相机模型与投影变换

章国锋

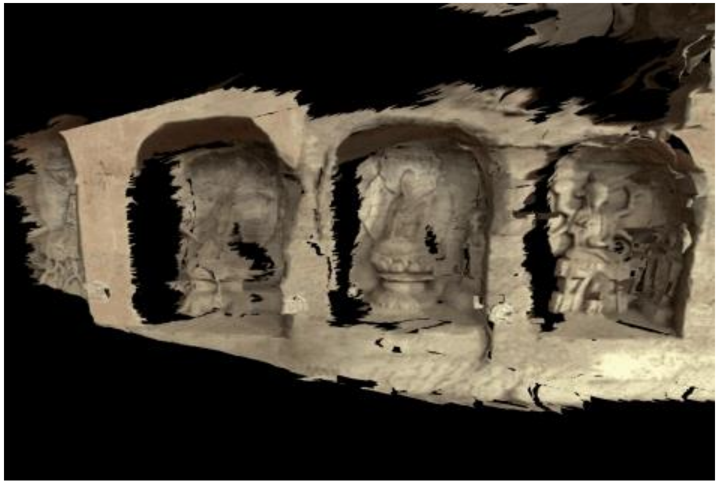
浙江大学**CAD&CG**国家重点实验室

视频场景重建的流程

运动恢复结构

深度恢复

三维重建



齐次坐标

在原有的坐标上增加一个维度：

$$\begin{bmatrix} x \\ y \end{bmatrix} \rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \qquad \begin{bmatrix} x \\ y \\ z \end{bmatrix} \rightarrow \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

新增的维度并不会增加自由度：

$$(x, y, z, w) \quad w \neq 0 \rightarrow (x/w, y/w, z/w)$$

齐次坐标

使用2D坐标完成平移：

$$x' \rightarrow \begin{bmatrix} u' \\ v' \end{bmatrix} = \begin{bmatrix} u + t_u \\ v + t_v \end{bmatrix} = x + t$$

使用齐次坐标完成平移：

$$x' \rightarrow \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_u \\ 0 & 1 & t_v \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = Tx$$

齐次坐标

使用齐次坐标判断点是否在线上

$$l = \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix}$$

$$x^T l = \begin{bmatrix} u & v & 1 \end{bmatrix} \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix} = 0$$

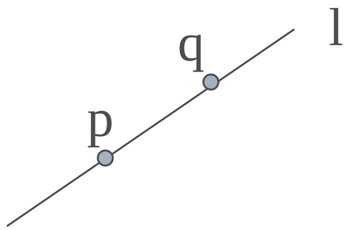
使用齐次坐标判断点是否在平面上

$$\pi = \begin{bmatrix} n_1 \\ n_2 \\ n_3 \\ d \end{bmatrix}$$

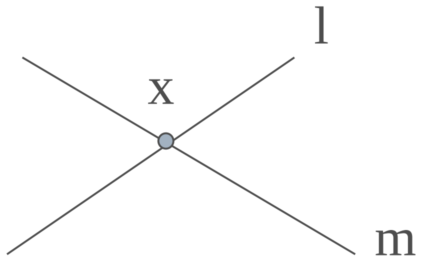
$$x^T \pi = \begin{bmatrix} x & y & z & 1 \end{bmatrix} \begin{bmatrix} n_1 \\ n_2 \\ n_3 \\ d \end{bmatrix} = 0$$

齐次坐标

- 两个点定义一条直线: $\mathbf{l} = \mathbf{p} \times \mathbf{q}$



- 两条直线定义一个点: $\mathbf{x} = \mathbf{l} \times \mathbf{m}$



齐次坐标

使用齐次坐标完成平移和放缩：

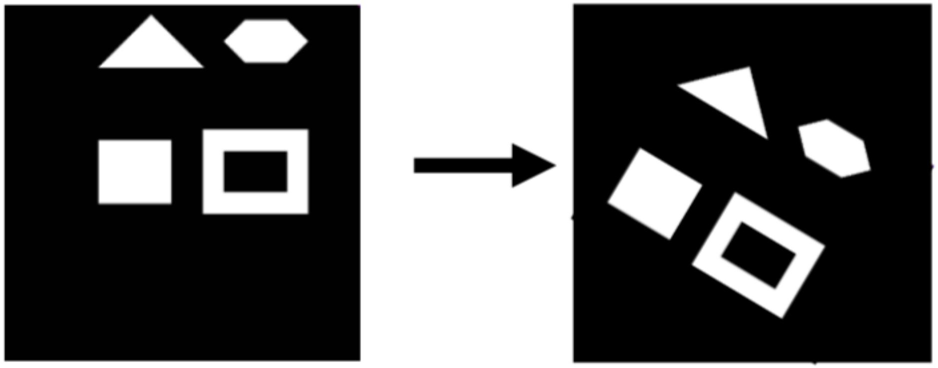
$$x' \rightarrow \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} s_u & 0 & s_u t_u \\ 0 & s_v & sv_t v \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} s_u & 0 & 0 \\ 0 & s_v & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & t_u \\ 0 & 1 & t_v \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = STx$$

使用齐次坐标完成旋转和平移：

$$x' \rightarrow \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta & t_u \\ sin\theta & cos\theta & t_v \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_u \\ 0 & 1 & t_v \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} cos\theta & -sin\theta & 0 \\ sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = TRx$$

等距变换

2D的等距变换具有三自由度，这种变换是保距离的

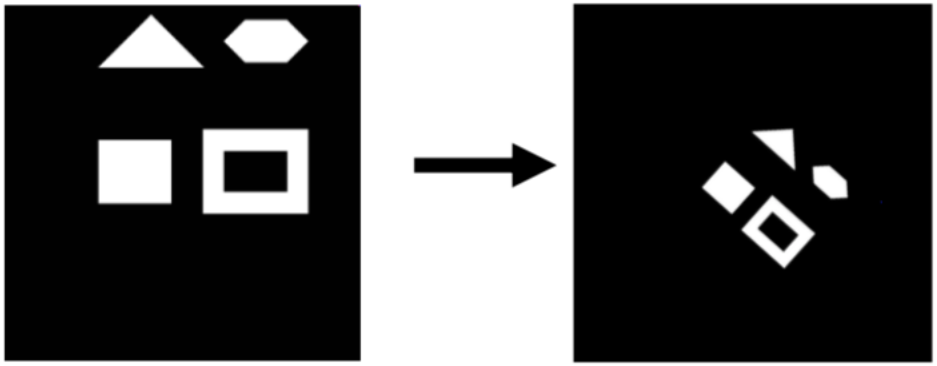


$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

R为旋转矩阵，t为平移向量

相似变换

2D的相似变换具有四自由度，这种变换是保角度的

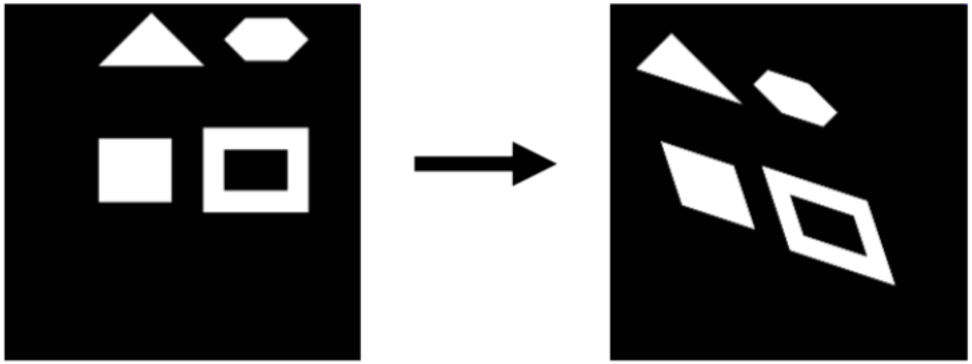


$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} sR & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

s为相似变换因子

仿射变换

2D的仿射变换具有六自由度，这种变化是保平行的

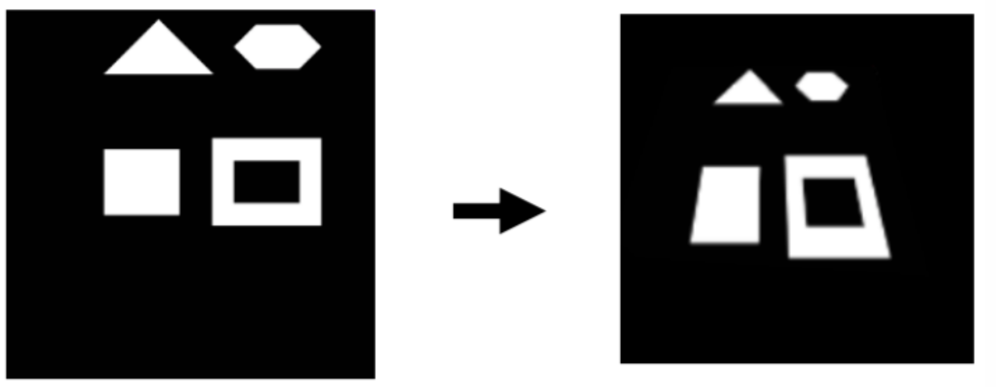


$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} A & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = R(\theta)R(-\phi)SR(\phi) \qquad S = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}$$

射影变换

2D的射影变换具有八自由度，这种变化是保同线性的



$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} A & t \\ v & b \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \qquad H = \begin{bmatrix} A & t \\ v & b \end{bmatrix}$$

求解射影变换

$$H = \begin{bmatrix} A & t \\ v & b \end{bmatrix} = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \qquad x' = Hx$$

$$u_2 = \frac{h_1 u + h_2 v + h_3}{h_7 u + h_8 v + h_9}$$

$$v_2 = \frac{h_1 u + h_2 v + h_3}{h_4 u + h_5 v + h_6}$$

展开两个等式中得到：

$$h_1 u_1 + h_2 u_2 + h_3 - h_7 u_1 u_2 - h_8 v_1 u_2 - h_9 u_2 = 0$$

$$h_4 u_1 + h_5 u_2 + h_6 - h_7 u_1 v_2 - h_8 v_1 v_2 - h_9 v_2 = 0$$

求解射影变换

$$h_1u_1 + h_2u_2 + h_3 - h_7u_1u_2 - h_8v_1u_2 - h_9u_2 = 0$$

$$h_4u_1 + h_5u_2 + h_6 - h_7u_1v_2 - h_8v_1v_2 - h_9v_2 = 0 \tag{1}$$

我们使用下述表示

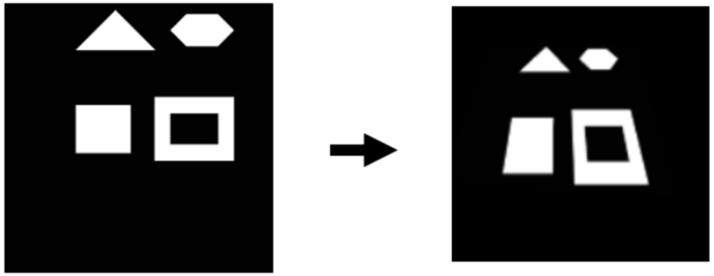
$$A_i = \begin{pmatrix} u_1 & v_1 & 1 & 0 & 0 & 0 & -u_1u_2 & -v_1u_2 & -u_2 \\ 0 & 0 & 0 & u_2 & v_2 & 1 & -u_1v_2 & -v_1v_2 & -v_2 \end{pmatrix}$$

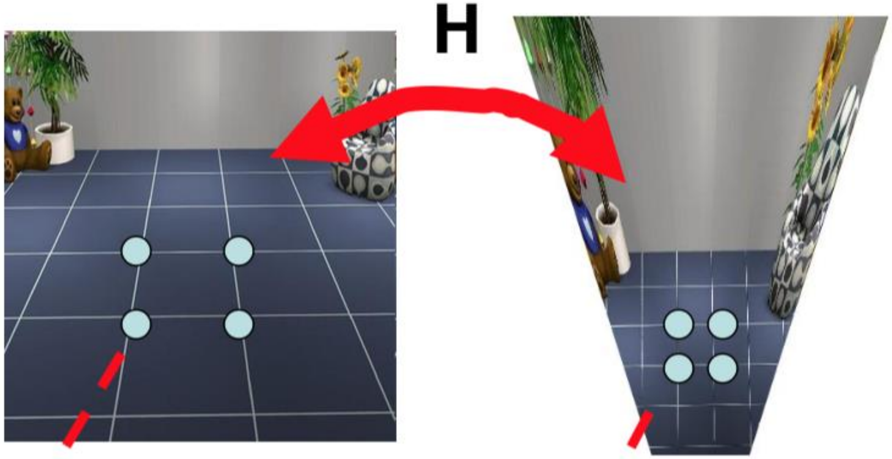
$$\mathbf{h}^{\cdot} = \begin{pmatrix} h_1 & h_2 & h_3 & h_4 & h_5 & h_6 & h_7 & h_8 & h_9 \end{pmatrix}$$

则等式(1)可以改写成如下形式：

$$A_i\mathbf{h} = 0$$

求解射影变换



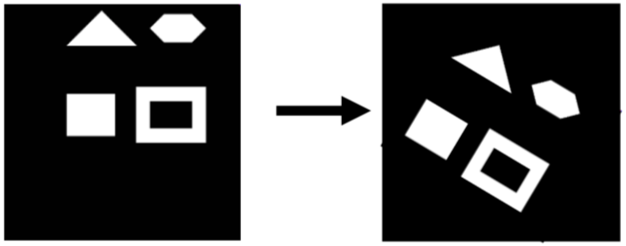


$$A_i \mathbf{h} = 0$$

x

x′

2D齐次坐标的线性变换



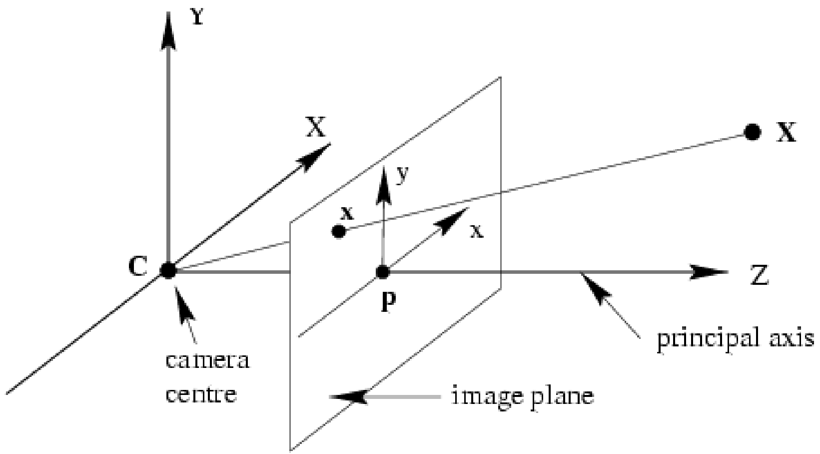
$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

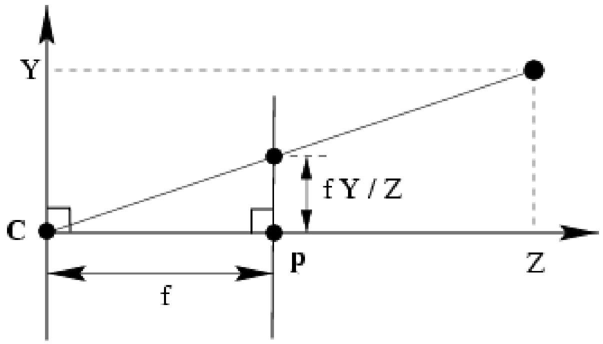
$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} sR & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} A & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} A & t \\ v & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

针孔相机模型





投影方程：

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

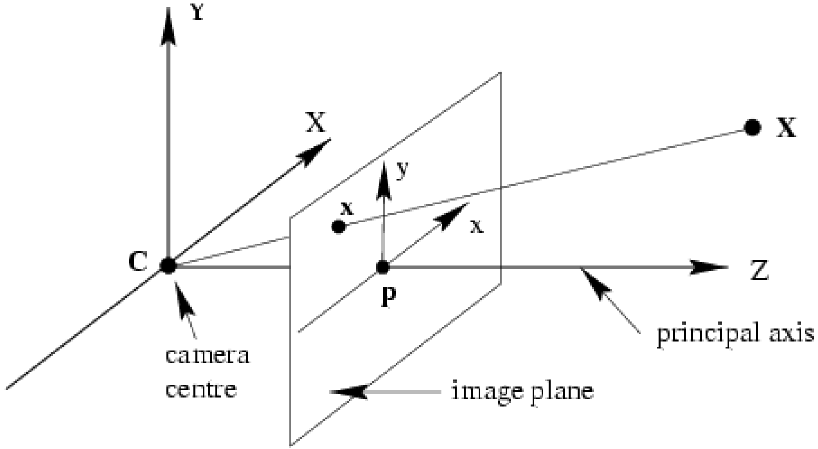
$$y = f\frac{Y}{Z}$$

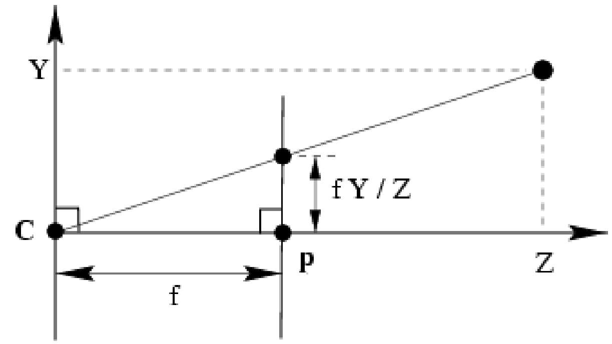
齐次坐标表示：

$$\begin{pmatrix} x \\ y \\ f \end{pmatrix} \sim \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Richard Hartley and Andrew Zisserman."Multiple View Geometry in Computer Vision". Cambridge University Press, Second Edition 2004.

求解射影变换

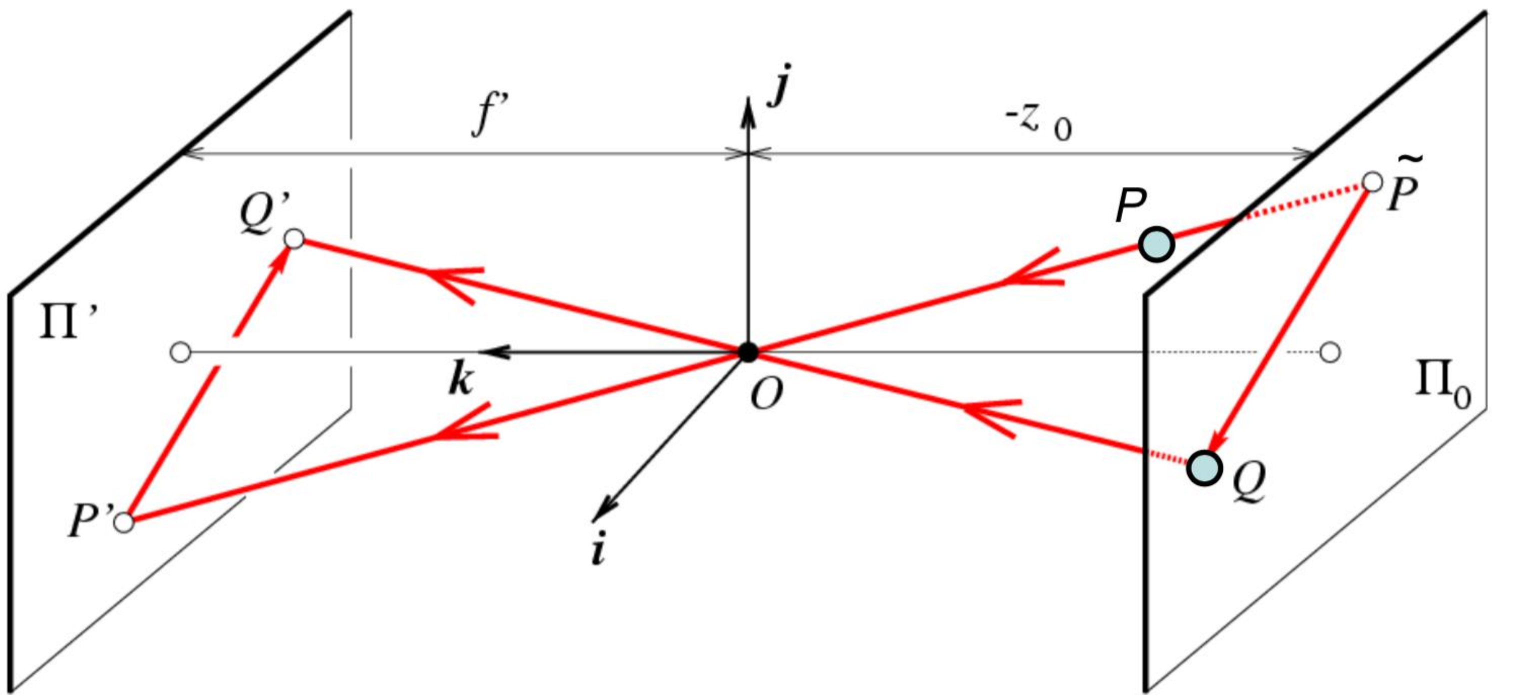




$$\begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \underbrace{\begin{bmatrix} f & & \\ & f & \\ & & 1 \end{bmatrix}}_{\mathrm{K}} \underbrace{\begin{bmatrix} 1 & & & 0 \\ & 1 & & 0 \\ & & 1 & 0 \end{bmatrix}}_{[\mathrm{R}|\mathrm{t}]} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Richard Hartley and Andrew Zisserman."Multiple View Geometry in Computer Vision". Cambridge University Press, Second Edition 2004.

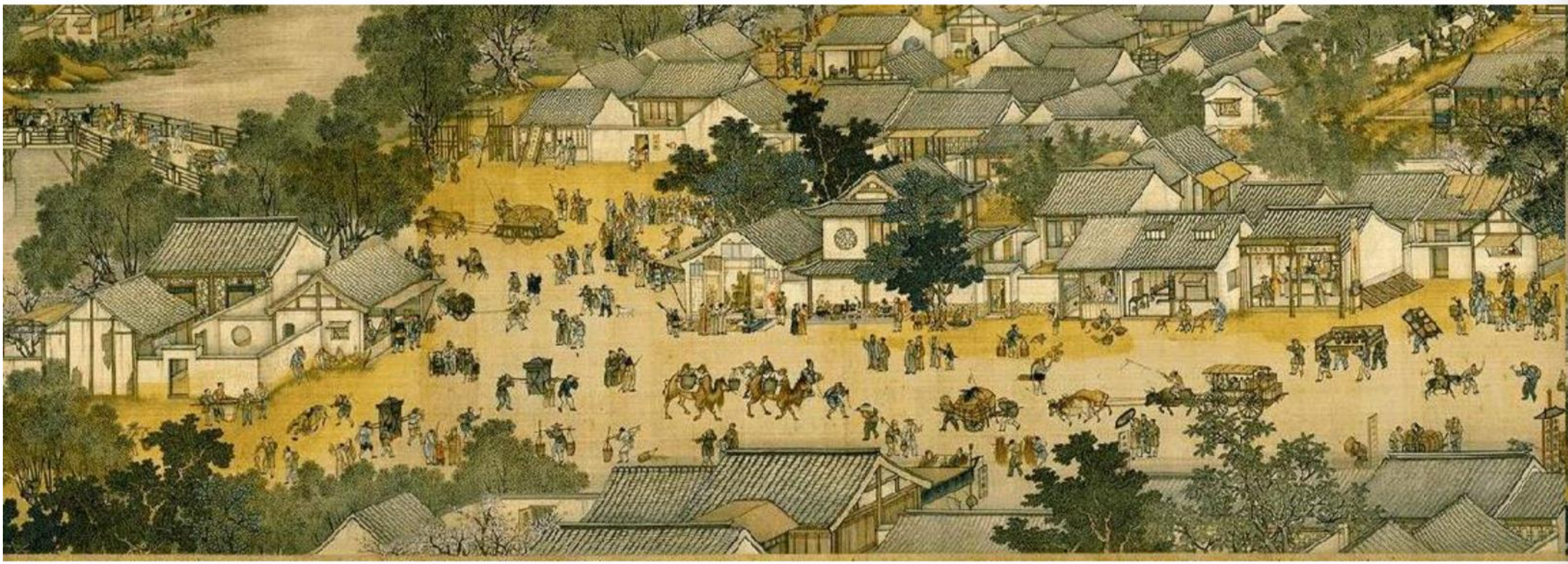
弱透视投影



$$(u, v) = (-f\frac{x}{z}, -f\frac{y}{z}) \rightarrow (u, v) = (-mx, -my)$$

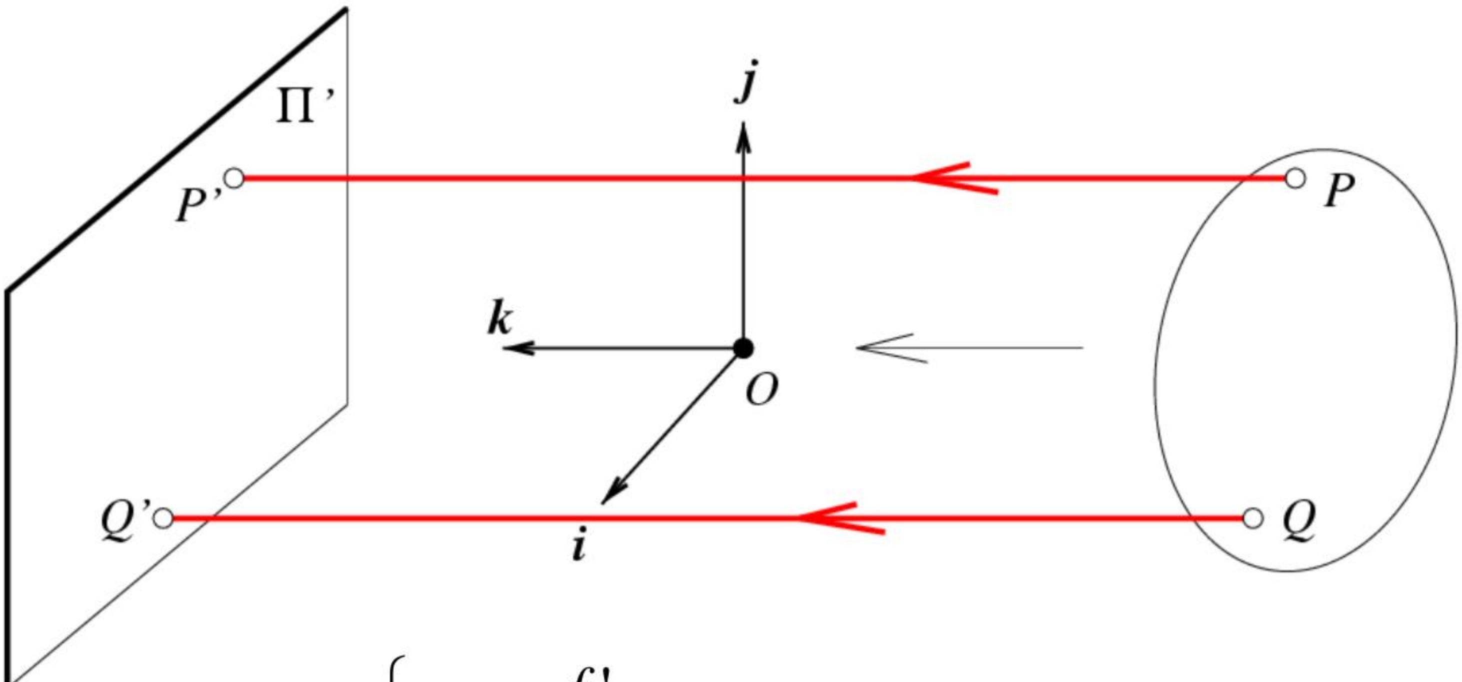
Richard Hartley and Andrew Zisserman."Multiple View Geometry in Computer Vision". Cambridge University Press, Second Edition 2004.

弱透视投影



清明上河图节选

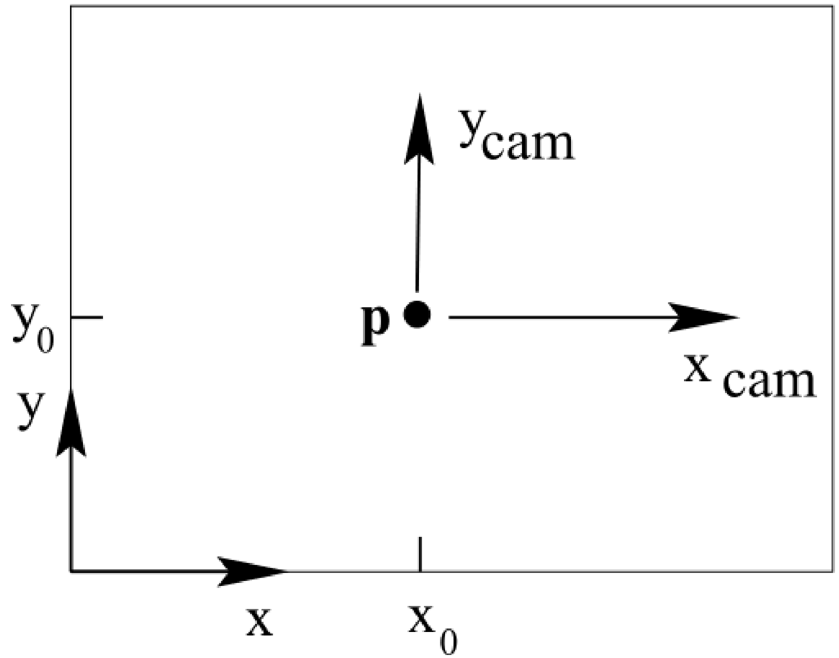
正交投影



$$\begin{cases} x' = -\dfrac{f'}{z}x \\ y' = -\dfrac{f'}{z}y \end{cases} \rightarrow \begin{cases} x' = -x \\ y' = -y \end{cases}$$

Richard Hartley and Andrew Zisserman."Multiple View Geometry in Computer Vision". Cambridge University Press, Second Edition 2004.

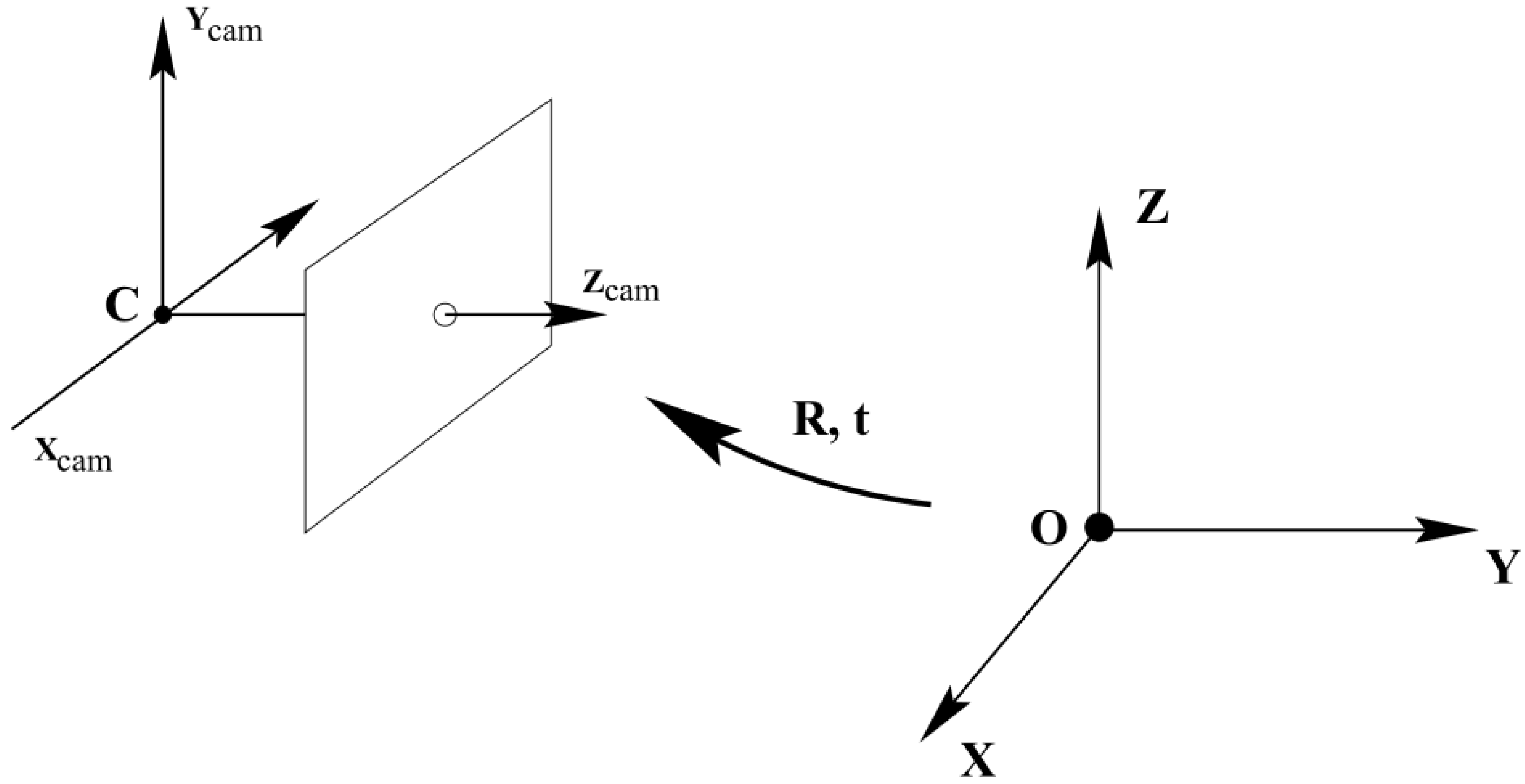
主点的偏移



$$\begin{pmatrix} fX/Z + x_0 \\ fY/Z + y_0 \\ 1 \end{pmatrix} \sim \begin{pmatrix} fX + Zx_0 \\ fY + Zy_0 \\ Z \end{pmatrix} = \begin{bmatrix} f & & x_0 & 0 \\ & f & y_0 & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Richard Hartley and Andrew Zisserman."Multiple View Geometry in Computer Vision". Cambridge University Press, Second Edition 2004.

相机的外部参数



Richard Hartley and Andrew Zisserman."Multiple View Geometry in Computer Vision". Cambridge University Press, Second Edition 2004.

透视相机模型

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

$$\mathrm{P} = \mathrm{K}\left[\mathrm{R} \mid \mathrm{t}\right]$$

11 DoF (5+3+3)

径向畸变

- 比如鱼眼镜头：

- 数学模型：

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \sim \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \mathrm{R}\left(\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R}^\top & -\mathbf{R}^\top \mathbf{t} \\ 0_3^\top & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \right)$$

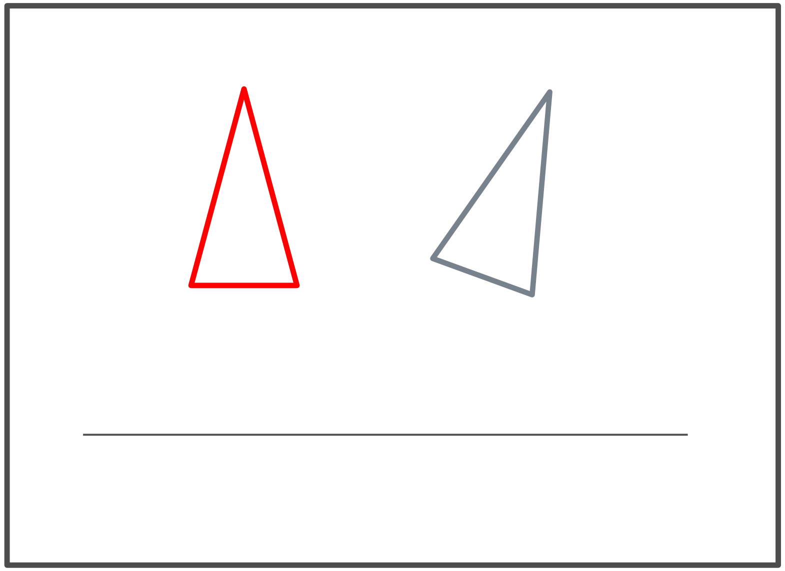
$$\mathrm{R}\,(x, y) = (1 + K_1(x^2 + y^2) + K_2(x^2 + y^2)^2 + \ldots) \begin{bmatrix} x \\ y \end{bmatrix}$$

(Marc Pollefeys)

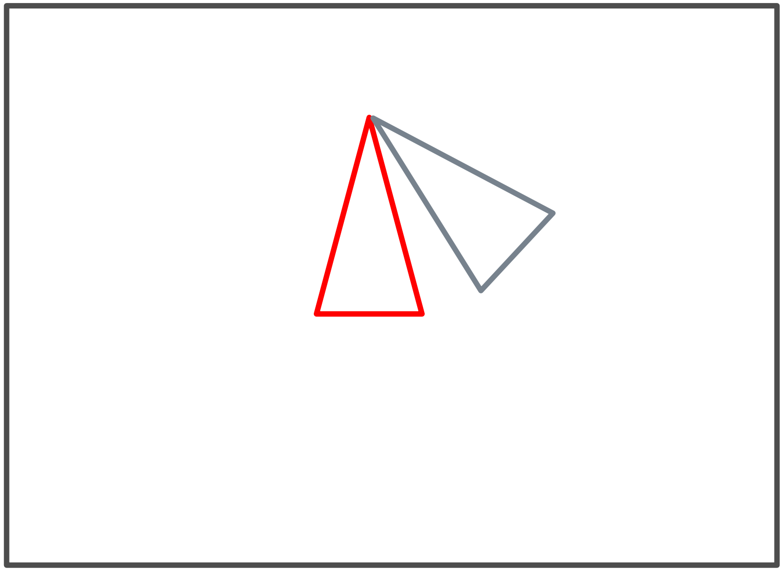
径向畸变矫正例子

(Marc Pollefeys)

单应性矩阵



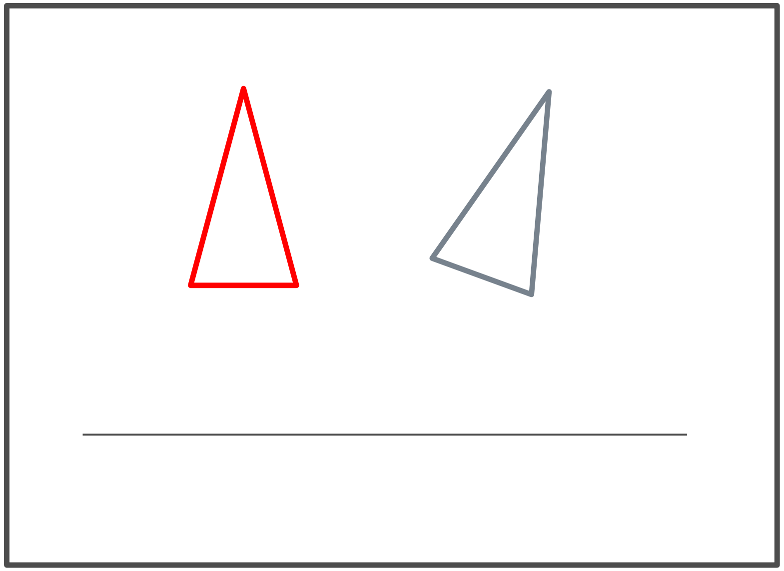
拍摄平面



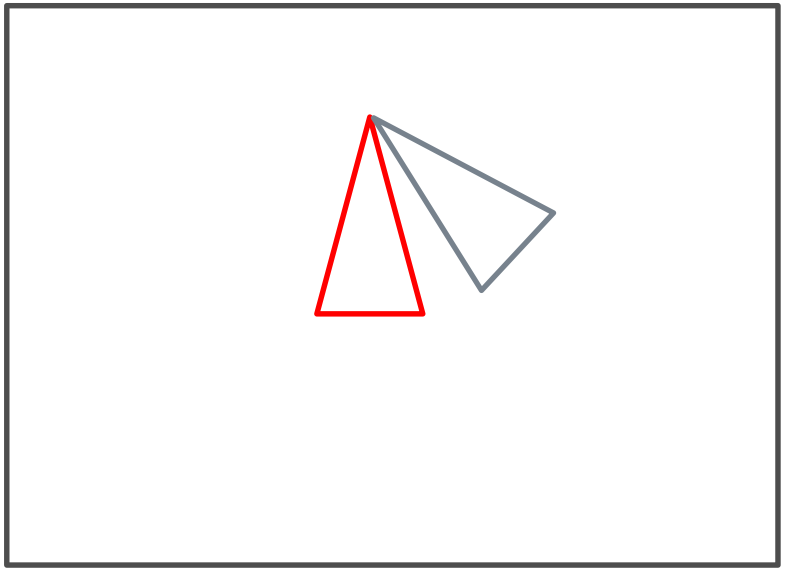
纯旋转

$$c\begin{pmatrix} u_2 \\ v_2 \\ 1 \end{pmatrix} = \begin{pmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{pmatrix} \begin{pmatrix} u_1 \\ v_1 \\ 1 \end{pmatrix}$$

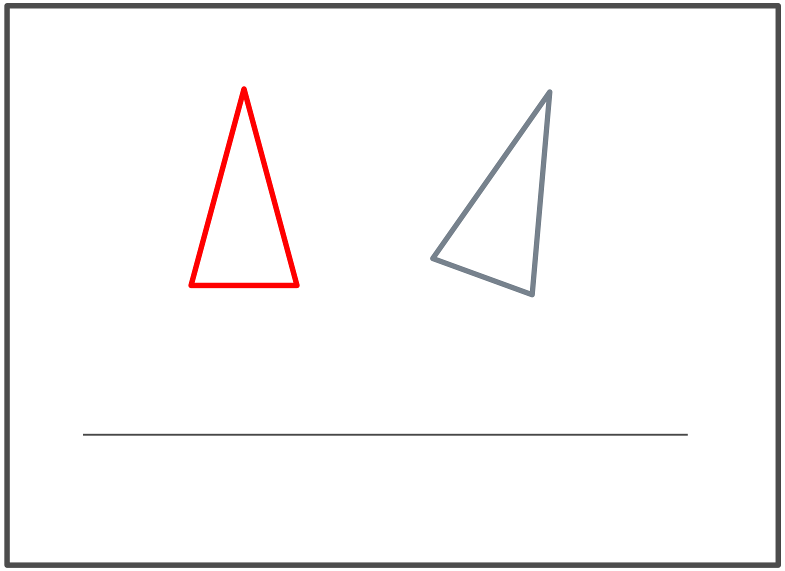
单应性矩阵



单应性矩阵



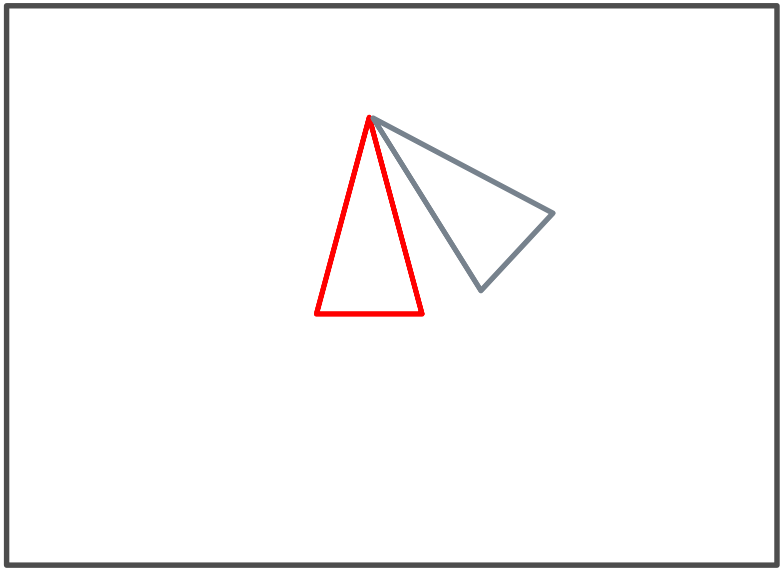
单应性矩阵



拍摄平面

$$\pi = \begin{bmatrix} \mathbf{n} \\ d \end{bmatrix}$$

$$H = K_2(R - \frac{tn}{d})K_1^{-1}$$



纯旋转

$$H = K_2RK_1^{-1}$$

向量外积的矩阵表示

向量的外积 **v**×**x** 可以被表示为矩阵乘法形式

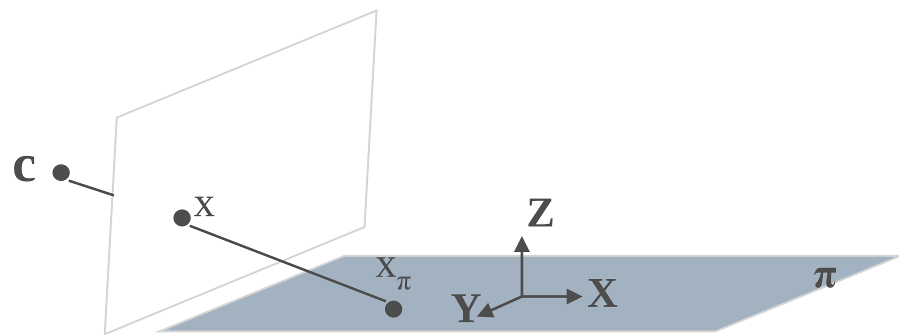
$$\mathbf{v}\times\mathbf{x}=[\mathbf{v}]_\times\mathbf{x}$$

其中

$$[v]_\times=\begin{bmatrix} 0 & -v_z & v_y \\ v_z & 0 & -v_x \\ -v_y & v_x & 0 \end{bmatrix}$$

- [**v**]× 是一个 3 × 3 秩为2的斜对称矩阵.
- **v** 是矩阵[**v**]×的零向量： **v** × **v** = [**v**]×**v** = **0**.

平面射影变换



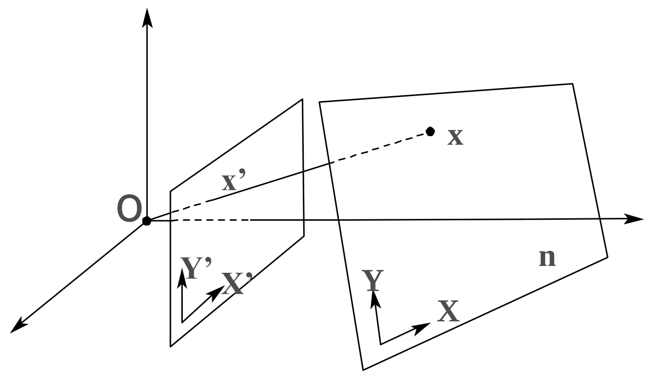
选择该平面作为世界坐标系的Z=0的平面，那么3 ×4的摄影矩阵可以被归约：

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{pmatrix} \begin{pmatrix} X \\ Y \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} p_{11} & p_{12} & p_{14} \\ p_{21} & p_{21} & p_{24} \\ p_{31} & p_{32} & p_{34} \end{pmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}$$

其中该3×3 的矩阵代表了从一个平面到另一个平面的变换

射影变换

$$\begin{pmatrix} x_1' \\ x_2' \\ x_3' \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$



$\mathbf{x}' = \mathrm{H}\mathbf{x}$, 其中 H 是一个 3 × 3 非奇异的齐次矩阵.

- 在使用透视相机成像时，这是一种常用的变换来构建世界和成像平面之间的关系
- 射影变换又被称为具有单应性或共线性。
- H 拥有 8 个自由度

多视图几何

- 运动恢复结构
  - 从多张图片或者视频序列中自动回复相机参数和场景三维结构

Noah Snavely, Steven M. Seitz, Richard Szeliski.
"Photo tourism: Exploring photo collections in 3D". 2016.

双视图几何

3D???

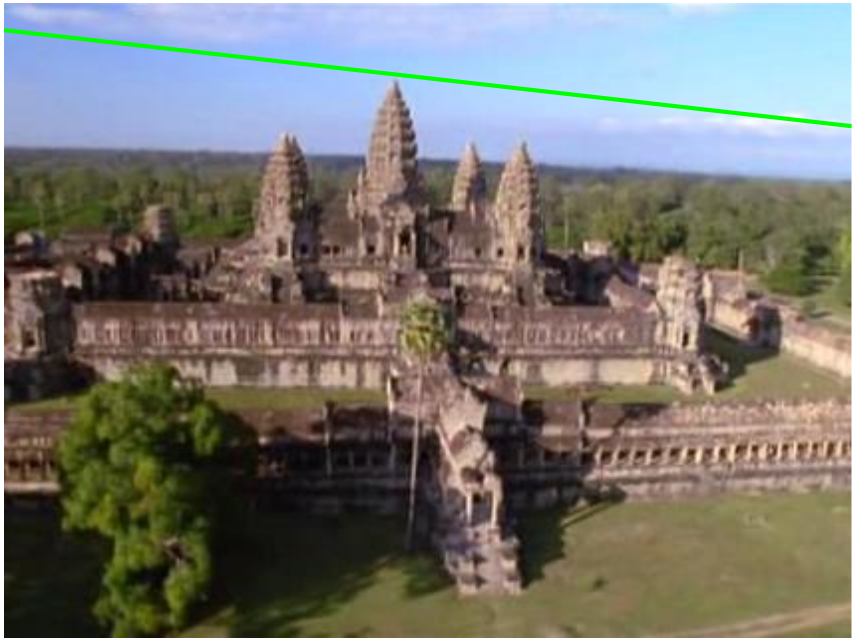
双视图几何

3D???

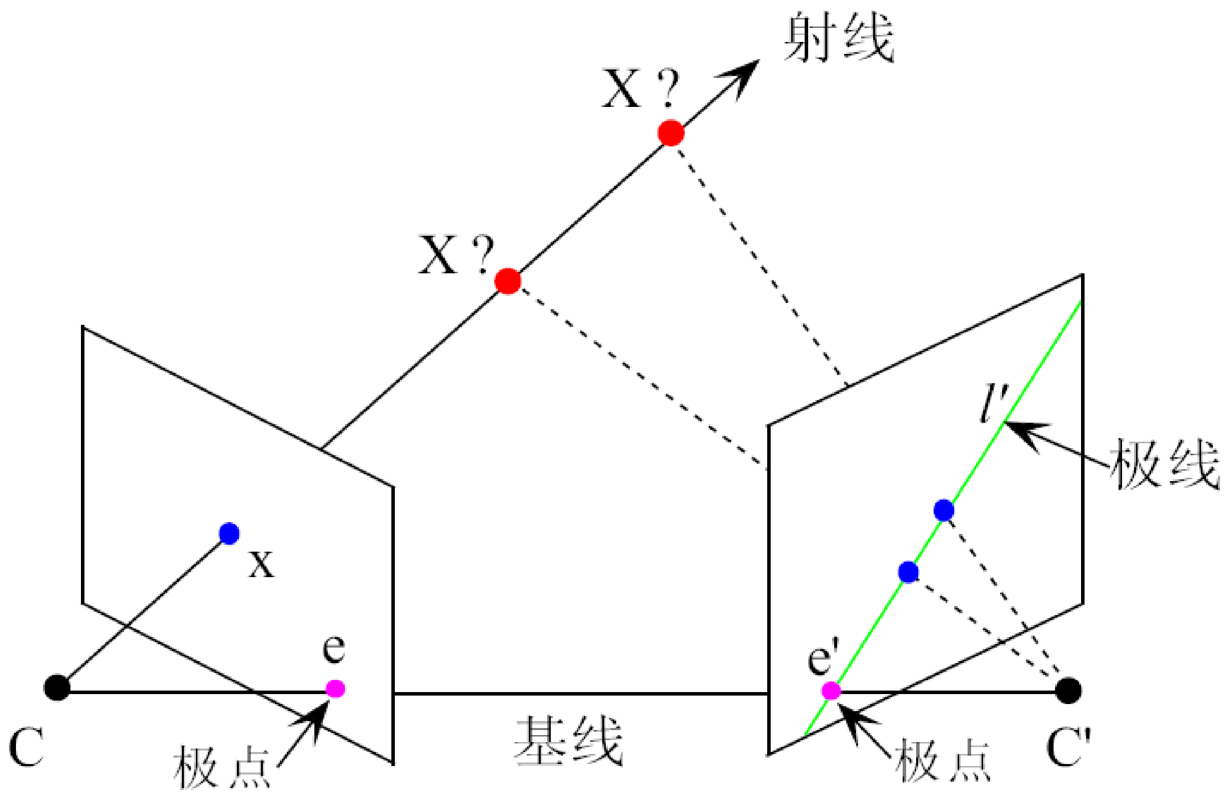
 

双视图几何

3D: 极线几何

极线几何



$$\hat{\mathbf{x}}'^{\top} F \hat{\mathbf{x}} = 0$$

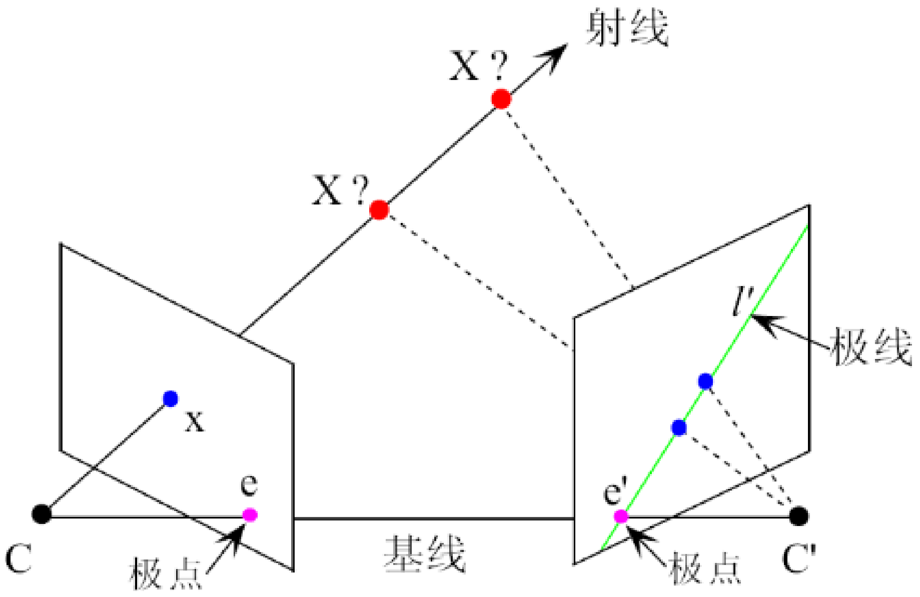
基础矩阵

- 只跟两个视图的相对相机姿态和内参有关

$$F = K_2^{-T}[t]_\times RK_1^{-1}$$

- F 是一个3 × 3 秩为2的矩阵
- Fe = **0**
- 7个自由度
- 最少7对匹配点就可以求解
  - 七点法
  - 八点法

OpenCV: cvFindFundamentalMat()



八点法求解基础矩阵

根据对极几何关系，基本矩阵 F 满足

$$\hat{x}'^{\top} F \hat{x} = 0$$

若设 $\mathbf{f} = (f_{11}, f_{12}, f_{13}, f_{21}, f_{22}, f_{23}, f_{31}, f_{32}, f_{33})^{\top}$

那么对极几何关系又可以写作：

$$\begin{pmatrix} \hat{x}'_1\hat{x}_1 & \hat{x}'_1\hat{x}_2 & \hat{x}'_1 & \hat{x}'_2\hat{x}_1 & \hat{x}'_2\hat{x}_2 & \hat{x}'_2 & \hat{x}_1 & \hat{x}_2 & 1 \end{pmatrix} \mathbf{f} = 0$$

若存在 n 对对应点，F 应满足如下的线性系统：

$$A\mathbf{f} = \begin{pmatrix} \hat{x}'_{11}\hat{x}_{11} & \hat{x}'_{11}\hat{x}_{12} & \hat{x}'_{11} & \hat{x}'_{12}\hat{x}_{11} & \hat{x}'_{12}\hat{x}_{12} & \hat{x}'_{12} & \hat{x}_{11} & \hat{x}_{12} & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \hat{x}'_{n1}\hat{x}_{n1} & \hat{x}'_{n1}\hat{x}_{n2} & \hat{x}'_{n1} & \hat{x}'_{n2}\hat{x}_{n1} & \hat{x}'_{n2}\hat{x}_{n2} & \hat{x}'_{n2} & \hat{x}_{n1} & \hat{x}_{n2} & 1 \end{pmatrix} \mathbf{f} = 0$$

基础矩阵

- $\mathbf{f}$ 为 9 维向量，若要有解，rank(F) 至多为 8
  - 在 rank(F) = 8 时，$\mathbf{f}$ 的方向是唯一的
  - 通过至少 8 对对应点，可恰好得到使 $\mathbf{f}$ 方向唯一的 F
- $\mathbf{f}$ 为 F 的右零空间的基向量，可用 SVD(F) 求得
- 当对应点超过8对时且可能有外点时，我们一般先使用RANSAC方法来求解并筛选出内点，并求解得到最优的F^*。

基础矩阵

- 在得到初解后，我们一般还要根据所有内点对F做非线性优化，其中g为距离度量函数， x_i, x_i'为匹配点对：

$$\operatorname*{argmin}_{F} \sum_{i} g(x_i, x_i')$$

- 一般使用LM算法来优化该目标函数。
- 常见的两种距离度量
  - 辛普森距离
  - 对称极线距离

基础矩阵

- 一阶几何误差(first-order geometric error)，又名辛普森距离(Sampson distance）：
  - 令 $e = x_i'^T F x_i$，$J = \frac{\delta(x_i'^T F x_i)}{\delta x_i}$则第 i 对对应点的辛普森距离为

$$\frac{e^T e}{JJ^T} = \frac{(x_i'^T F x_i)^2}{JJ^T} = \frac{(x_i'^T F x_i)^2}{(F x_i)_1^2 + (F x_i)_2^2 + (F^T x_i'^T)_1^2 + (F^T x_i'^T)_2^2}$$

- 对称极线距离(symmetric epipolar distance)，它形式上与辛普森距离很像，但是度量的是点到极线的距离：

$$\frac{(x_i'^T F x_i)^2}{(F x_i)_1^2 + (F x_i)_2^2} + \frac{(x_i'^T F x_i)^2}{(F^T x_i'^T)_1^2 + (F^T x_i'^T)_2^2}$$