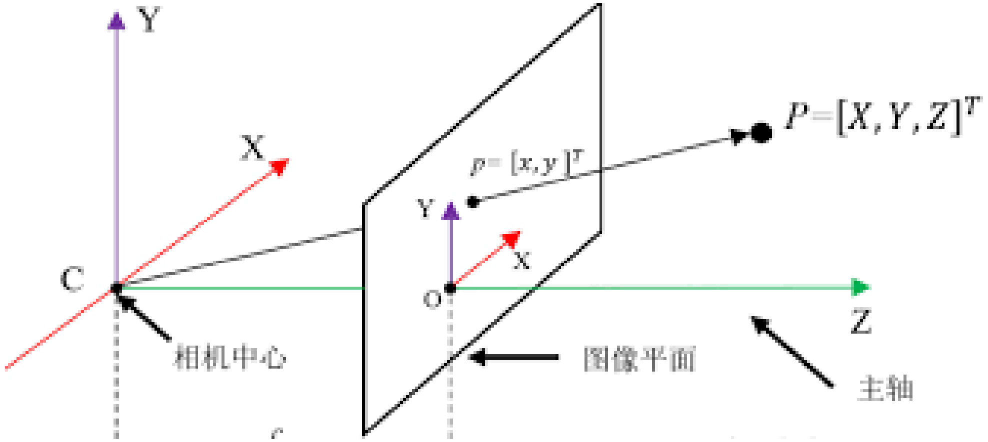
2 SLAM中涉及的坐标系明确

3 地图点属性

4 地图点代码讲解

5 讨论与交流
（如何有效缓解科研焦虑，在焦虑中成长）

# 1 3D空间点的几何形成原理

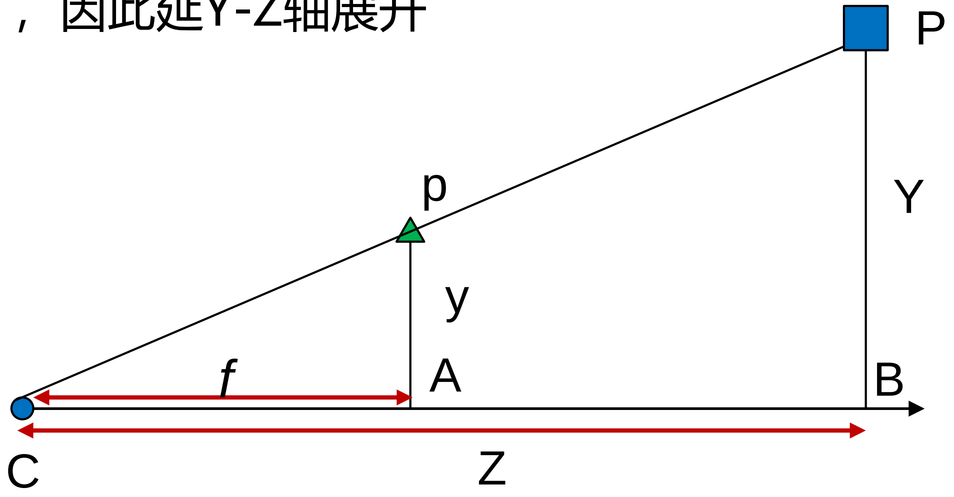



空间点 $P=[X,Y,Z]^T$

像平面投影点 $p=[u,v]^T$

像平面/焦平面： 平行于相机，$Z=f$（焦距）的平面

由于像平面平行于相机坐标系的X轴，因此延Y-Z轴展开




在Y-Z平面上，同一个地图点、投影点以及相机中心之间的空间几何结构内，存在一个相似三角形：

$CpA \simeq CPB$

$$\frac{y}{f}=\frac{Y}{Z} \Rightarrow y=\frac{Y}{Z}f$$

同理，若按照XZ平面展开，可得到

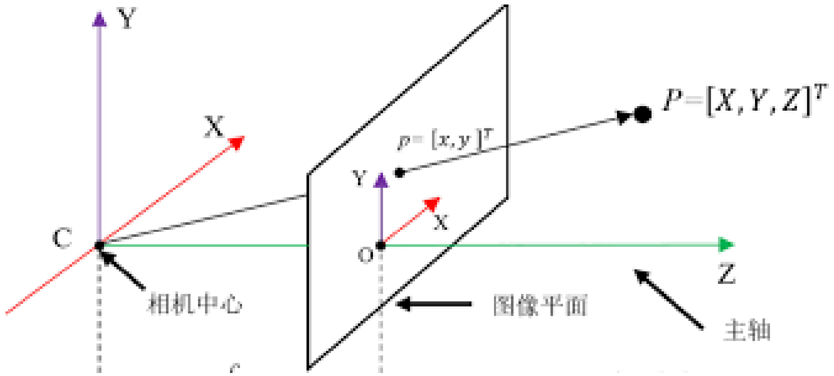
$$\frac{x}{f}=\frac{X}{Z} \Rightarrow x=\frac{X}{Z}f$$

因此，3D空间点-2D像素点之间存在如下关系：

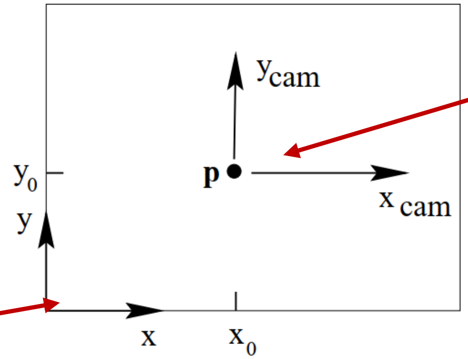
$$[X,Y,Z]^T \mapsto \left[f\frac{X}{Z}, f\frac{Y}{Z}\right]^T$$

使用齐次坐标表示（扩展维数，以形成等式关系）

$$\begin{bmatrix} \frac{fX}{Z} \\ \frac{fY}{Z} \\ 1 \end{bmatrix} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{f}{Z} & 0 & 0 \\ 0 & \frac{f}{Z} & 0 \\ 0 & 0 & \frac{1}{Z} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

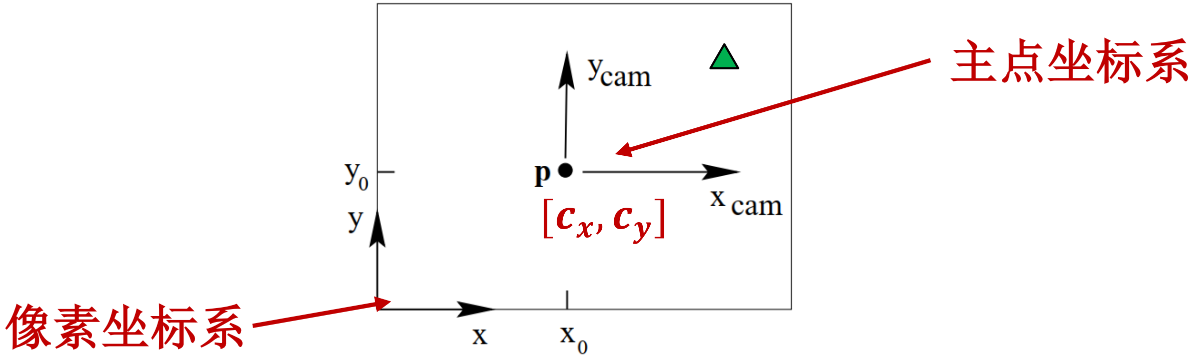



这里有一个默认假设，就是像平面中，像素坐标系（x,y的值）XOY是沿Z轴，空间上相当于相机坐标系的平移。
实际上不是这样




理想情况下

实际情况下




主点坐标系下： $u = \frac{fX}{Z}, v = \frac{fY}{Z}$

像素坐标系下： $u = \frac{fX}{Z} + c_x$， $v = \frac{fY}{Z} + c_y$

3D空间点-2D像素点(像素坐标系)之间存在如下关系：

$$[X, Y, Z]^T \mapsto \left[ f\frac{X}{Z} + c_x, \quad f\frac{Y}{Z} + c_y \right]^T$$

同上，引入齐次坐标

$$\begin{bmatrix} \frac{fX}{Z} + c_x \\ \frac{fY}{Z} + c_y \\ 1 \end{bmatrix} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{f}{Z} & 0 & \frac{c_x}{Z} \\ 0 & \frac{f}{Z} & \frac{c_y}{Z} \\ 0 & 0 & \frac{1}{Z} \end{bmatrix}}_{\text{内参矩阵K}} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

当前相机坐标系下的空间点坐标

当前像素坐标系下特征的2D位置

由于相机的内参矩阵大多数情况下服从出厂设置，因此在SLAM中认为相机内参矩阵已知。

**利用此公式，可以实现2D特征点到当前空间坐标系的地图点转换**

SLAM中的正向投影：3D点(X,Y,Z)到2D特征点（u，v）

$$Zp_c = KP_c$$

$$\begin{cases} u = f_x \dfrac{X}{Z} + c_x \\ v = f_y \dfrac{Y}{Z} + c_y \end{cases}$$

SLAM中的反向投影：自相机中心出发，经2D(u,v)特征点形成空间射线，3D地图点位于射线上的某一点（X,Y,Z）

$$P_c = K^+ p_c \quad K^+\text{是K的伪逆}$$

$$\begin{cases} X = (u - c_x)/f_x \\ Y = (v - c_y)/f_y \\ Z = 1 \end{cases}$$

- **深度为1的3D地图点**
- **以相机为中心，1为半径的球面上的一点**
- **以相机为中心，过特征点的一条射线**

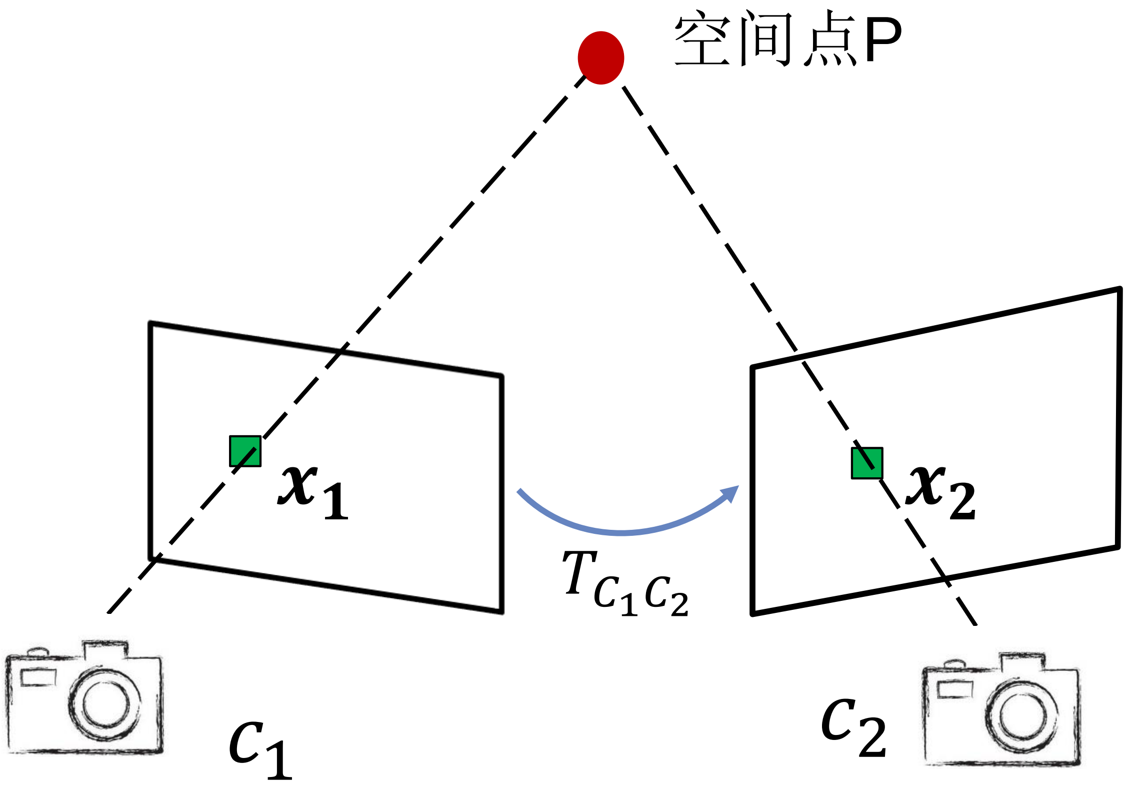
$$\begin{cases} X = (u - c_x)/f_x \\ Y = (v - c_y)/f_y \\ \quad Z = 1。 \end{cases}$$

缺失深度信息

基于视觉的SLAM最终目标是求解相机运动的空间姿态变化。从2D图像到位姿求解，不可避免的是需要大量的空间地图点作为位姿求解的桥梁，因此空间中点的3D位置的精度就变得尤为重要

## 三角测量法

双目相机由于可以在同一时刻获得不同视场的两张图像，因此相比于单目可以更好的估计场景深度信息，因此对于单张图像的地图点位置估计，三角测量法利用单目相机在不同位置对同一场景的图像信息，实现场景深度估计。




根据之前的匹配系列的介绍，可以获得两张图像的2D匹配点以及相关的姿态变换

$$sx_1 = K * P_{C_1}$$

$$sx_2 = K * P_{C_2}$$

# DLT直接线性变换

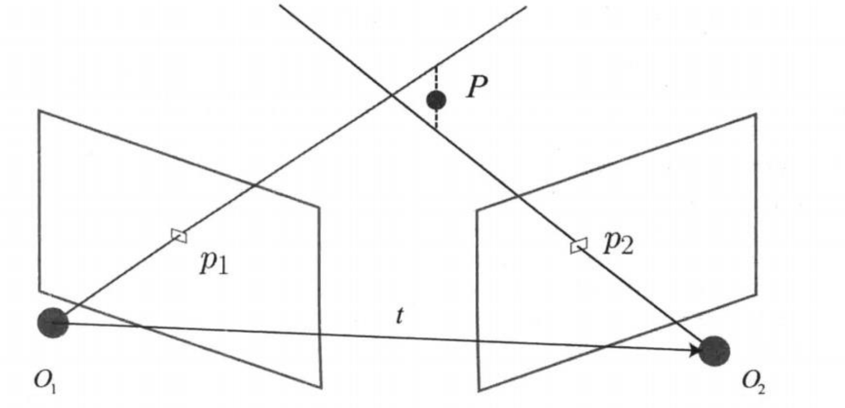



图 7-9 三角化获得地图点深度



**当前已知条件:**

图像1的位姿：$T_{C_1W}$ 图像2的位姿：$T_{C_2W}$

匹配成功的点对像素坐标$p_1(u_1,v_1)$ $p_2(u_1,v_1)$

**目标:**

获得空间点P的世界坐标

根据相机摄影原理，2D图像坐标与3D空间点存在如下关系

$$\boxed{s\begin{pmatrix}u_1\\v_1\\1\end{pmatrix}=\left(KT_{C_1W}P_W\right)_{1:3}}$$

齐次坐标与非齐次坐标之间的变换

$$s\begin{pmatrix}u_1\\v_1\\1\end{pmatrix}=M_{3*4}P_W$$

拆开并用最后一行尺度因子的关系消去尺度因子对前两行的影响

$$M_2P_Wu_1-M_0P_W=0$$
$$M_2P_Wv_1\ -M_0P_W=0$$

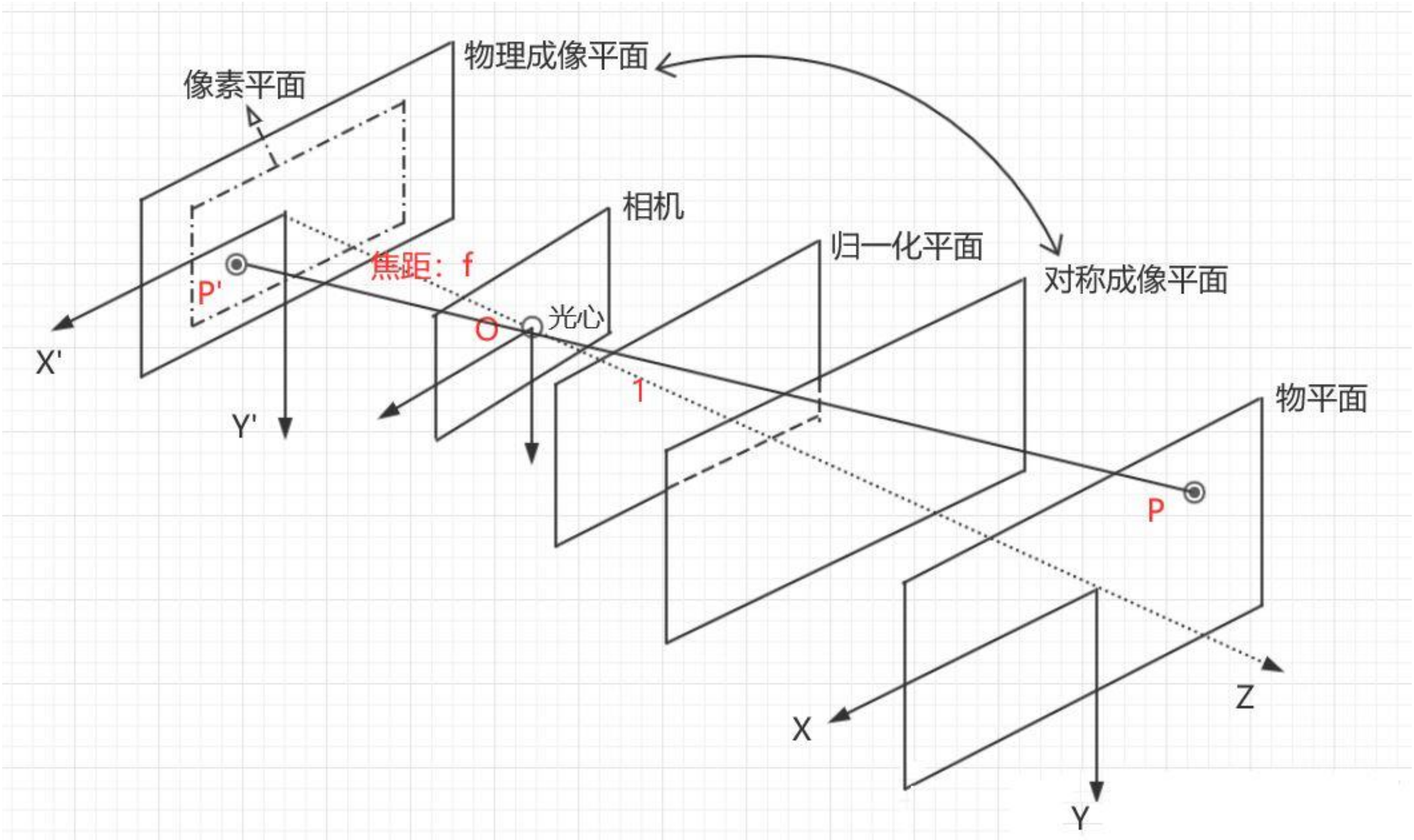
图2 同理

$$M_2'P_Wu_2-M_0'P_W=0$$
$$M_2'P_Wv_2\ -M_0'P_W=0$$

SVD分解求解超定方程获得空间点坐标

TwoViewReconstruction.cc中
Trangulate函数

# 2 SLAM中涉及的坐标系明确

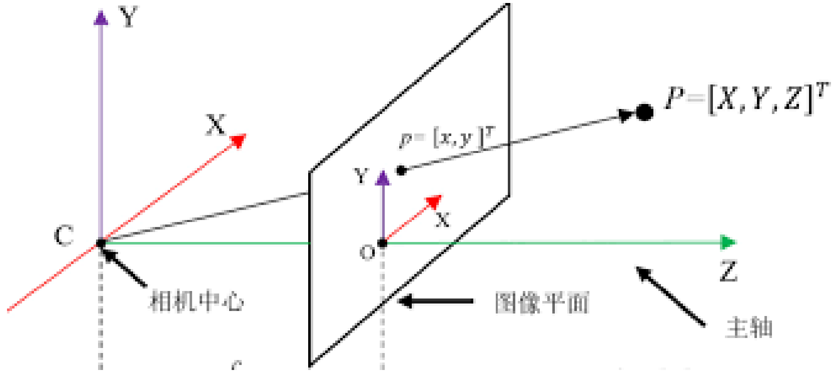

物理成像平面
像素平面
相机
归一化平面
对称成像平面
物平面
焦距：f
P'
O
光心
1
P
X'
Y'
X
Y
Z

- 世界坐标系->相机坐标系

$$\begin{bmatrix}X_C\\Y_C\\Z_C\end{bmatrix} = R_W^C\begin{bmatrix}X_W\\Y_W\\Z_W\end{bmatrix} + t_W^C \qquad \begin{bmatrix}X_C\\Y_C\\Z_C\\1\end{bmatrix} = \begin{bmatrix}R_W^C & T_w^c\\0_{1*3} & 1\end{bmatrix}\begin{bmatrix}X_W\\Y_W\\Z_W\\1\end{bmatrix}$$

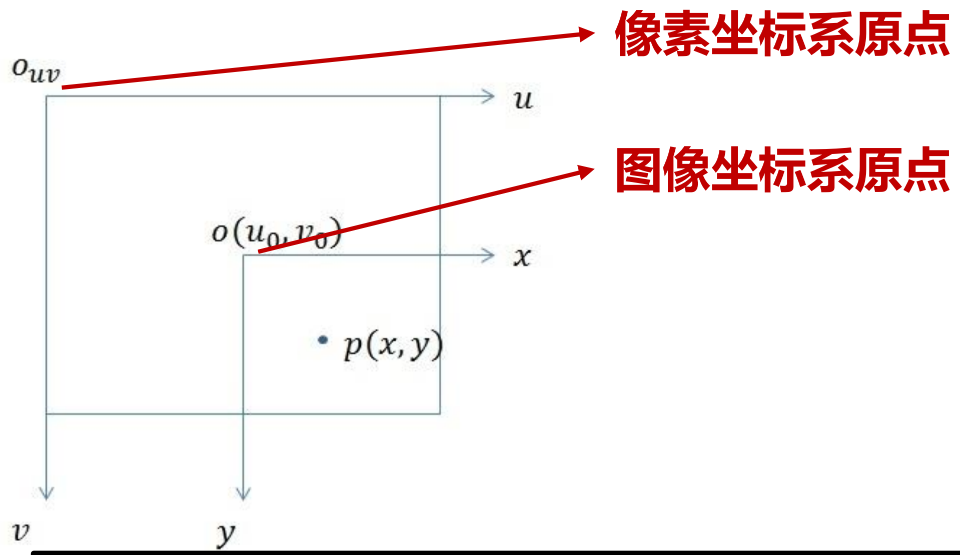
- 相机坐标系 -> 图像坐标系/物理平面(之间构造的小孔成像原理中的以相机为中心与原点的平面坐标系)

$$\begin{cases}x = f\dfrac{X_C}{Z_C}\\ y = f\dfrac{Y_C}{Z_C}\end{cases} \qquad Z_C\begin{bmatrix}x\\y\\1\end{bmatrix} = \begin{bmatrix}f & 0 & 0 & 0\\0 & f & 0 & 0\\0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}X_C\\Y_C\\Z_C\\1\end{bmatrix}$$



# SLAM中涉及的坐标系明确

- 图像坐标系 -> 像素坐标系




像素坐标系和图像坐标系之间存在一个度量单位的变换，即$u * dx = x$， $v * dy = y$

$$\begin{cases} u = \dfrac{x}{dx} + c_x \\ v = \dfrac{y}{dy} + c_y \end{cases}$$

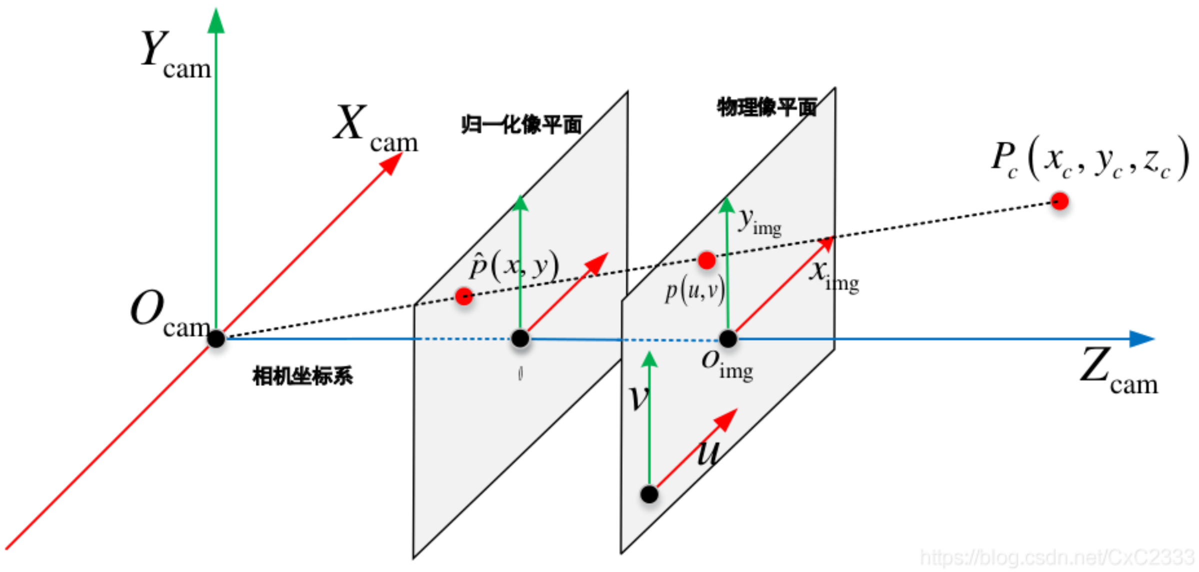
$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{dx} & 0 & c_x \\ 0 & \frac{1}{dy} & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\boldsymbol{Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{dx} & 0 & c_x \\ 0 & \frac{1}{dy} & c_y \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} * \begin{bmatrix} R_W^C & T_w^c \\ 0_{1*3} & 1 \end{bmatrix} \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix}}$$

- 归一化平面 距离相机中心的1的平面，平行于像平面 [理解球面的意义]

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \frac{1}{Z_c} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix}$$

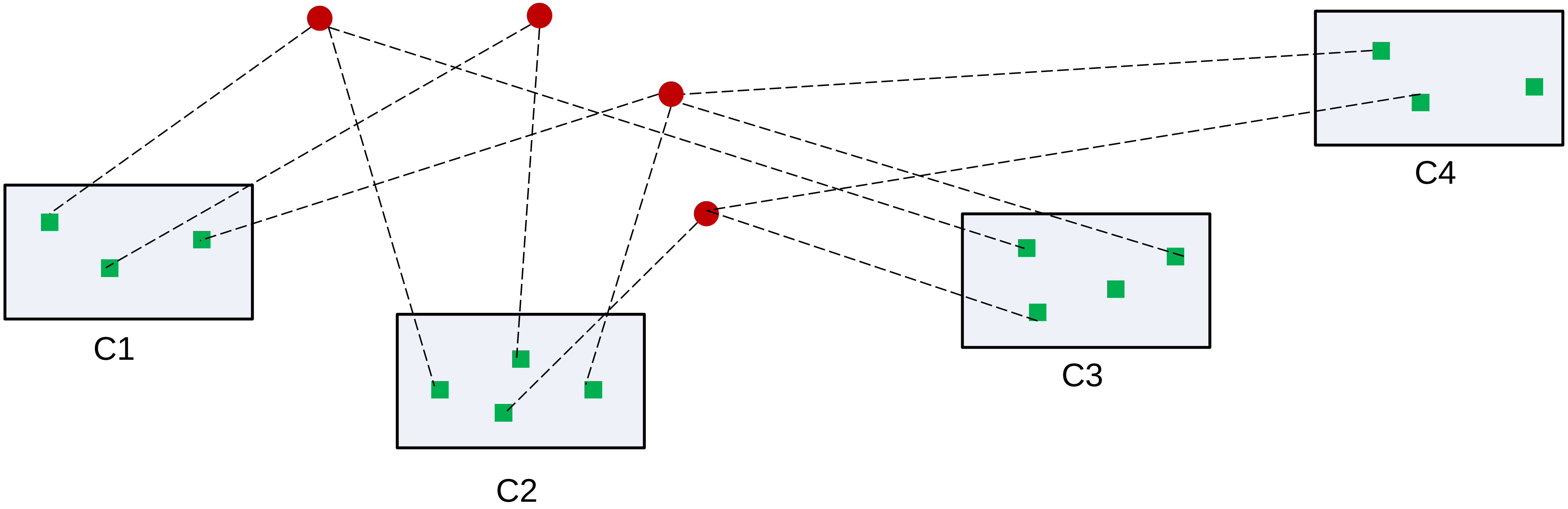
归一化平面和像平面之间的区别在于，归一化平面位于Z=1的位置，像平面位于Z=f的位置



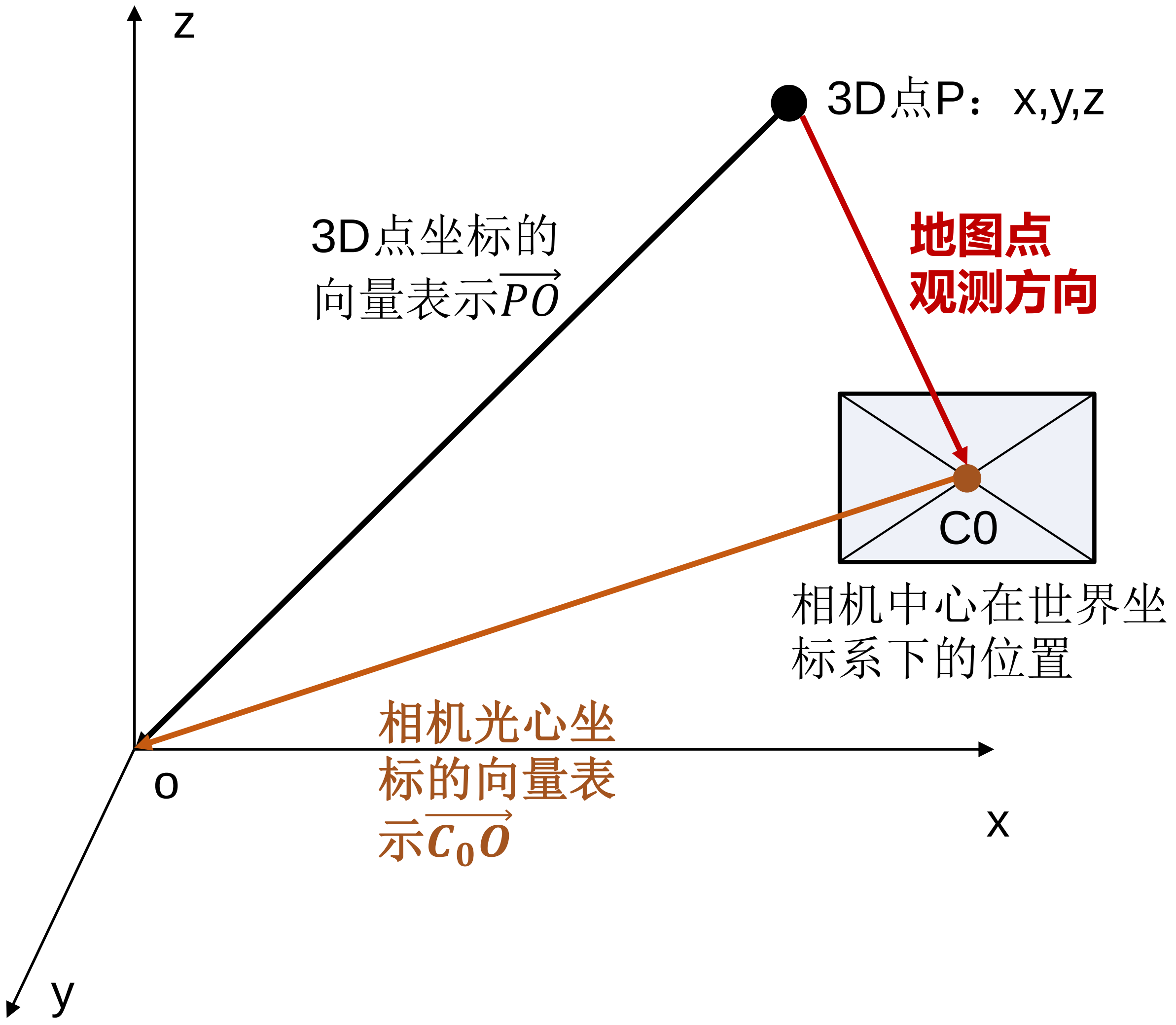
# 3 地图点属性

SLAM中一般地图点是被长期保存在系统中，用于跟踪和轨迹约束，因此需要对地图点设置很多属性，以区分不同的地图点，并更好的用于优化

- 观测：观测到地图点的图像帧以及对应的特征点



- 地图点的观测方向：



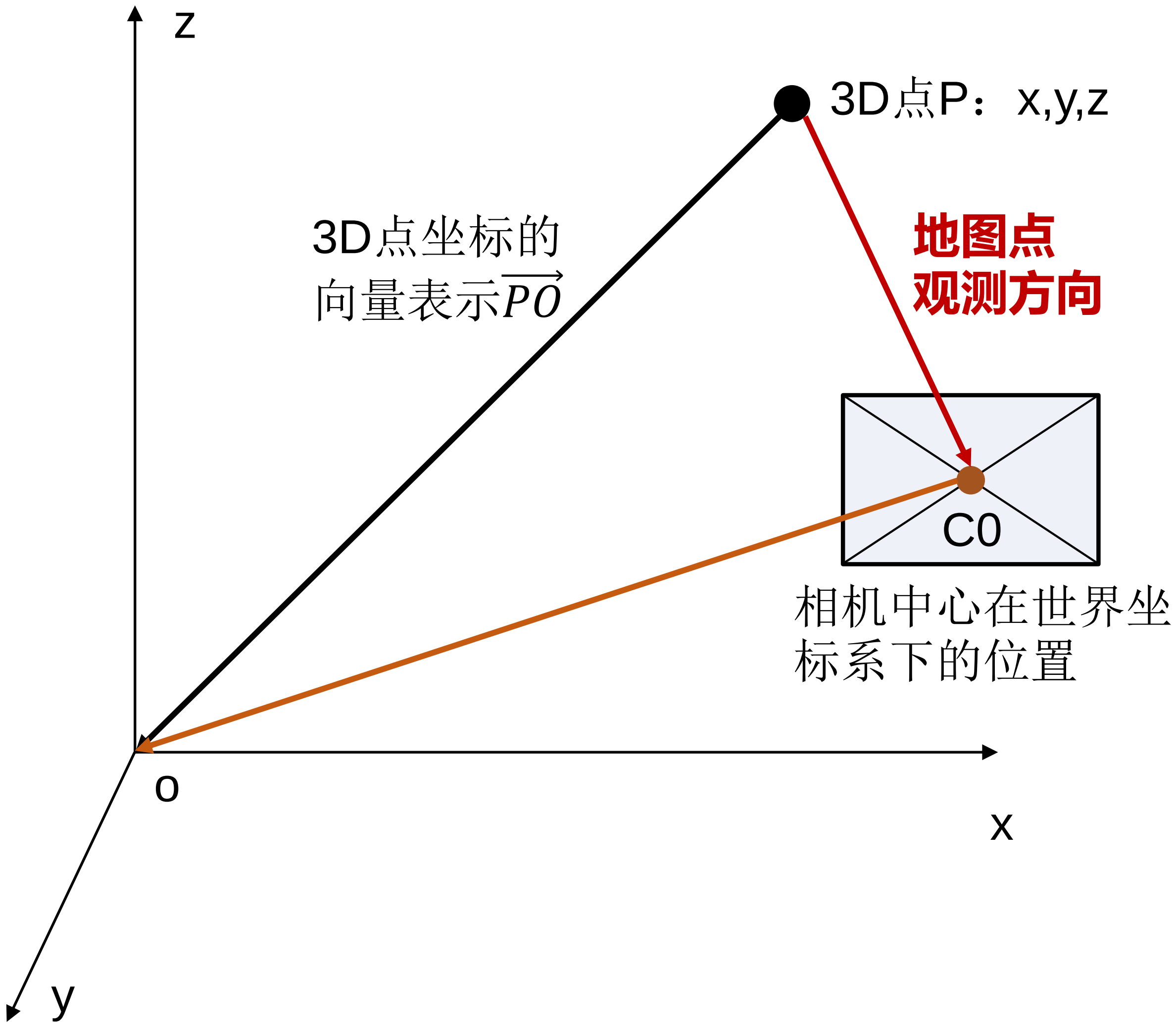

观测方向：

$$Vector = \overrightarrow{PO} - \overrightarrow{C_0O} = \overrightarrow{PC_O}$$

归一化（方向向量）：

$$normal = \frac{Vector}{\|Vector\|} = \frac{\overrightarrow{PC_O}}{\|\overrightarrow{PC_O}\|}$$

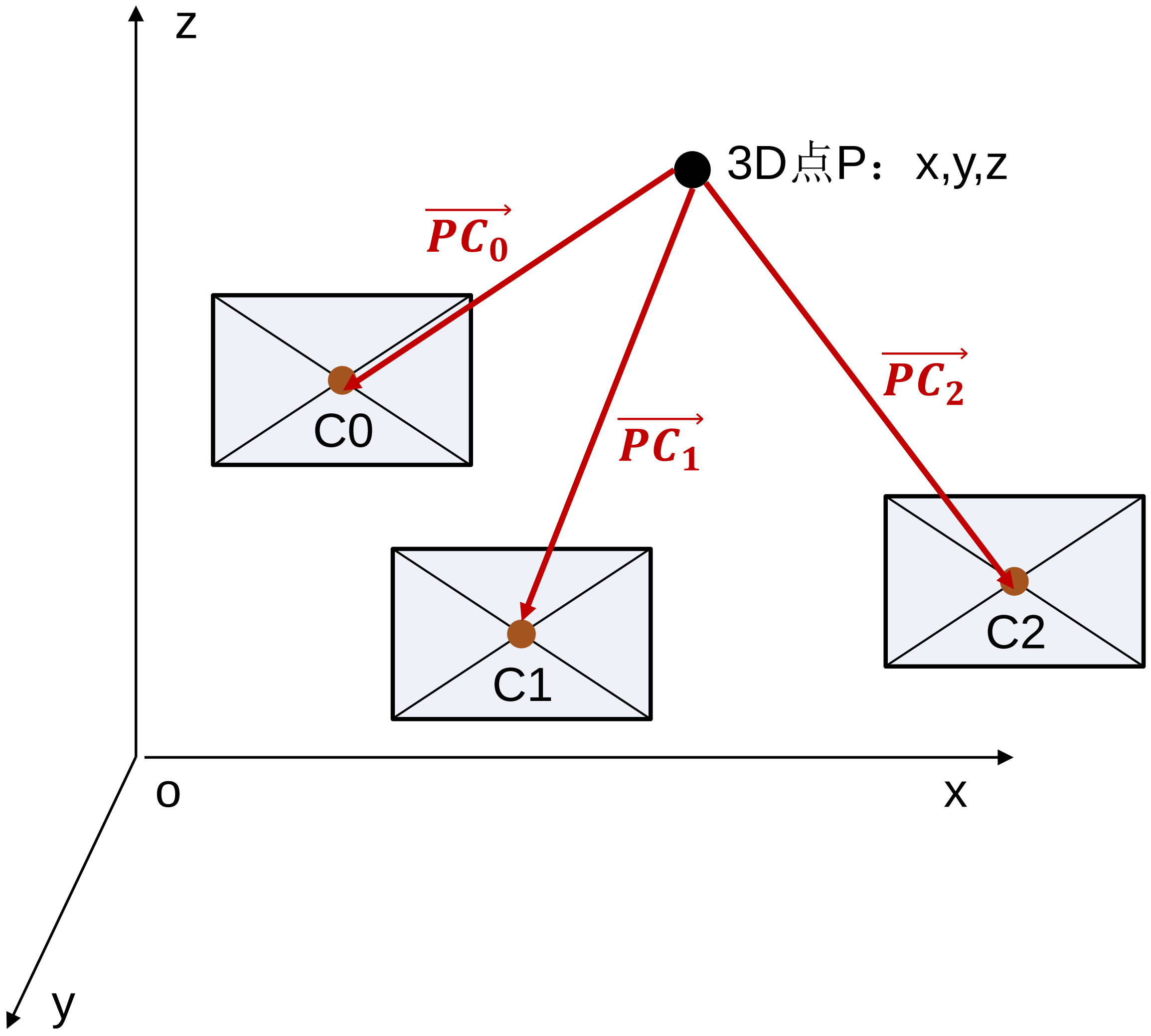
- 地图点的距离：




地图点到图像的距离：

$$dis = \|\overrightarrow{PC_0}\|$$

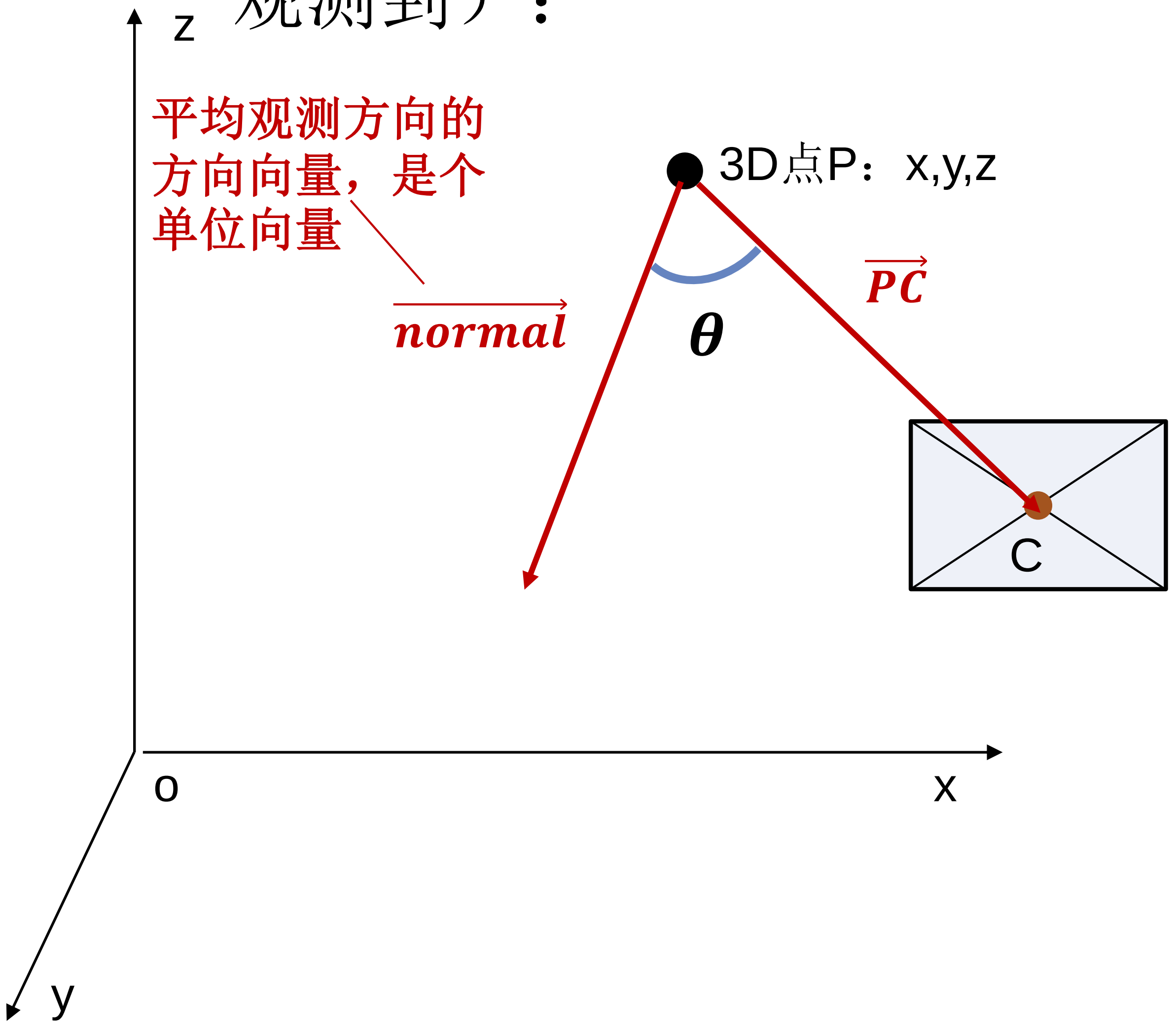
- 地图点的平均观测方向：




平均观测方向：

$$\vec{v} = \frac{1}{3} * \left( \frac{\overrightarrow{PC_0}}{\left\| \overrightarrow{PC_0} \right\|} + \frac{\overrightarrow{PC_1}}{\left\| \overrightarrow{PC_1} \right\|} + \frac{\overrightarrow{PC_2}}{\left\| \overrightarrow{PC_2} \right\|} \right)$$

- 地图点的单帧观测余弦值（用来判断地图点是否能被当前图像帧所观测到）：




根据两个向量之间的夹角余弦定理：

$$cos\theta = \frac{\vec{a} * \vec{b}}{\|\vec{a}\| * \|\vec{b}\|}$$

# 地图点属性

- 地图点的深度：
  一般指的是地图点在当前图像空间坐标系下的3D坐标的范数

$$depth = \left\| \begin{bmatrix} X_C & Y_C & Z_C \end{bmatrix}^T \right\|$$

# 4 地图点代码讲解

# 5 讨论与交流

（如何有效缓解科研焦虑，在焦虑中成长）

# 欢迎关注3D视觉工坊

公众号

我们这里有3D视觉算法、SLAM、点云处理、三维重建、计算机视觉、深度学习、自动驾驶、图像处理、技术干货以及前沿paper分享！

交流群请添加客服

如果你也想成为主讲人，欢迎加入我们。

➢ 报名方式：请发送邮件至vision3d@yeah.net

**客服微信，咨询课程**

**3D视觉工坊知识星球**

- 课程PPT和注释代码
- 补充知识点　PDF版和视频版
- 答疑

# 感谢聆听

Thanks for Listening