CS 4495 计算机视觉

校准与投影几何 (1)

亚伦-博比克

互动计算机学院

行政管理

- 问题集 2:
  - 查找 PDF???? 有什么问题?

    http://www.cc.gatech.edu/~afb/classes/CS4495-Fall2013/

    或

    http://www.cc.gatech.edu/~afb/classes/CS4495-Fall2013/ProblemSets/PS2/ps2-descr.pdf

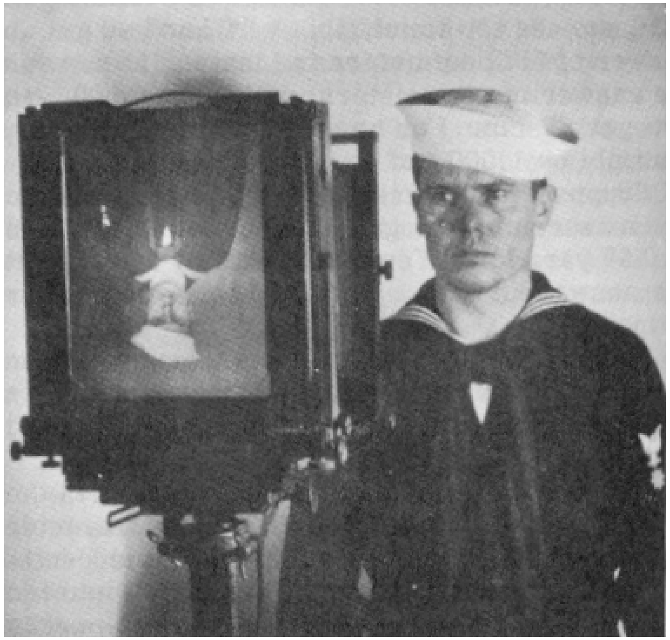
- 今天：真正使用同质系统来表示投影。以及如何进行校准。

- Forsyth 和 Ponce，1.2 和 1.3

上次...

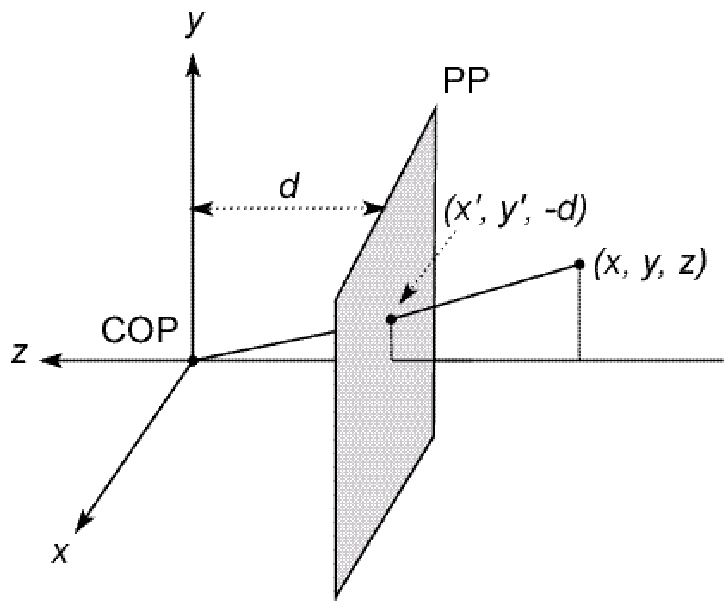
什么是图像?

- 上一次：函数 - 强度值的二维模式
- 这次：三维点的二维投影



图自海军人事局编写的《美国海军基本光学和光学仪器手册》。1969 年由多佛尔出版公司重印。

建模预测



- 坐标系
  - 我们将使用针孔模型作为近似值
  - 将光学中心（投影中心）置于原点
  - 将图像平面（投影平面）*置于* COP 前方

    - 为什么?
- 摄像机*沿负* Z 轴向下拍摄
    - 如果我们想要右旋坐标，就需要它

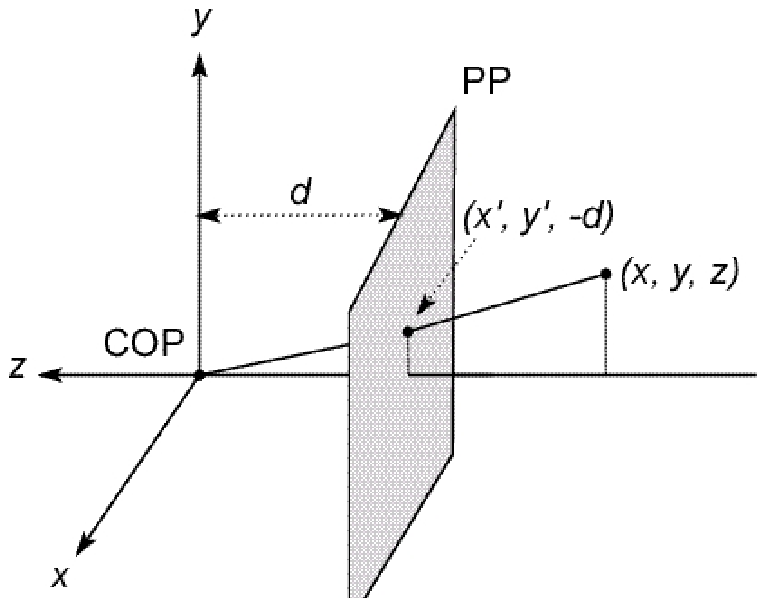
建模预测

- 投影方程
  - 计算从 (x,y,z) 到 COP 的射线与 PP 的交点
  - 利用相似三角形得出

远处的物体更小

$$(x, y, z) \rightarrow (-d\frac{x}{z},\ -d\frac{y}{z},\ -d)$$

- 我们抛出最后一个坐标，就得到了推算结果：



$$(x, y, z) \rightarrow (-d\frac{x}{z},\ \ -d\frac{y}{z})$$

或者

- 假设焦距为正数，z 为距离：

$$x' = u = f\frac{x}{|z|}$$

$$y' = v = f\frac{y}{|z|}$$

均质坐标

- 这是线性变换吗?
  - 否 - 除以 Z 是非线性的

技巧：增加一个坐标：

$$(x, y) \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

同质图像（二维）坐标

$$(x, y, z) \Rightarrow \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

同质场景（三维）坐标

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} \Rightarrow (x/w, y/w) \qquad \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} \Rightarrow (x/w, y/w, z/w)$$

从均质坐标转换

尺度不变的同质坐标

透视投影

- 投影是使用同质坐标的矩阵乘法：

$$\begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} fx \\ fy \\ z \end{bmatrix} \Rightarrow \left(f\frac{x}{z}, f\frac{y}{z} \right)$$

$$\Rightarrow (u, v)$$

这就是所谓的透视投影

- 该矩阵是投影矩阵

- 矩阵的最大定义比例为

S.塞茨

几何相机校准

用相机告诉你关于这个世界的事情：

- 世界坐标与图像坐标之间的关系：*几何相机校准，参见* Forsyth 和 Ponce、

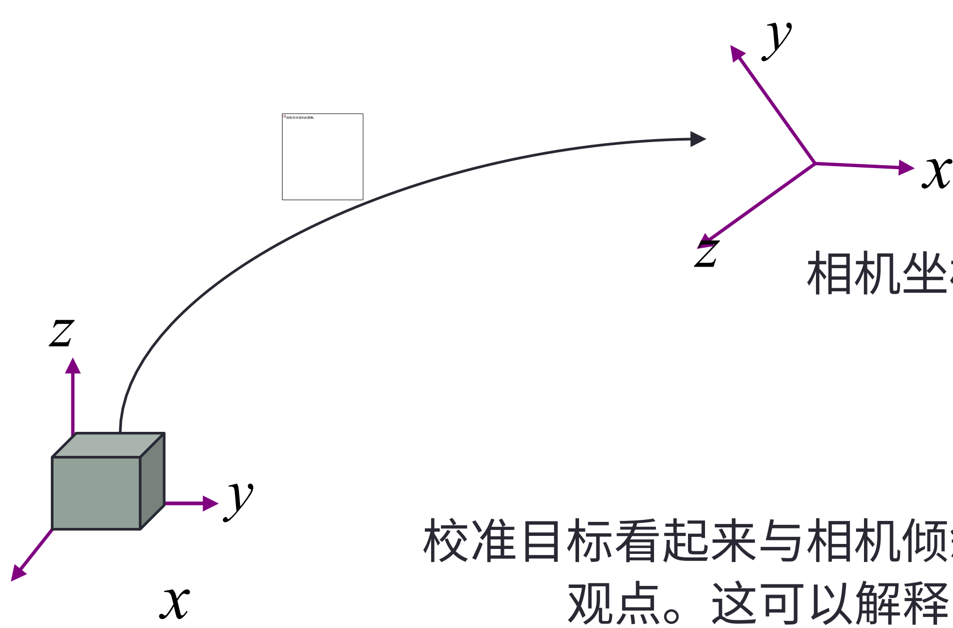
  1.2 和 1.3。另参见 Szeliski 第 5.2 和 5.3 节。

- 由 2 个变换组成：
  - 从某个（任意的）世界坐标系到摄像机的 3D 坐标系。***外部参数***（*摄像机姿态*）

- 通过投影将摄像机框架中的 3D 坐标转换为 2D 图像平面。***内部参数***

相机姿势

为了应用摄像机模型，场景中的物体必须用*摄像机坐标*表示。



相机坐标

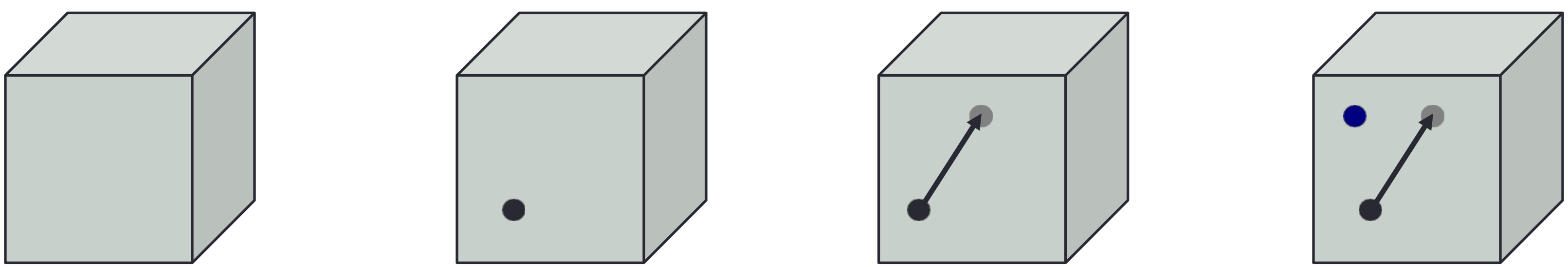
世界坐标

校准目标看起来与相机倾斜观点。这可以解释为

坐标系的差异。

刚体变形

- 需要一种方法来指定刚体的六个自由度。
- 为什么是 6 DOF?



刚体是点的集合，其相对位置不会改变

固定一个点，三个 DOF

3

固定第二个点，增加两个 DOF（必须保持距离约束）

+2

第三点

增加了

一个

DOF，

用于围

绕直线

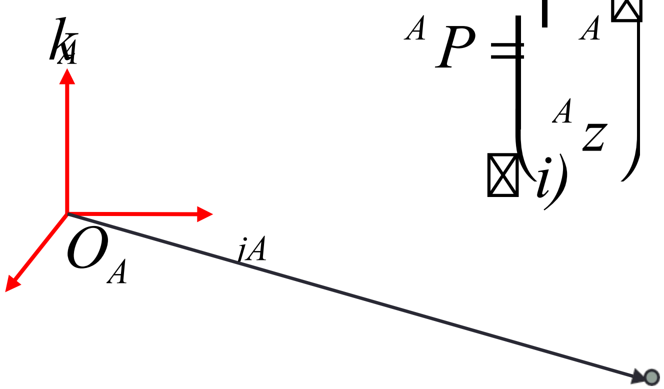
旋转

+1

注释（来自 F&P）

- 上标引用坐标框架
- AP 是 P 在 A 帧中的坐标
- BP 是 P 在 B 帧中的坐标

$$ {}^{A}P = \begin{pmatrix} {}^{A}x \\ {}^{A}y \\ {}^{A}z \end{pmatrix} \Leftrightarrow \overline{OP} = \left({}^{A}x \cdot \overline{i_A}\right) + \left({}^{A}y \cdot \overline{j_A}\right) + \left({}^{A}z \cdot \overline{k_A}\right) $$



iA

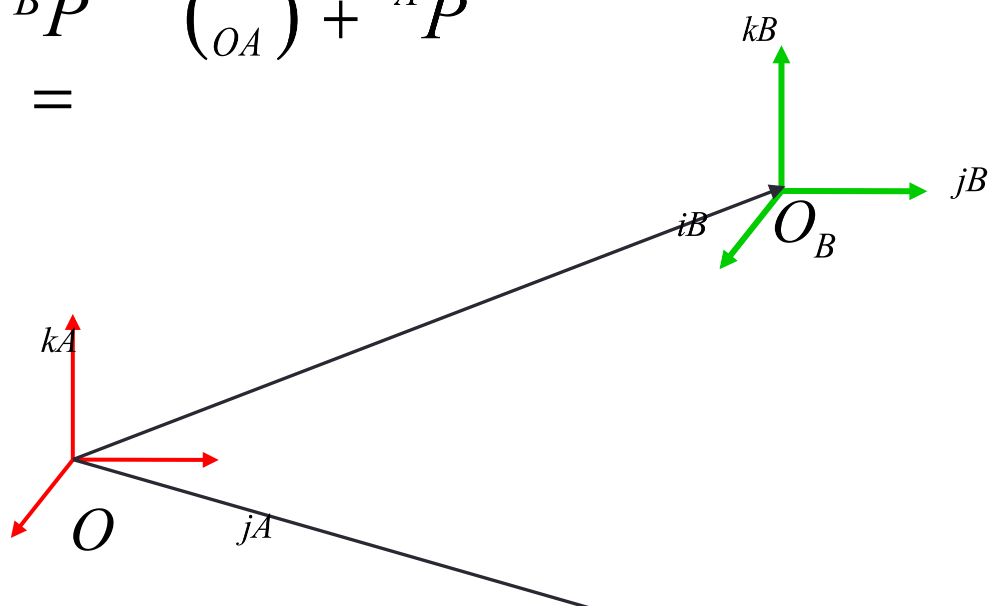
P

仅限翻译

$$^{B}P = {}^{A}P + {}^{B}(O_{A})$$

或

$$^{B}P = {}^{B}(O_{A}) + {}^{A}P$$



A

i_A

P

翻译

- 使用同质坐标，平移可以用矩阵乘法表示。

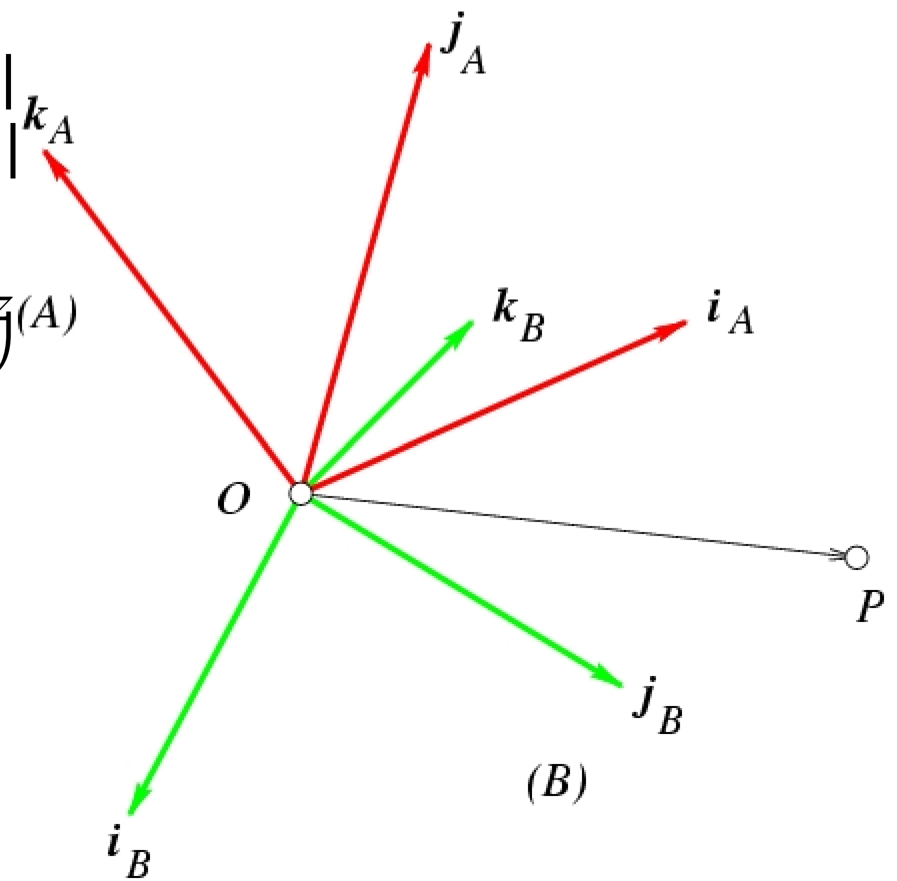
$$ {}^{B}P = {}^{A}P + {}^{B}O_{A} $$

$$ \begin{bmatrix} {}^{B}P \\ 1 \end{bmatrix} = \begin{bmatrix} I & {}^{B}O_{A} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} {}^{A}P \\ 1 \end{bmatrix} $$

- 翻译是交换式的

旋转

$$\overrightarrow{OP} = \begin{pmatrix} i_A & j_A & k_A \end{pmatrix} \begin{pmatrix} {}^{A}x \\ {}^{A}y \\ {}^{A}z \end{pmatrix} = \begin{pmatrix} i_B & j_B & k_B \end{pmatrix} \begin{pmatrix} {}^{B}x \\ {}^{B}y \\ {}^{B}z \end{pmatrix}$$



$${}^{B}P = {}^{B}_{A}R\,{}^{A}P$$

${}^{B}_{A}R$ 表示用以下方式描述帧 A 的坐标系

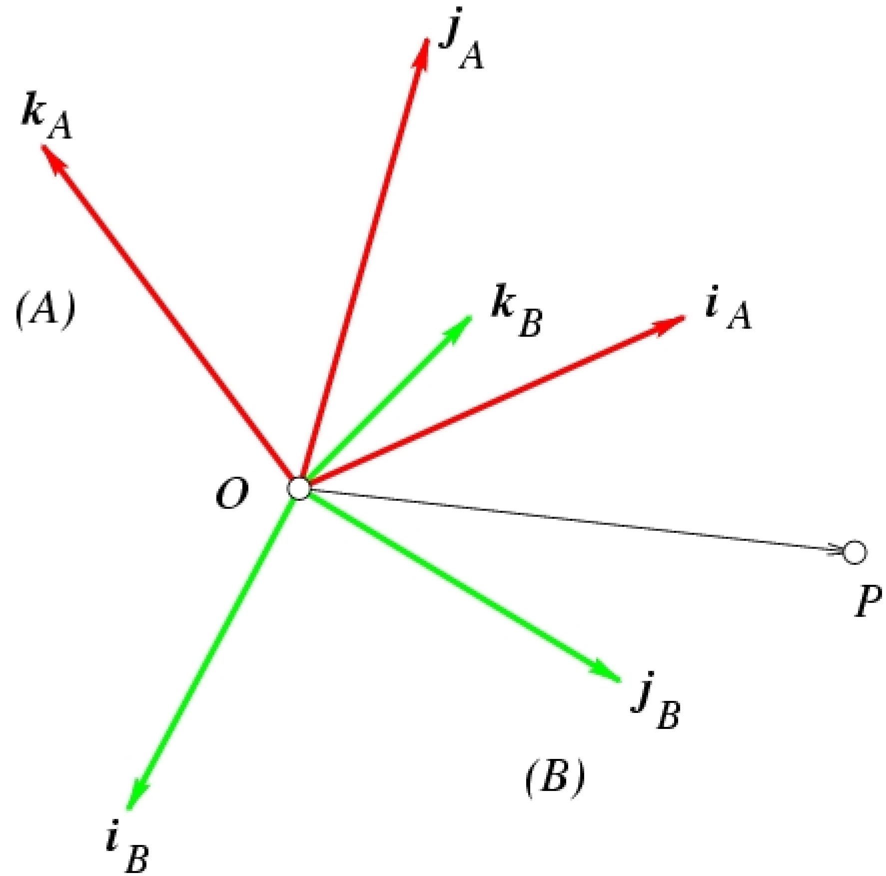
框架 B

旋转

$$ {}^{B}_{A}R = \begin{bmatrix} \mathbf{i}_A\cdot\mathbf{i}_B & \mathbf{j}_A\cdot\mathbf{i}_B & \mathbf{k}_A\cdot\mathbf{i}_B \\ \mathbf{i}_A\cdot\mathbf{j}_B & \mathbf{j}_A\cdot\mathbf{j}_B & \mathbf{k}_A\cdot\mathbf{j}_B \\ \mathbf{i}_A\cdot\mathbf{k}_B & \mathbf{j}_A\cdot\mathbf{k}_B & \mathbf{k}_A\cdot\mathbf{k}_B \end{bmatrix} $$

$$ = \begin{bmatrix} {}^{B}\mathbf{i}_A & {}^{B}\mathbf{j}_A & {}^{B}\mathbf{k}_A \end{bmatrix} $$

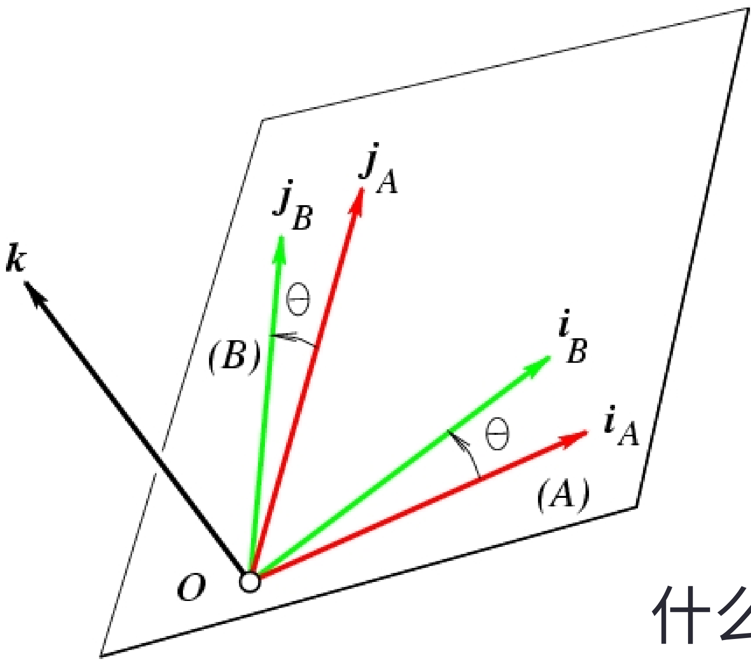
$$ = \begin{bmatrix} {}^{A}\mathbf{i}_B^{T} \\ {}^{A}\mathbf{j}_B^{T} \\ {}^{A}\mathbf{k}_B^{T} \end{bmatrix} $$

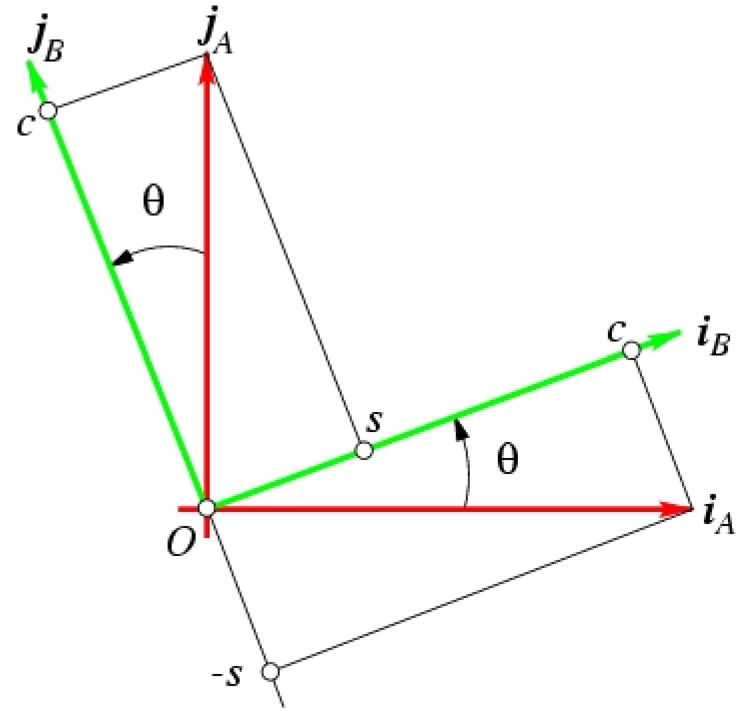


为什么?

正交矩阵

举例说明：绕 Z 轴旋转





什么是旋转矩阵？

$$R_Z(\theta) = \begin{bmatrix} \text{⊠⊠⊠}(\theta) & -\sin(\theta) & 0 \\ \text{⊠⊠⊠}(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

将 3 个组合起来，就可以任意旋转

- 欧拉角Z、X'、Z'
- 航向、俯仰滚动：世界 Z、新 X、新 Y
- 三个基本矩阵：顺序很重要，但我们不着重讨论这个问题

$$R_Z(\theta) = \begin{bmatrix} ☒☒☒(\theta) & -\sin(\theta) & 0 \\ ☒☒☒(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad R_X(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & & \end{bmatrix}$$

☒(κ)

☒

☒

0

$$\sin(\phi) \qquad\qquad \cos(\phi)$$

$$R_Y(\kappa) = \begin{bmatrix} & & \\ 0 & 1 & 0 \\ \sin(\kappa) & 0 & \cos(\kappa) \end{bmatrix}$$

同质坐标旋转

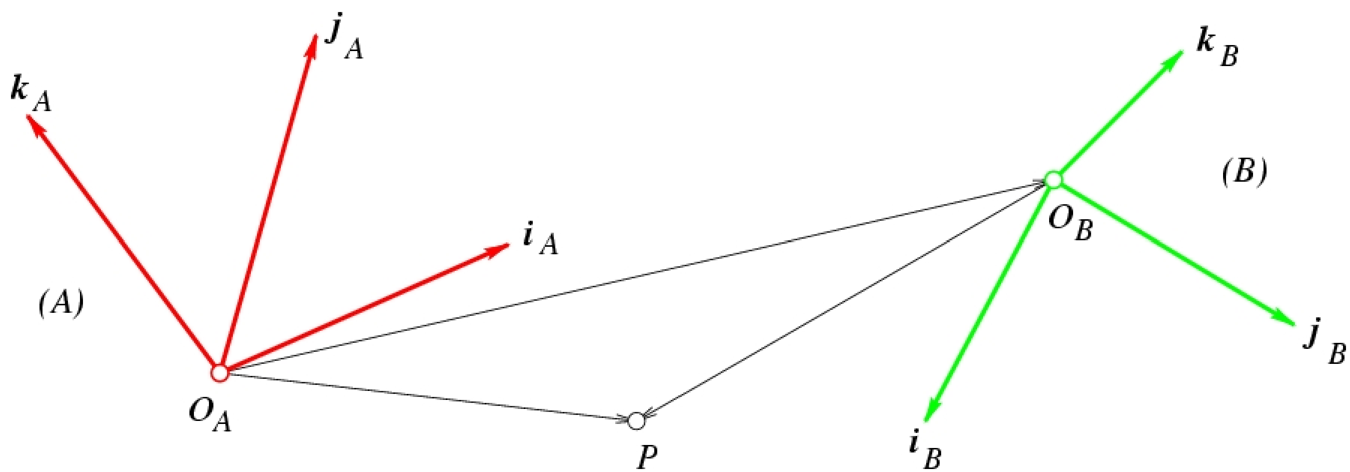
- 使用同质坐标，旋转可以表示为矩阵乘法。

$$BP = {}^{B}_{A}R\, AP$$

$$\begin{bmatrix} {}^{B}P \\ 1 \end{bmatrix} = \begin{bmatrix} {}^{B}_{A}R & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} {}^{A}P \\ 1 \end{bmatrix}$$

- 旋转不交换

刚性变换



$$^{B}P = \square^{BA}_{A}\,\square + \square^{B}_{A}$$

刚性变换（续）

- 使用同质坐标进行统一处理。

$$\begin{bmatrix} {}^{B}P \\ 1 \end{bmatrix} = \begin{bmatrix} I & {}^{B}O_{A} \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} {}^{B}_{A}R & 0 \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} {}^{A}P \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} {}^{B}_{A}R & {}^{B}O_{A} \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} {}^{A}P \\ 1 \end{bmatrix}$$

可逆!

$$\longrightarrow \quad \begin{bmatrix} {}^{B}P \\ 1 \end{bmatrix} = {}^{B}_{A}T \begin{bmatrix} {}^{A}P \\ 1 \end{bmatrix}$$

$\lfloor \quad \rfloor \qquad \lfloor \quad \rfloor$

平移和旋转

从 A 框到 B 框：

非均质（"规则"）坐标→队列

$$ {}^{B}\vec{p} = {}^{B}_{A}R \; {}^{A}\vec{p} + \boxtimes^{B}_{A} $$

3x3 旋转矩阵

基因坐标

$$ {}^{B}\vec{p} = \left({}^{B}_{A}R \quad \boxtimes \quad \vec{at} \right) \begin{bmatrix} x \\ \vdots \\ z \end{bmatrix} $$

同质

坐标可以让我们写

出坐标

转换为

$$\begin{pmatrix} 0 & 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} 1 \end{bmatrix}$$

单一矩阵!

从世界到相机

从世界画面旋转到摄像机画面

从世界画面到摄像机画面的转换

$$ {}^{C}\vec{p} = {}^{C}_{W}R\,{}^{W}\vec{p} + {}^{C}\vec{t}_{W} $$

指向相框

点在世界格局

非均质坐标

$$ \begin{pmatrix} {}^{C}\vec{p} \\ 1 \end{pmatrix} = \begin{pmatrix} {}^{C}_{W}R & {}^{C}_{W}\vec{t} \\ 0\quad 0\quad 0 & 1 \end{pmatrix} \begin{pmatrix} {}^{W}\vec{p} \\ 1 \end{pmatrix} $$

均质坐标

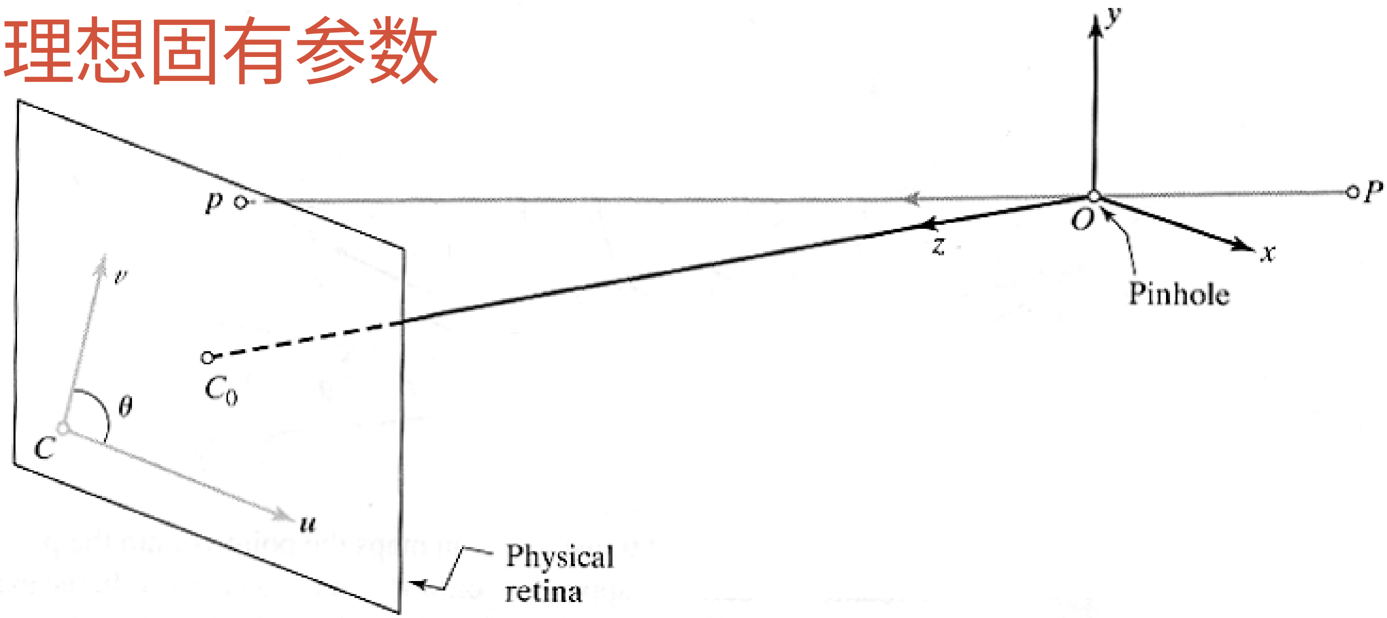
从世界到相机是

外参数矩阵（4x4）

(如果用于下一步的投影，有时是 3x4，不用担心反转）。

现在，从 3D 相机到图像...

将相机 3D (x,y,z) 转换为 2D (u,v) 或 (x',y')：理想固有参数



$$u = f\frac{x}{z}$$

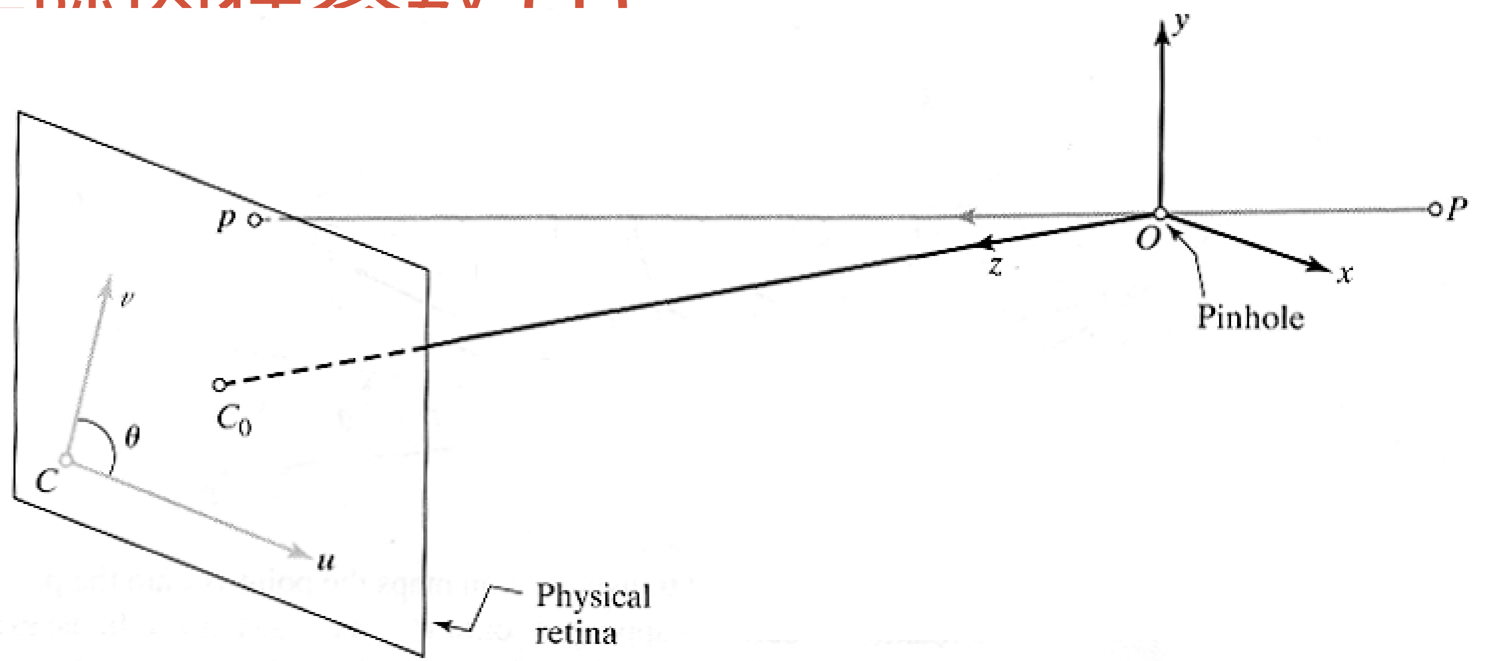
理想视角投影

$$v = f^{y}$$

z

—

实际内在参数 (1)



但 "像素 "是以某种任意的空间单位表示的

z

u

$=$

α

x

$-$

z

v

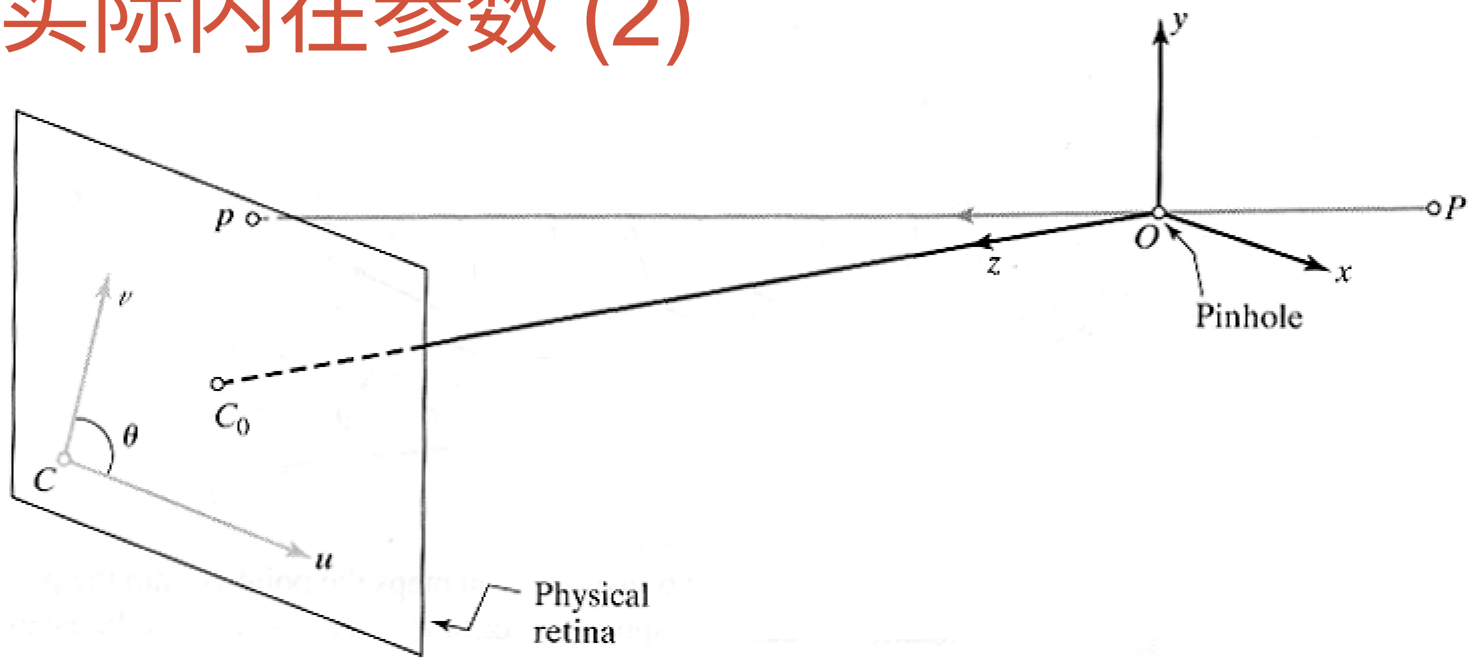
$-$

$=$

α

y

实际内在参数 (2)



也许像素是

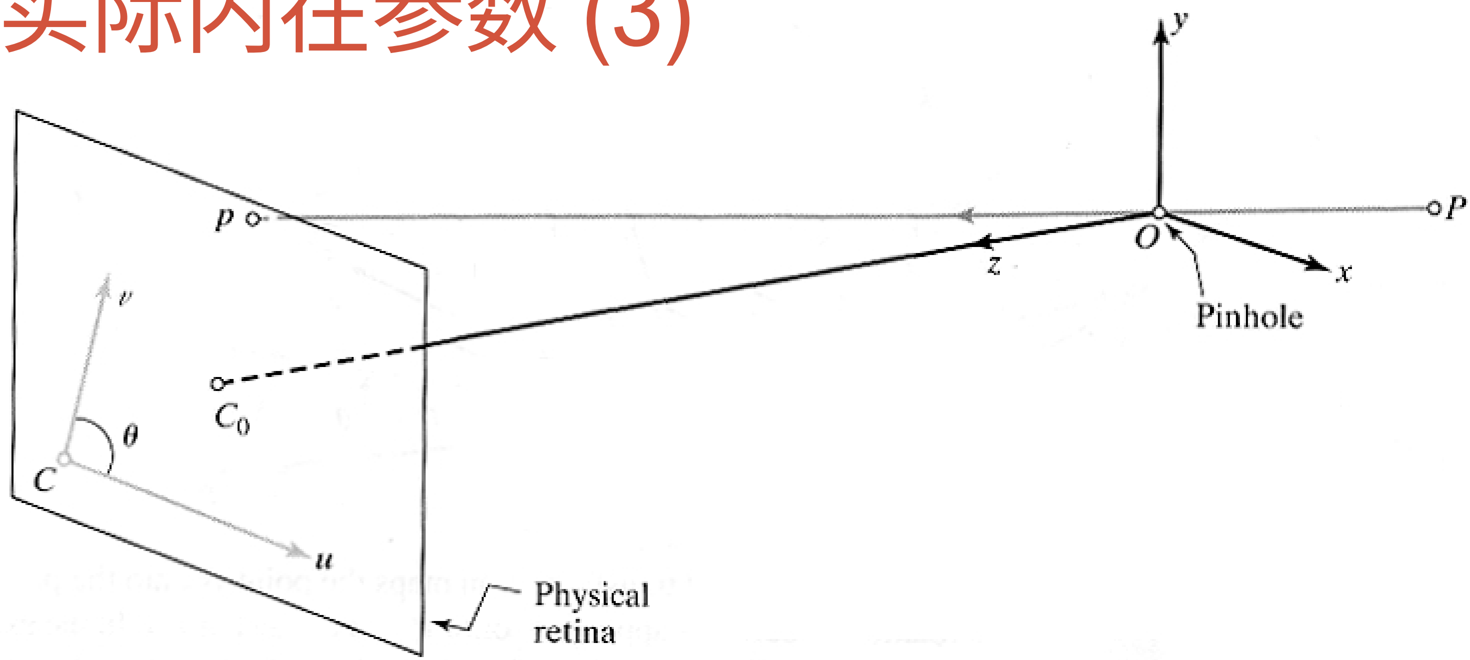
不方正

$$u = \alpha \frac{x}{z}$$

$$v = \beta \frac{y}{z}$$

Z

实际内在参数 (3)



我们不知道

摄像机像素坐

标的原点

$$u = \alpha \quad x \quad +$$

$$\beta$$

$$u$$

$$v =$$

0

z

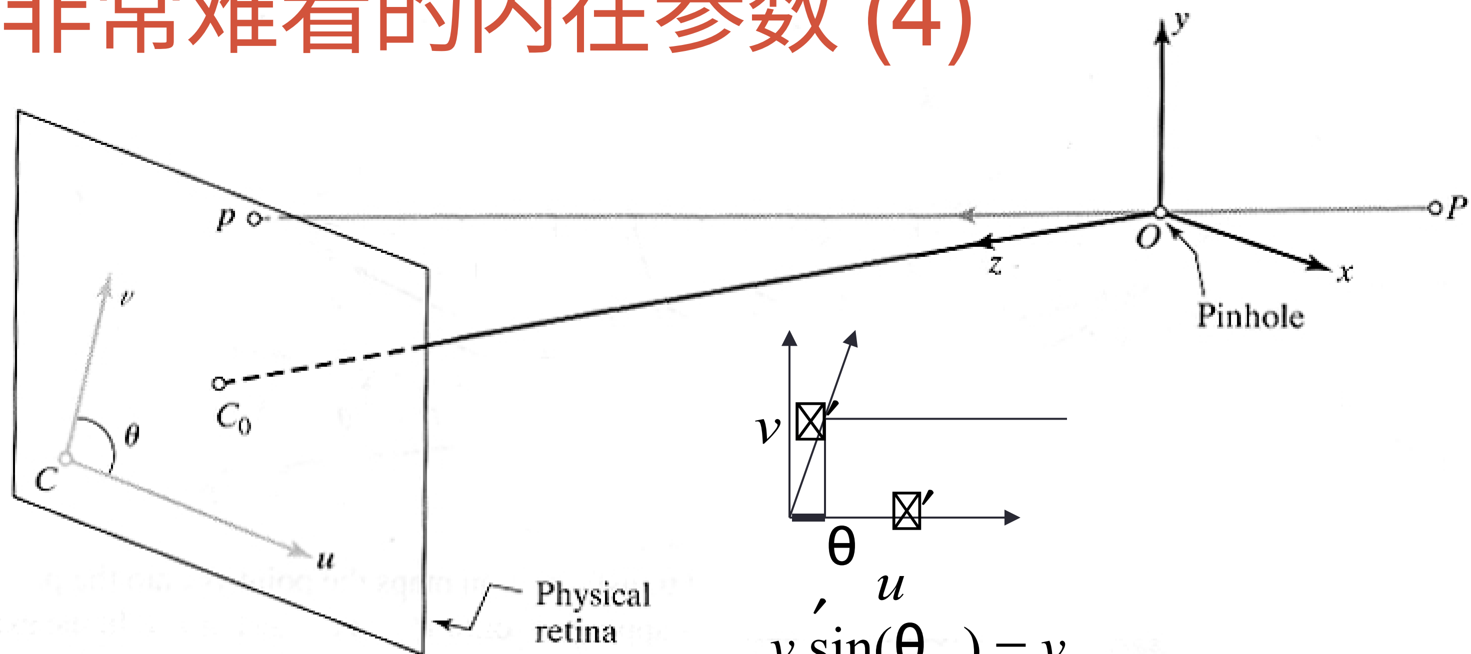
y

$+ v$ $—$

0

z

非常难看的内在参数 (4)



$$v'\sin(\theta) = v$$

$$u' = u - \cos(\theta)v' = u - \cot(\theta)v$$

摄像机之间可能

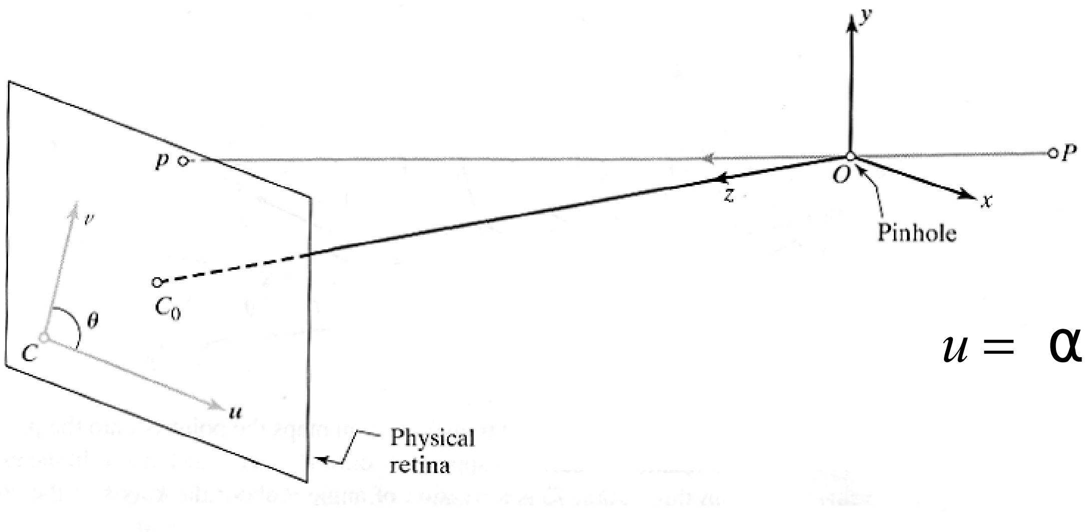
存在偏差

像素轴

$$u = \alpha\frac{x}{z} - \alpha\cot(\theta)\frac{y}{z} + u_0$$

$$v = \frac{\beta}{\sin(\theta)} \frac{y}{z} + v_0$$

内在参数，同质坐标



注意除以 z

$$u = \alpha\frac{x}{z} - \alpha\cot(\theta)\frac{y}{z} + u_0$$

$$v = \frac{\beta}{\sin(\theta)}\frac{y}{z} + v_0$$

使用同质坐标，我们可以将其写成

$$\begin{pmatrix} z*u \\ z*v \end{pmatrix} = \begin{pmatrix} \alpha & -\alpha\cot(\theta) & u_0 & 0 \\ 0 & \frac{\beta}{\sin(\theta)} & v_0 & 0 \end{pmatrix}\begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

在同构

z

$\rightarrow$

$0 \quad 0 \quad 10$

1

在基于摄像头的

3D 坐标中

像素

$$p' = \mathbf{K} \; {}^{C}\vec{p}$$

更友好、更温和的固有技术

- 可以使用更简单的符号来表示内在字符--最后一列为零：

$$K = \begin{bmatrix} f & s & c_x \\ 0 & & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

s - 偏斜

a - 长宽比（5 DOF）

- 如果像素为正方形，无偏斜，且光学中心位于中心（假设原点位于中间）：

$$K = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

在这种情况下，只有一个DOF，即焦距f

在同质坐标中结合外在和内在校准参数

像素

3D 相机坐标

世界 3D 坐标

$$\vec{p}\,' = \mathrm{K}\;{}^{C}\vec{p} \qquad \text{内在}$$

$${}^{C}\vec{p} = \begin{pmatrix} {}^{C}_{W}R & {}^{C}_{W}t \\ 0\;0\;0 & 1 \end{pmatrix} {}^{W}\vec{p} \qquad \text{外在}$$

$$\vec{p}\,' = K \underbrace{\begin{pmatrix} {}^{C}_{W}R & {}^{C}_{W}t \\ 0\;0\;0 & 1 \end{pmatrix}}\, {}^{W}\vec{p}$$

$$\vec{p}' = M \quad \vec{p^{W}}$$ (如果 K 是 3x4)

相同等式的其他写法

像素坐标

世界坐标

$$\vec{p}\,' = M \; {}^{W}\vec{p}$$

从均质坐标转换回来的结果是

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \simeq \begin{pmatrix} s*u \\ s*v \\ s \end{pmatrix} = \begin{pmatrix} \cdot & m_1^T & \cdot & \cdot \\ \cdot & m_2^T & \cdot & \cdot \\ \cdot & m_3^T & \cdot & \cdot \end{pmatrix} \begin{pmatrix} {}^{W}p_x \\ {}^{W}p_y \\ {}^{W}p_z \\ 1 \end{pmatrix}$$

$$u = \frac{m_1 \cdot \vec{P}}{m_3 \cdot \vec{P}}$$

$$m \cdot P$$

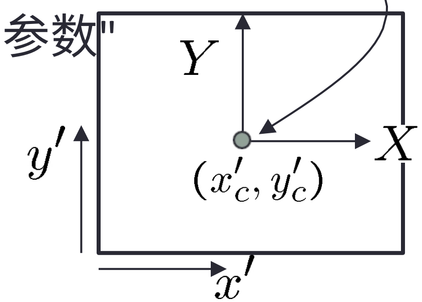
射影

$$v = \frac{2}{m3 \cdot P}$$

终于找到了相机参数

摄像机（及其矩阵）**M**（或 H）由几个参数描述

- 光学中心从世界坐标原点出发的平移量 **T**
- 图像平面的旋转 **R**
- 焦距 f、原理点 (x'_c、y'_c)、像素大小 (s_x、s_y)
- **蓝色**参数称为 "**外在**参数"，红色参数称为 "**内在**参数"



投影方程

$$\mathbf{x} \cong \begin{bmatrix} s x \\ s y \\ s \end{bmatrix} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix}\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \mathbf{M}\mathbf{X}$$

- 投影矩阵模拟所有参数的累积效应
- 分解为一系列操作，非常有用

身份矩阵

$$\mathbf{M} = \begin{bmatrix} f/s_x & 0 & x'_c \\ 0 & af/s_y & y'_c \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} \mathbf{R}_{3x3} & \mathbf{0}_{3x1} \\ \mathbf{0}_{1x3} & 1 \end{bmatrix}\begin{bmatrix} \mathbf{I}_{3x3} & \mathbf{T}_{3x1} \\ \mathbf{0}_{1x3} & 1 \end{bmatrix}$$

DoFs：

5+0+3+3 = 11

$\| \lfloor 0 \quad 01 \rfloor \| \lfloor 0 \quad 0 \quad 1 \quad \rfloor \lfloor_{1x3} \quad \rfloor$

$0 \rfloor \rfloor \lfloor_{1x3}$

本征函数 预测 旋转 平移

- 这些参数的定义并**不**完全标准化

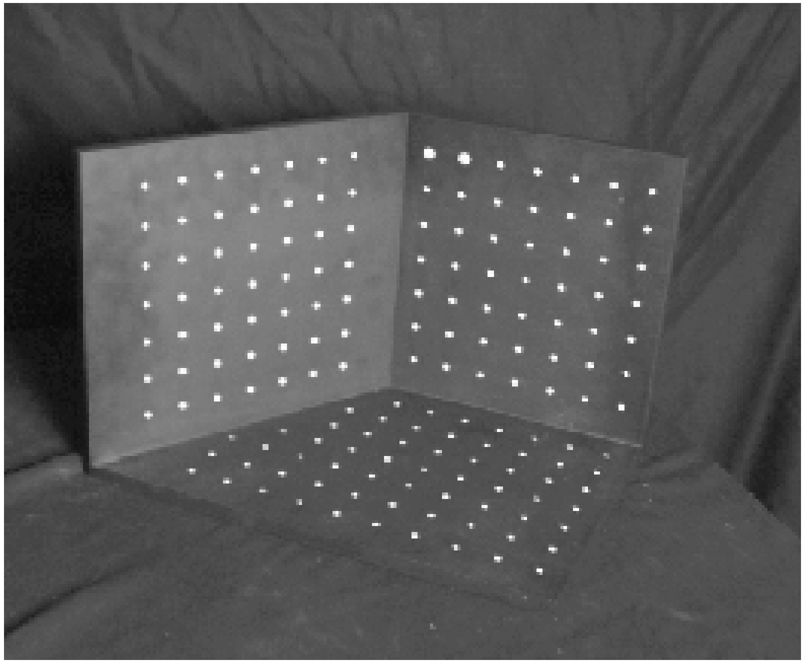
  特别是内在因素--因书而异

校准

- 如何确定 **M**（或 H）？

使用参照物进行校准

- 在场景中放置已知物体
  - 识别图像与场景的对应关系
  - 计算从场景到图像的映射

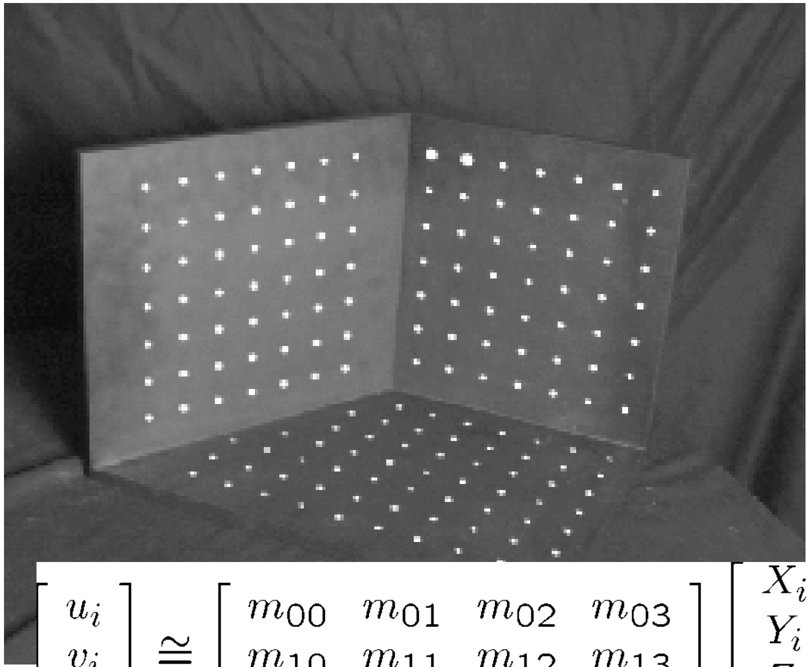


问题

- 必须对几何知识了如指掌
- 必须知道 3D->2D 对应关系

估计投影矩阵

- 在场景中放置已知物体
  - 识别图像与场景的对应关系
  - 计算从场景到图像的映射



$$\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} \cong \begin{bmatrix} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & m_{23} \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix}$$

重切分 - 根据已知 3D 点估算摄像机矩阵

- 投影相机矩阵

$$p = K\begin{bmatrix} R & t \end{bmatrix} P = MP$$

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \cong \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

- 只能达到一个刻度，

因此有 11 个 DOF。

直接线性校准 - 均质

$$\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} \cong \begin{bmatrix} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & m_{23} \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix}$$

$$\begin{aligned} u_i &= \frac{m_{00}X_i + m_{01}Y_i + m_{02}Z_i + m_{03}}{m_{20}X_i + m_{21}Y_i + m_{22}Z_i + m_{23}} \\ v_i &= \frac{m_{10}X_i + m_{11}Y_i + m_{12}Z_i + m_{13}}{m_{20}X_i + m_{21}Y_i + m_{22}Z_i + m_{23}} \end{aligned}$$

$$u_i(m_{20}X_i + m_{21}Y_i + m_{22}Z_i + m_{23}) = m_{00}X_i + m_{01}Y_i + m_{02}Z_i + m_{03}$$
$$v_i(m_{20}X_i + m_{21}Y_i + m_{22}Z_i + m_{23}) = m_{10}X_i + m_{11}Y_i + m_{12}Z_i + m_{13}$$

每个点有一对

程

$$\begin{bmatrix} X_i & Y_i & Z_i & 1 & 0 & 0 & 0 & 0 & -u_iX_i & -u_iY_i & -u_iZ_i & -u_i \\ 0 & 0 & 0 & 0 & X_i & Y_i & Z_i & 1 & -v_iX_i & -v_iY_i & -v_iZ_i & -v_i \end{bmatrix} \begin{bmatrix} m_{00} \\ m_{01} \\ m_{02} \\ m_{03} \\ m_{10} \\ m_{11} \\ m_{12} \\ m_{13} \\ m_{20} \\ m_{21} \\ m_{22} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

直接线性校准 - 均质

$$
\begin{bmatrix}
X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -u_1X_1 & -u_1Y_1 & -u_1Z_1 & -u_1 \\
0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -v_1X_1 & -v_1Y_1 & -v_1Z_1 & -v_1 \\
 & & & & & & \vdots & & & & & \\
X_n & Y_n & Z_n & 1 & 0 & 0 & 0 & 0 & -u_nX_n & -u_nY_n & -u_nZ_n & -u_n \\
0 & 0 & 0 & 0 & X_n & Y_n & Z_n & 1 & -v_nX_n & -v_nY_n & -v_nZ_n & -v_n
\end{bmatrix}
\begin{bmatrix} m_{00} \\ m_{10} \\ m \\ m02 \\ m03 \\ \boxtimes 10 \\ m11 \\ \boxtimes 2 \\ m13 \\ m20 \\ m21 \\ m22 \\ 23 \end{bmatrix}
=
\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}
$$

A $\quad$ **m** $\quad$ **0**

2n × 12 $\quad$ **12** $\quad$ **2n**

这是一组同质方程。

过度约束时，定义最小二乘法问题

– 减少 $\|\mathbf{Am}\|$

- 由于 **m** 的定义只限于比例范围，因此求解单位向量 **m***
- 解：**m*** = **ATA** 特征值*最小的*特征向量
- 6 个或更多点

SVD (奇异值分解) 技巧...

在 ||x||=1 的条件下，求使 ||Ax|| 最小的 **x**。

设 **A** = $UDVT$ (奇异值分解，**D** 为对角线）、

U 和 **V** 正交）因此

，最小化 ||**UDVTx**||

但是，$||UDVTx|| = ||$**DVTx**||，||x|| = ||**VTx**||，因

此，在 $||VTx|| = 1$ 的条件下，最小化 ||**DVTx**||

让 **y** = VTx：在 ||**y**||=**1** 的条件下，最小化

$\|\mathbf{Dy}\|$。

但是 **D** 是对角线，其值是递减的。因此，当

$$\mathbf{y} = (0,0,0...,0,1)^T$$

因此，$\mathbf{x} = \mathbf{Vy}$ 是 V 的最后一列。 [正：$V^T = V^{-1}$]

而且，**A** 的奇异值是 **ATA** 特征值的平方根，**V** 的列是特征向量。(演示一下? ）

直接线性校准 - 不均匀

- 另一种方法：1 个在右下角，11 个在左上角

$$\begin{bmatrix} \boxtimes \\ v \;\square \\ 1 \end{bmatrix} \begin{bmatrix} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

- 现在是 "常规 "最小二乘法，因为存在一个非变量方程中的项：

$$u_i = \frac{m_{00}X_i + m_{01}Y_i + m_{02}Z_i + m_{03}}{m_{20}X_i + m_{21}Y_i + m_{22}Z_i + 1}$$

如果 m_{23} 真的

$$m_{10}X_i + m_{11}Y_i + m_{12}Z_i + m_{13}$$

为零，那就危
险了！

直接线性校准（转换）

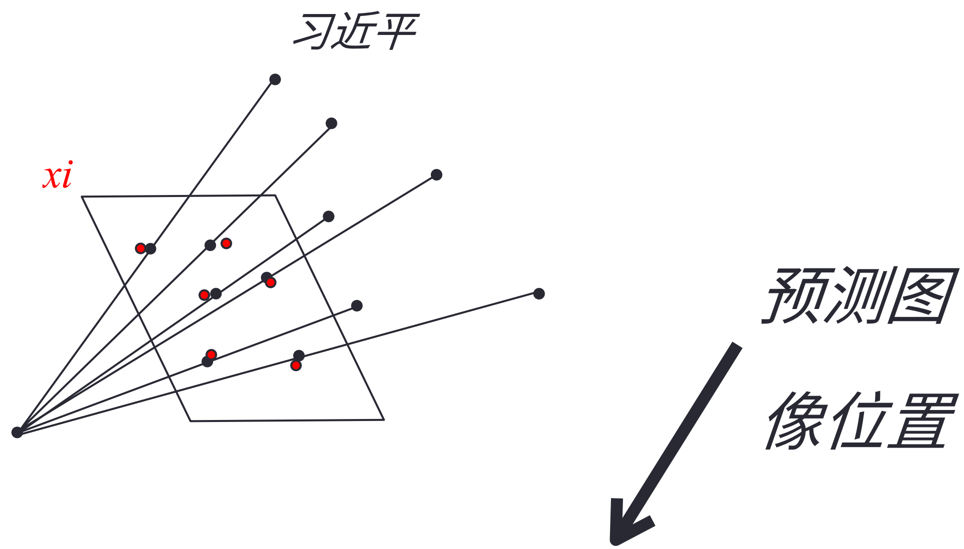
- 优势
  - 提出问题和解决问题都非常简单。例如，可以在一个问题集上完成
  - 这些方法被称为 "代数误差 "最小化。
- 缺点
  - 不会直接告诉你相机参数（稍后详述）
  - 不模拟径向变形
  - 难以施加限制（如已知焦距）

- 没有最小化正确的误差函数

因此，*非线性方法*更受青睐

- 定义三维投影点与图像位置之间的误差函数 E
  - E 是本征、外征、径向畸变的非线性函数
- 使用非线性优化技术使 E 最小化
  - 例如，牛顿法的变体（例如，莱文伯格-马夸特法）

几何误差



使 E 最小化 $= \sum_i d\big($⊠⊠′、⊠⊠⊠′$\big)$

分
钟
M

$$\sum_i d\left(\;',\; {}_{MXi}\right)$$

"黄金标准 "算法 *(哈特利和齐瑟曼)*

目标

给定 $n \geq 6$ 个三维到二维点的对应关系 $\{X_i \leftrightarrow x_i'\}$，确定 **M 的** "最大似然估计值"。

算法

(i) 线性解决方案：

(a)（可选）标准化： $\tilde{X}_i = UX_i$ $\tilde{x}_i = Tx_i$

(b) 直接线性变换的几何最小化

误差：以线性估计为起点，将误差最小化

几何误差：

$$\underset{\mathbf{M}}{分钟} \sum_i d(□□', MXi)$$

(ii) 去规范化：$\mathbf{M} = \mathbf{T}^{-1} \tilde{M} \mathbf{U}$

从 P 矩阵找到 3D 相机中心

- 符号略有变化。设 $\mathbf{M} = [\mathbf{Q} \mid \mathbf{b}]$ (3x4) - $\mathbf{b}$ 是 $\mathbf{M}$ 的最后一列
- 投影矩阵的空位摄像机。求 **C**，使得

$$\mathbf{MC} = \mathbf{0}$$

- 证明设 **X** 位于任意点 **P** 和 **C** 之间的某处

$$\mathbf{X} = \lambda\mathbf{P} + (1 \quad \lambda)\mathbf{C}$$

⊠

$$\mathbf{x} = \mathbf{MX} = \lambda\mathbf{MP} + (1 \quad \lambda)\mathbf{MC}$$

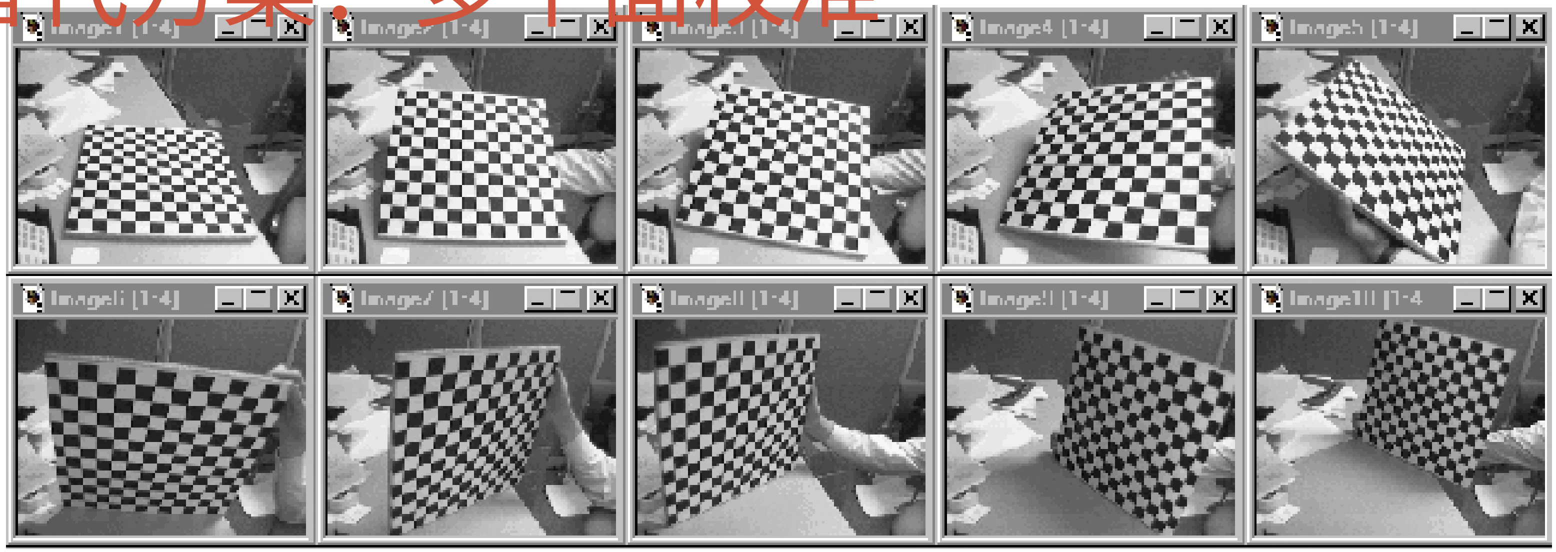
⊠

- 对于所有 P，PC 上的所有点都投影在 P 的图像上、

- 因此，C 摄像机中心必须位于无效空间内
- 还可以通过以下方式找到

$$\mathbf{C} = \begin{pmatrix} -\mathbf{Q}^{-1} \boxtimes \\ 1 \end{pmatrix}$$

替代方案：多平面校准



图片由英特尔公司 Jean-Yves Bouguet 提供。

优势

- 只需要一架飞机
- 不必知道位置/方向
- 网上有很好的代码！

- 英特尔 OpenCV 库：http://www.intel.com/research/mrl/research/opencv/
- Matlab 版本由 Jean-Yves Bouget 制作：
  http://www.vision.caltech.edu/bouguetj/calib_doc/index.html
- Zhengyou Zhang 的网站：http://research.microsoft.com/~zhang/Calib/