计算机视觉

立体视觉 上

中国传媒大学
COMMUNICATION UNIVERSITY OF CHINA

本节主题：

极线几何学

本节主题：

极线几何学

基础矩阵

我们如何从图像中恢复3D几何结构?

Shape-from

X


透视线索

纹理线索

阴影线索

聚焦线索

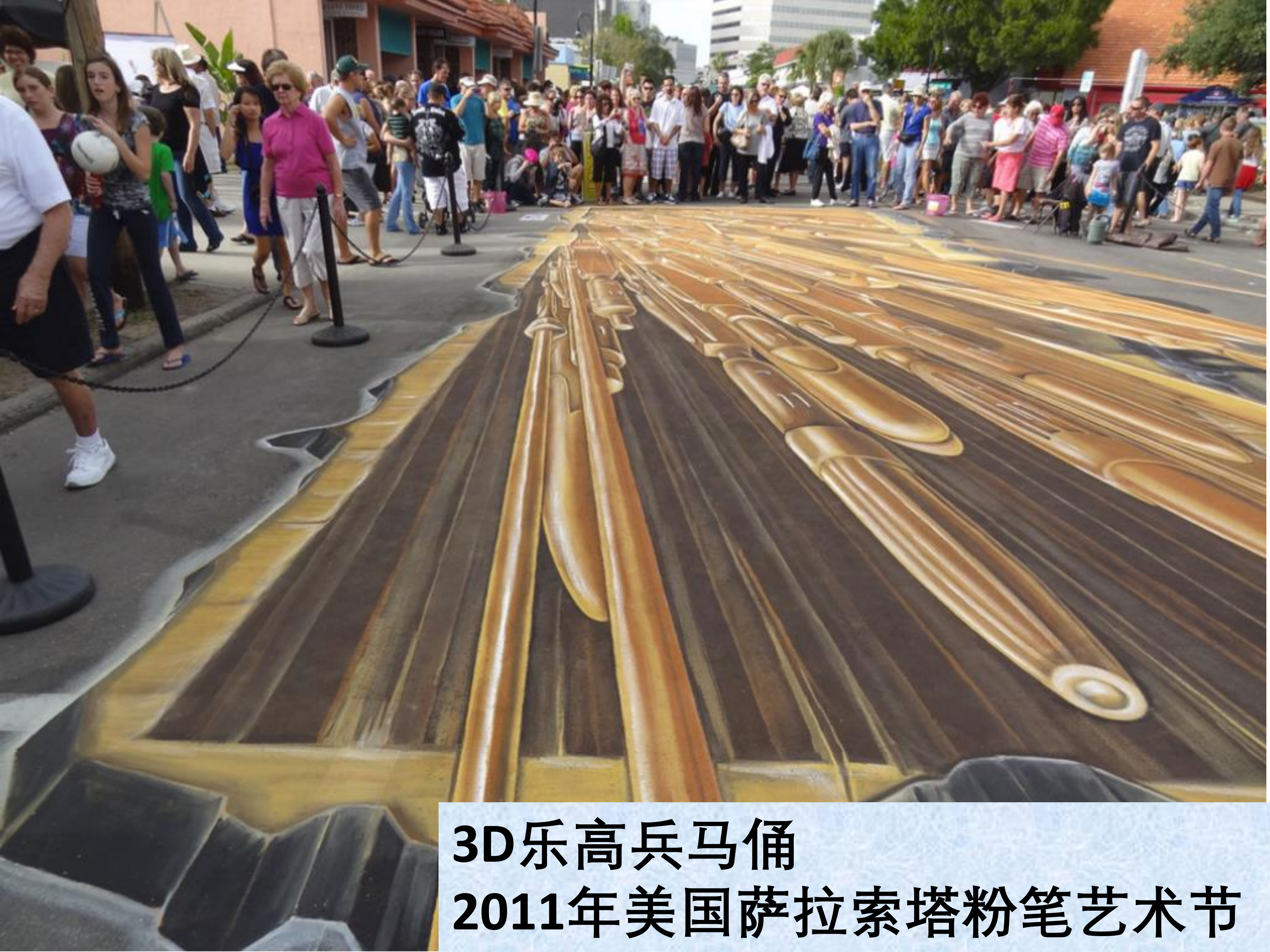
聚焦线索

为什么需要多视点？


单视点是有歧义的


STARBUCKS

3D乐高兵马俑
2011年美国萨拉索塔粉笔艺术节



3D乐高兵马俑
2011年美国萨拉索塔粉笔艺术节

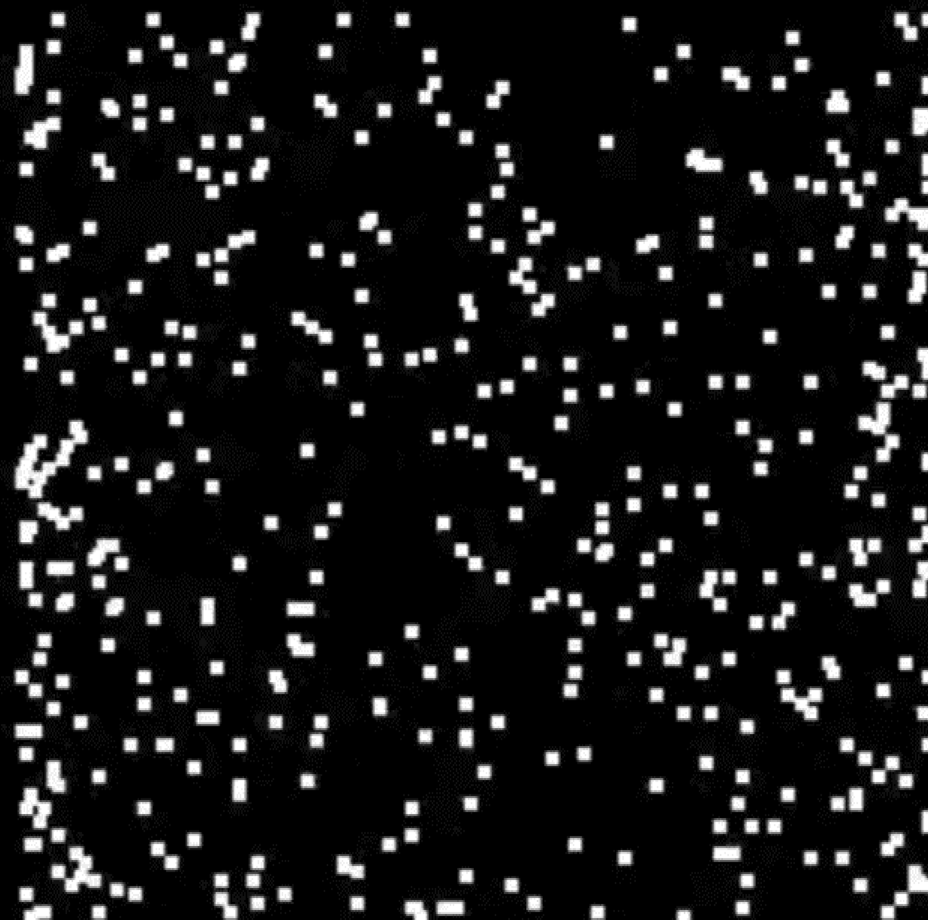

SPORTSNATION

Eker and Jepson, CVPR 2010

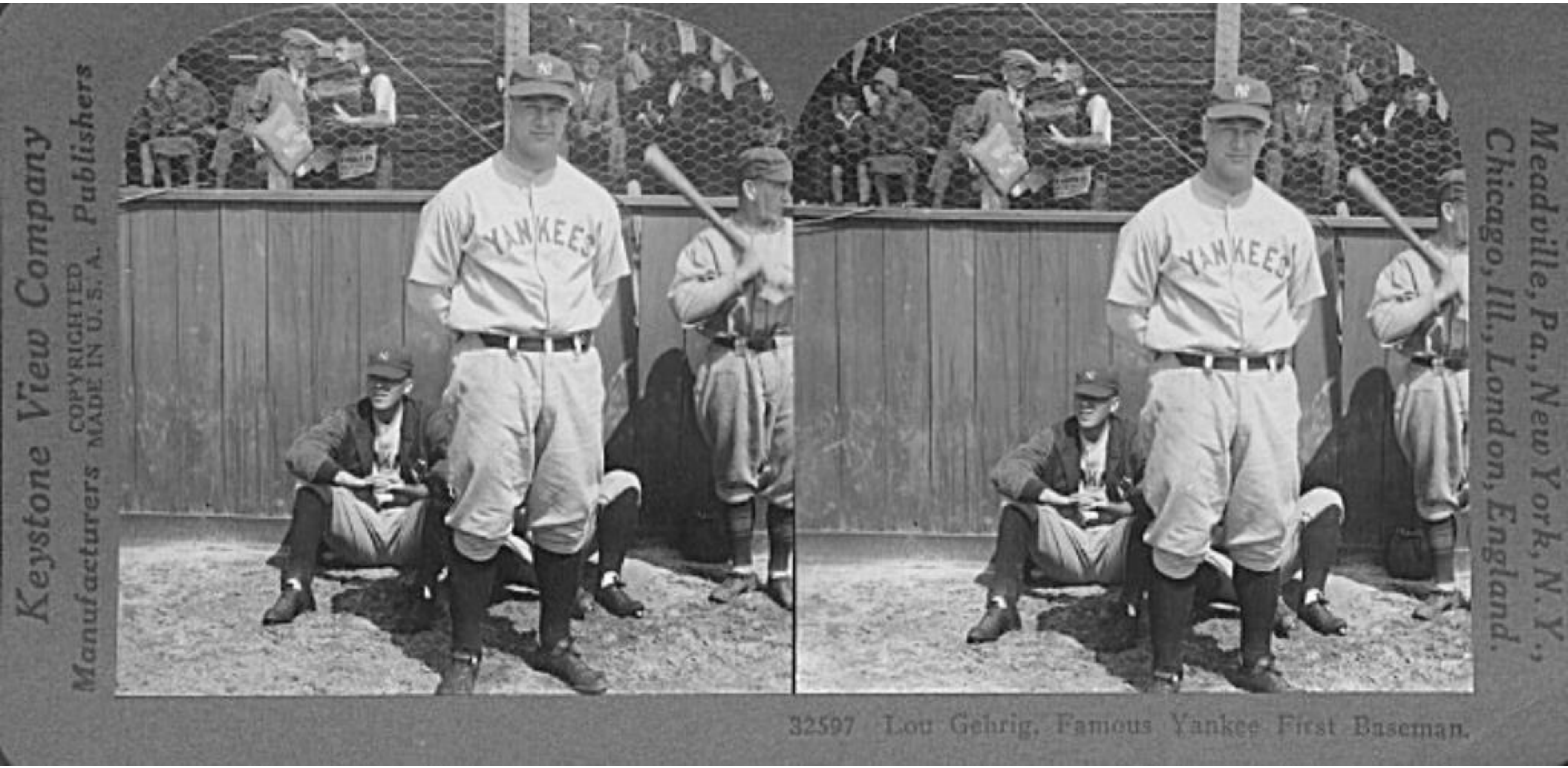

多视点
BCR '06

运动视差

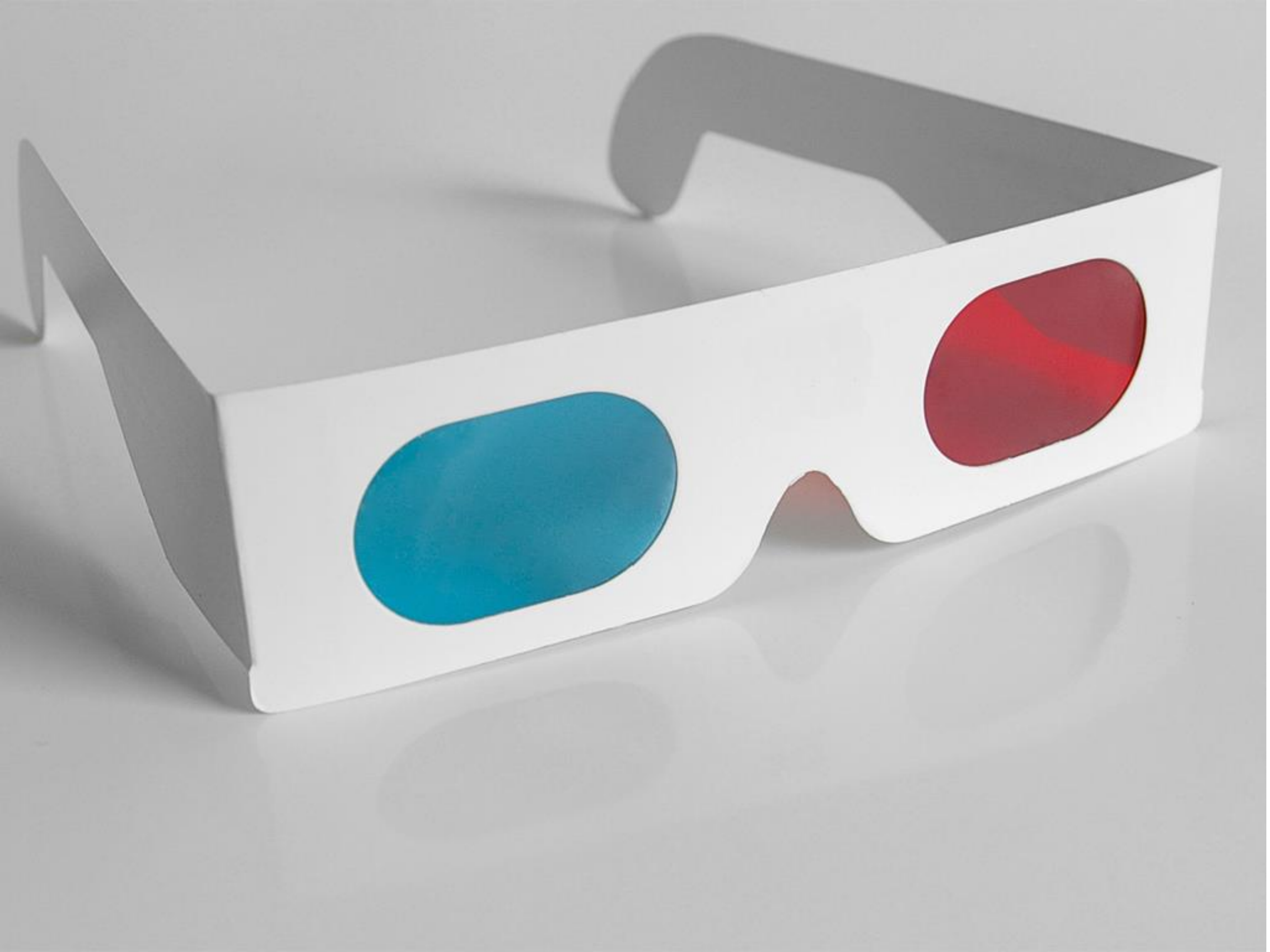
运动视差

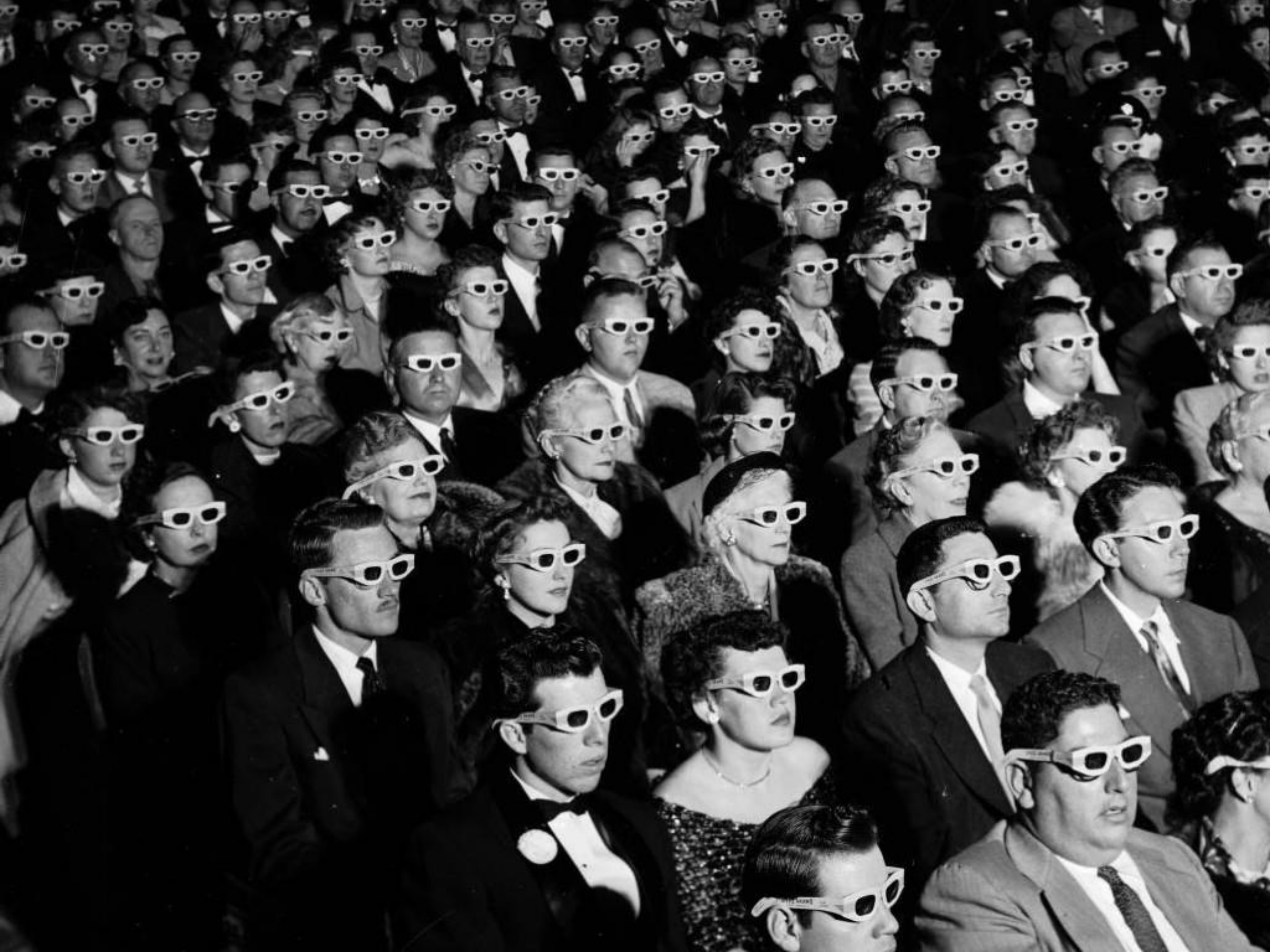


双目立体视觉



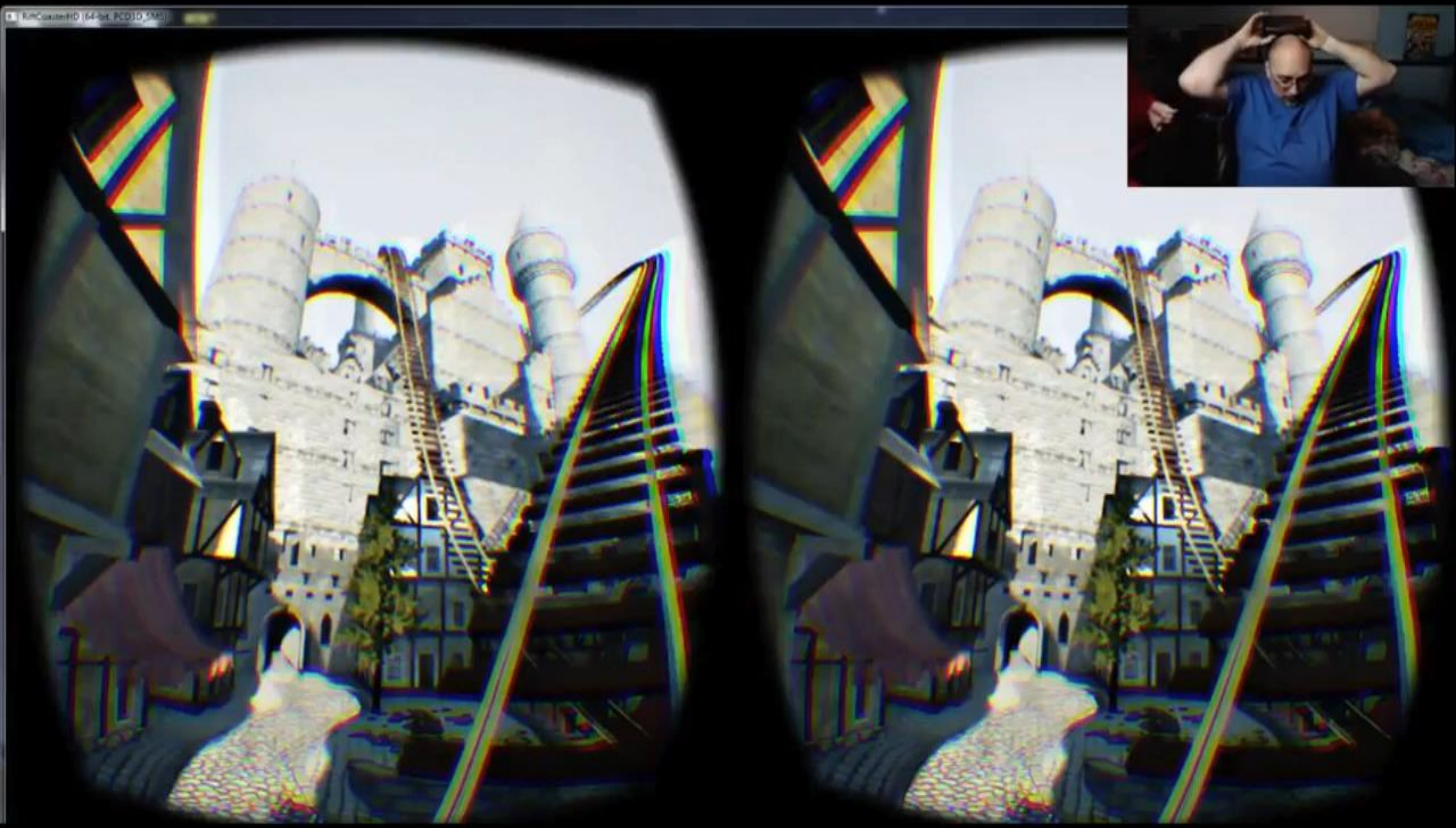
Lou Gehrig (1903-1941), 纽约扬基队一垒手



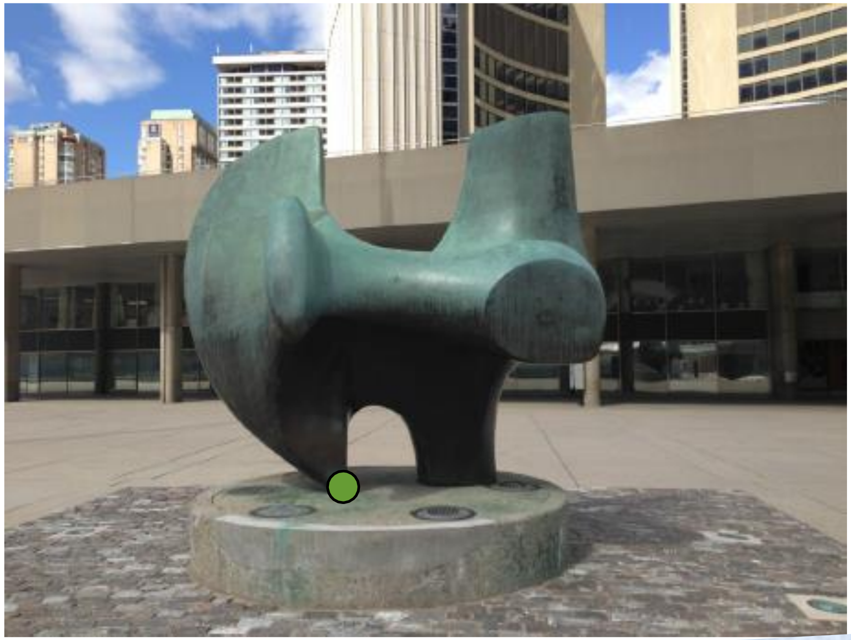



realD 3D

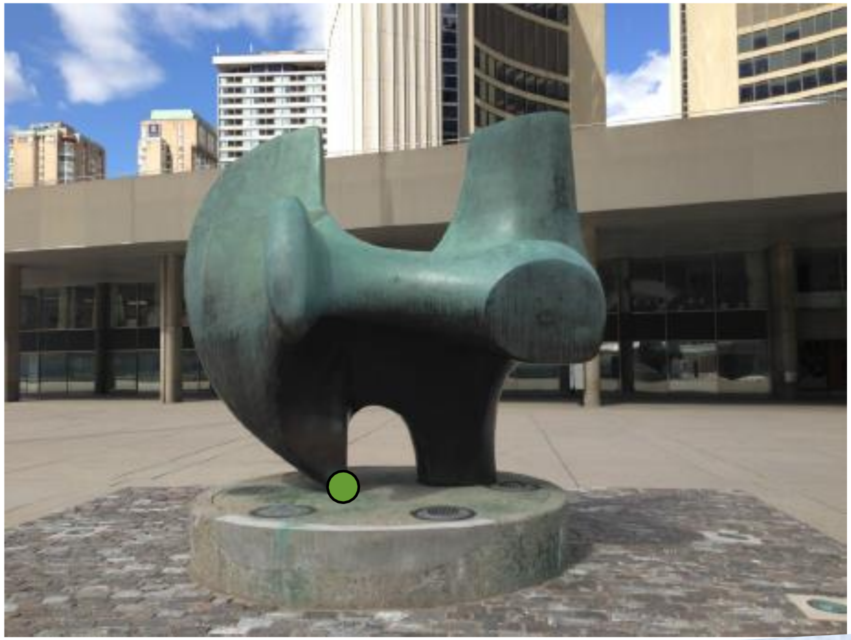

Oculus
Oculus

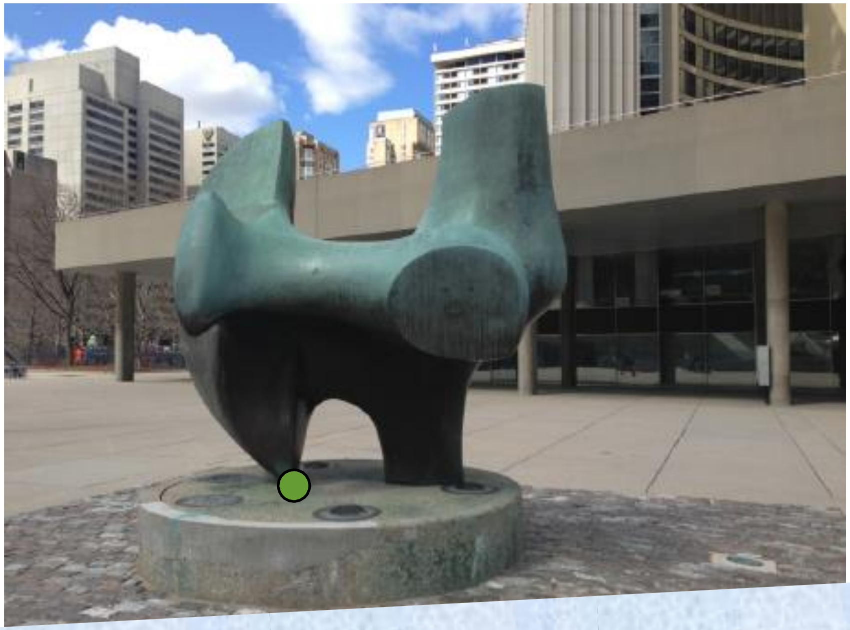


极线约束

给定左图中的一点



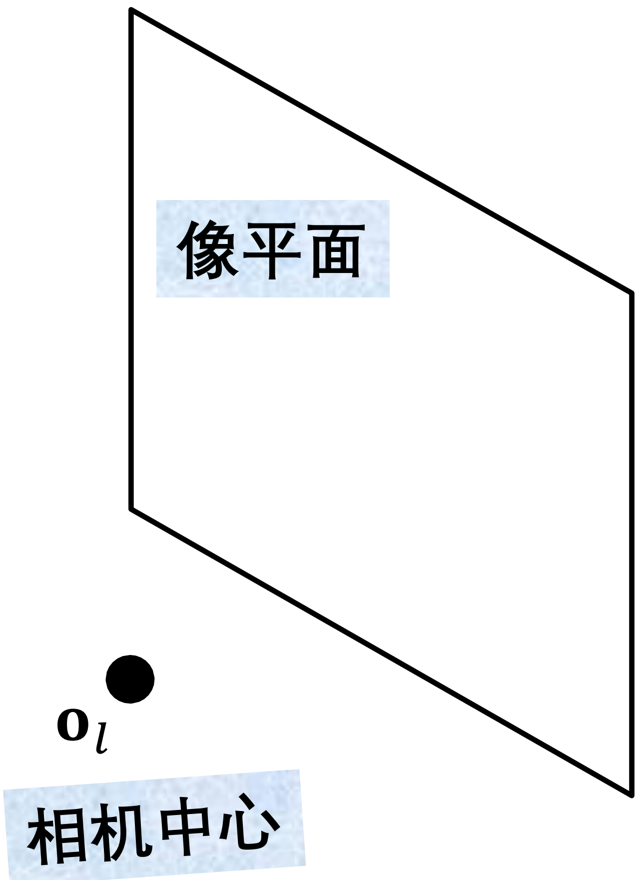


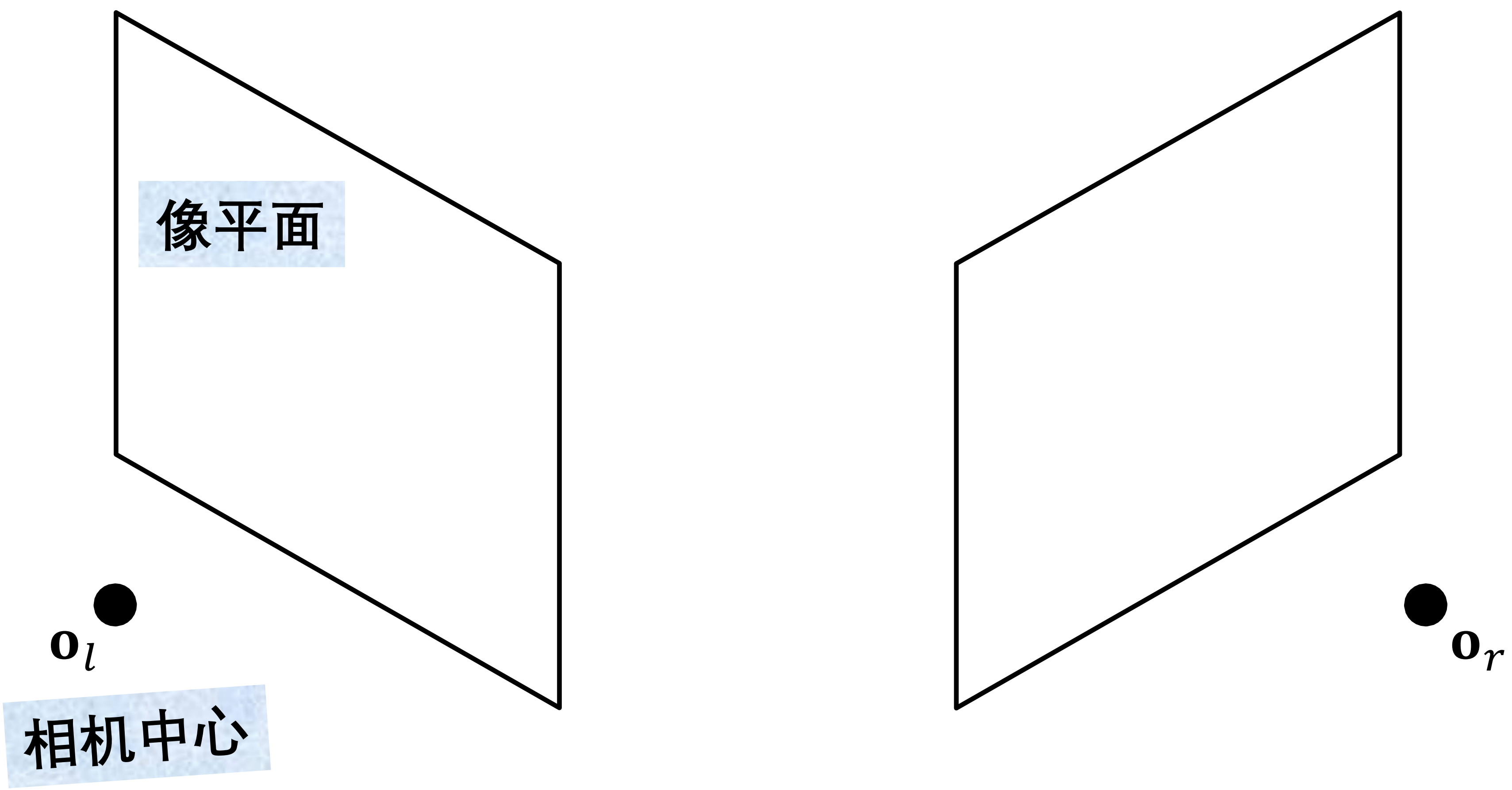
给定左图中的一点

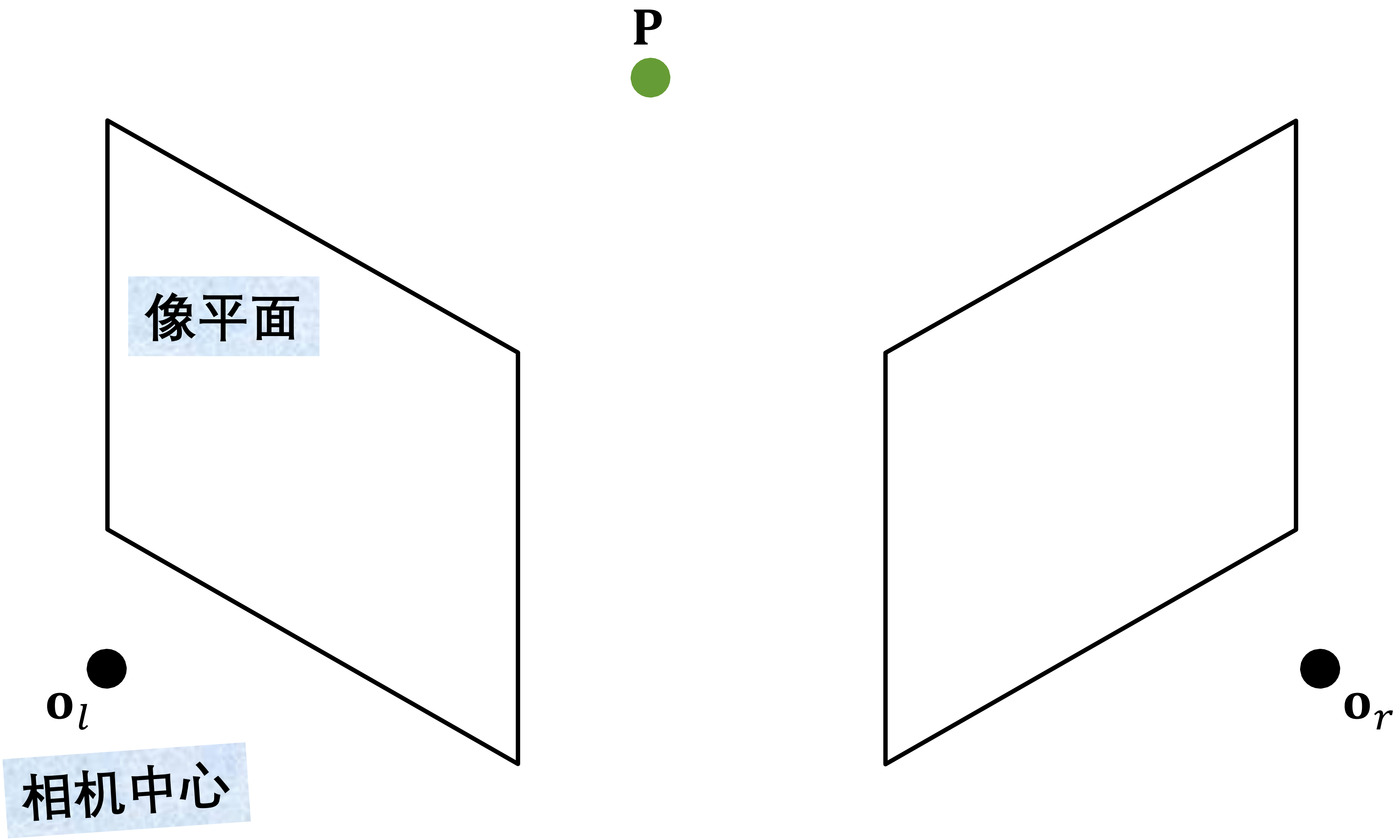
如何找出它在右图中的对应点

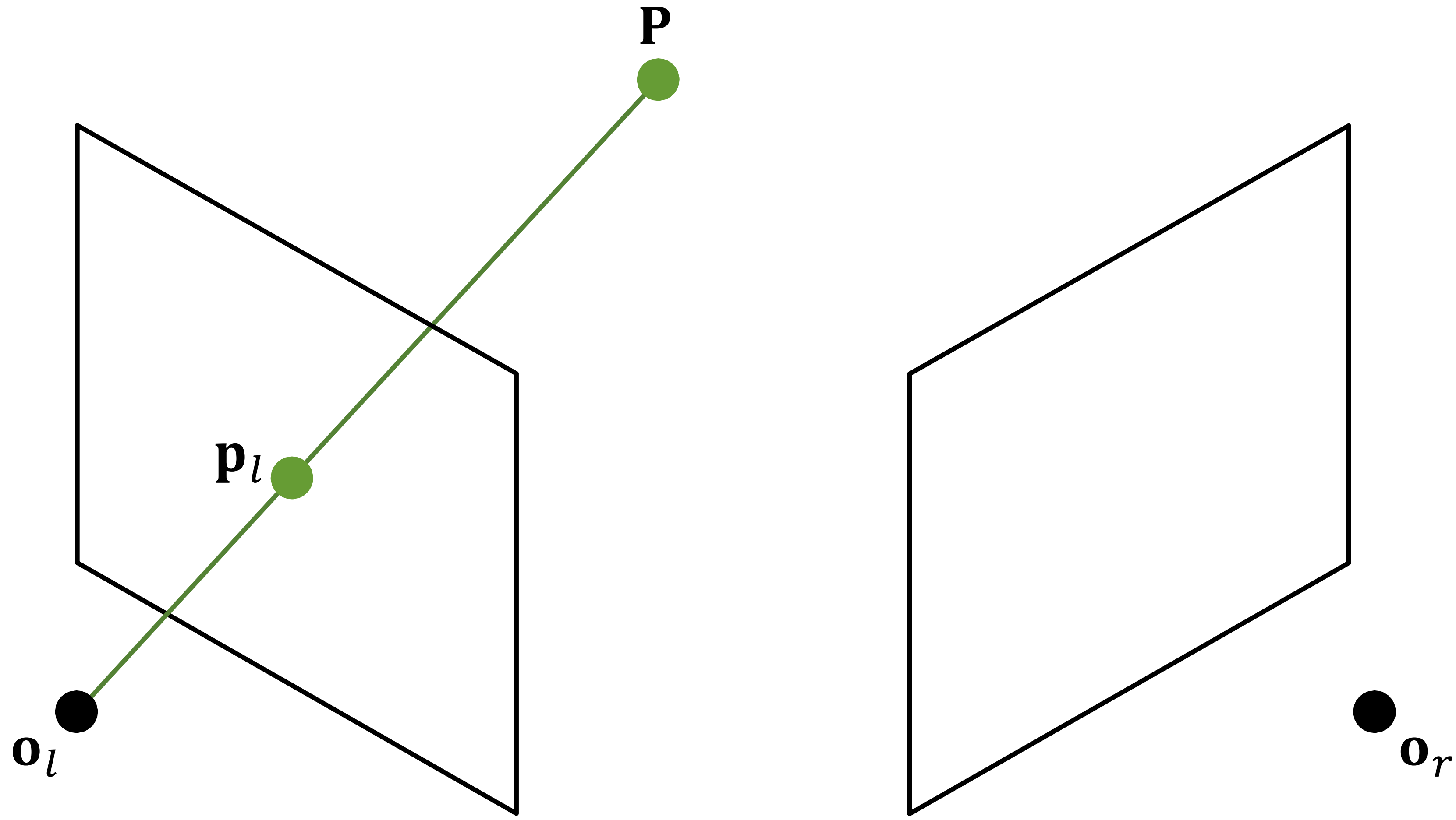
极线约束

极线约束

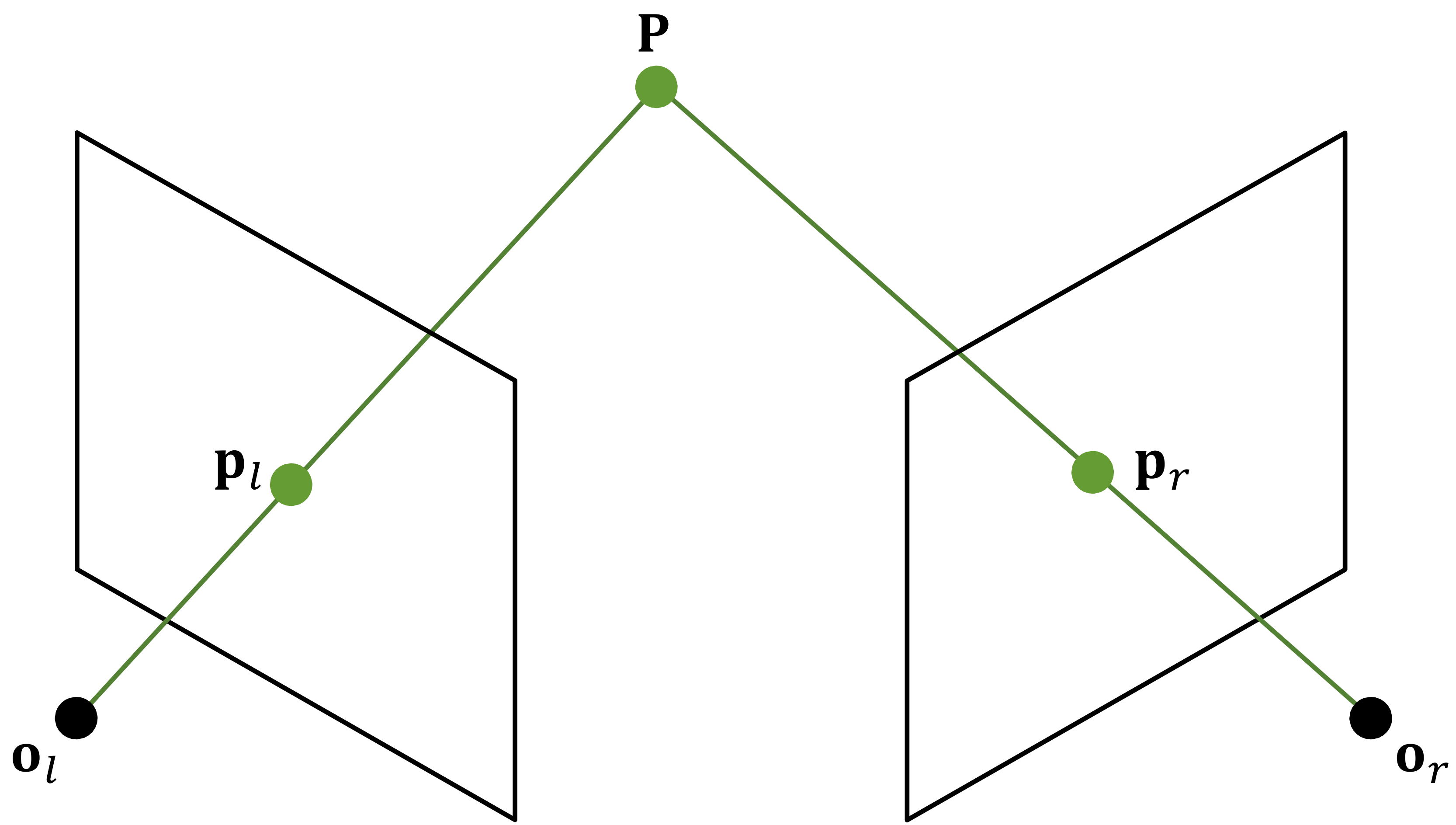

像平面
$\mathbf{o}_l$
相机中心

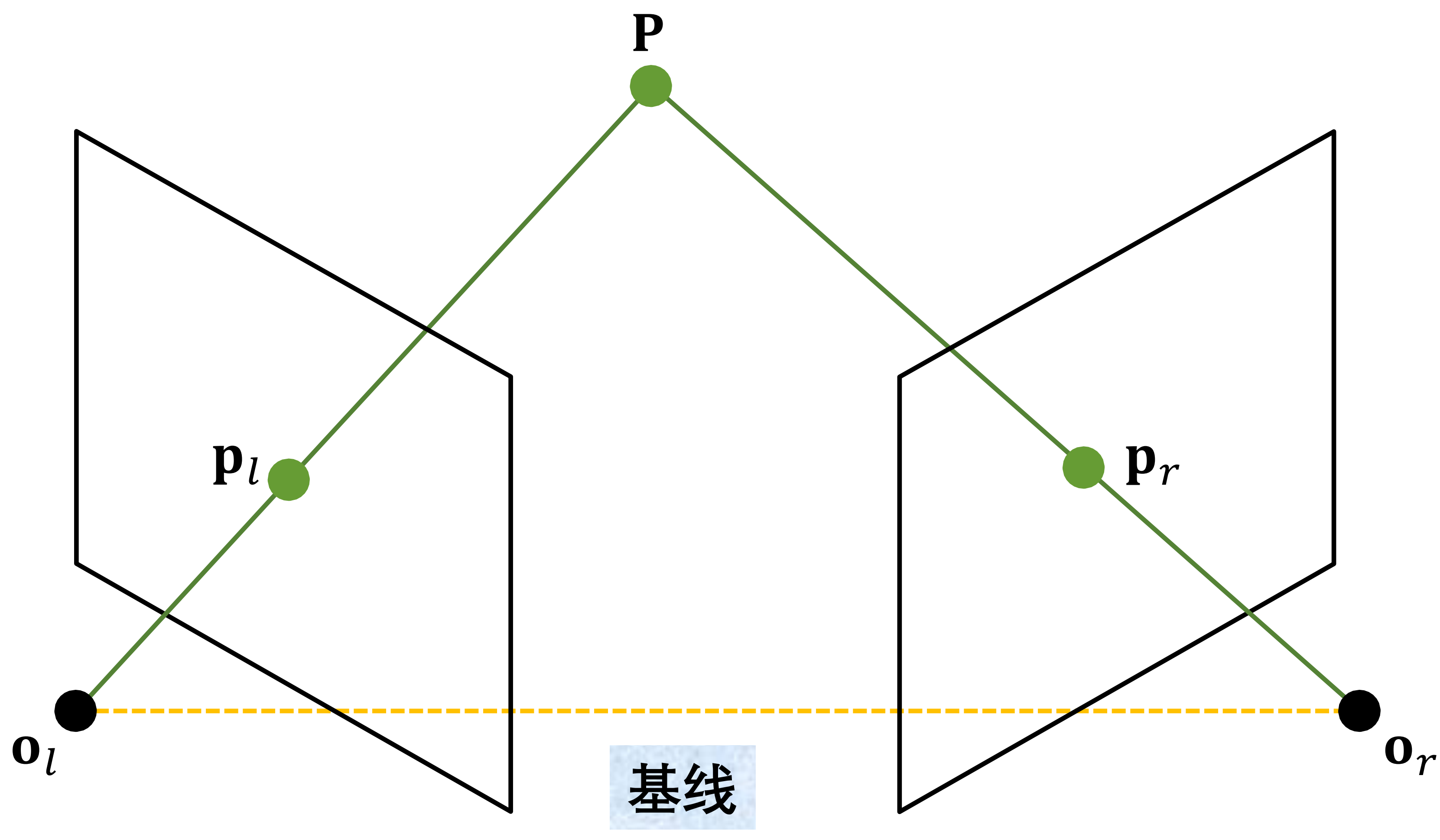

像平面
$\mathbf{o}_l$
$\mathbf{o}_r$
相机中心


P
像平面
o_l
o_r
相机中心

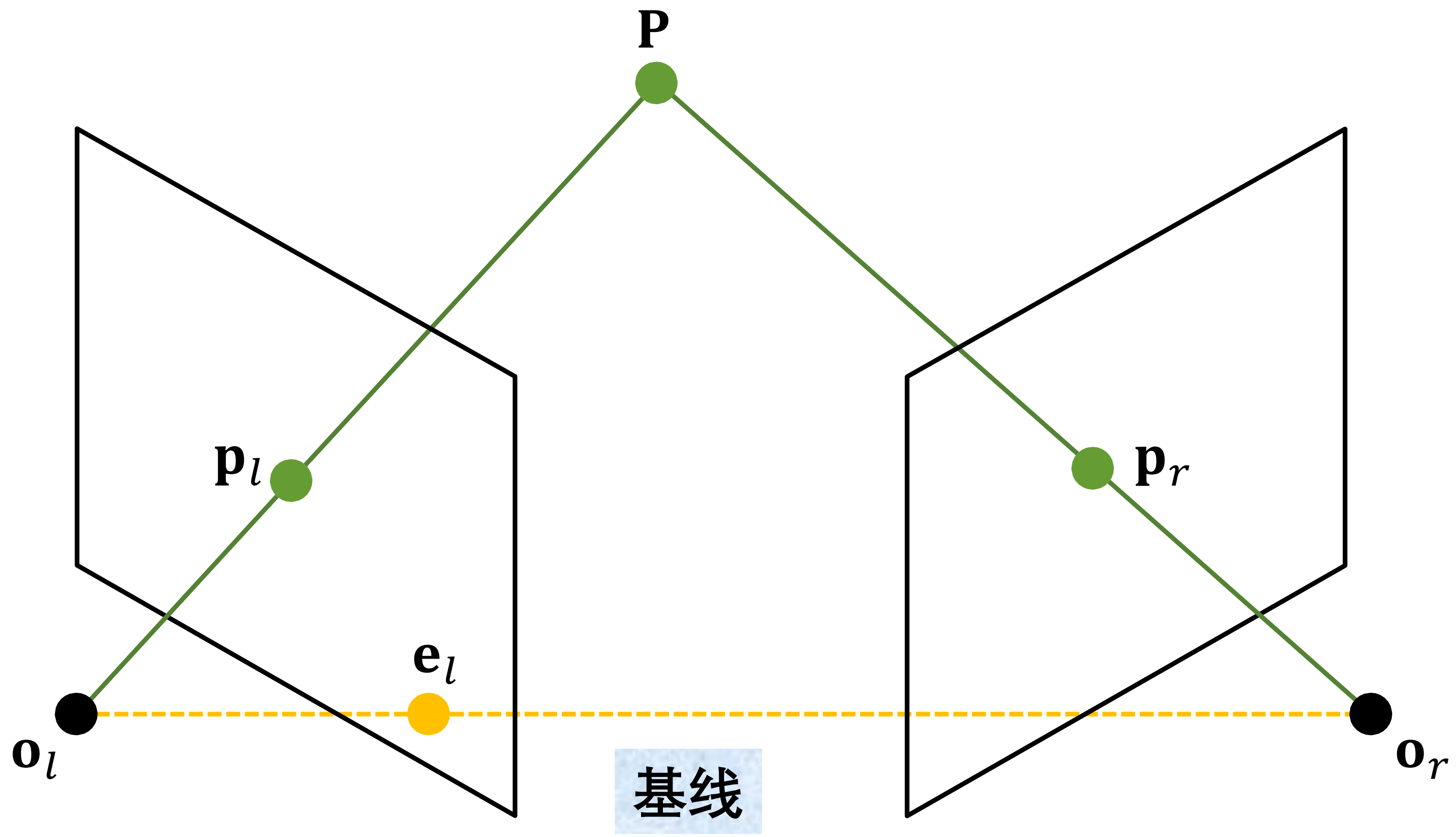


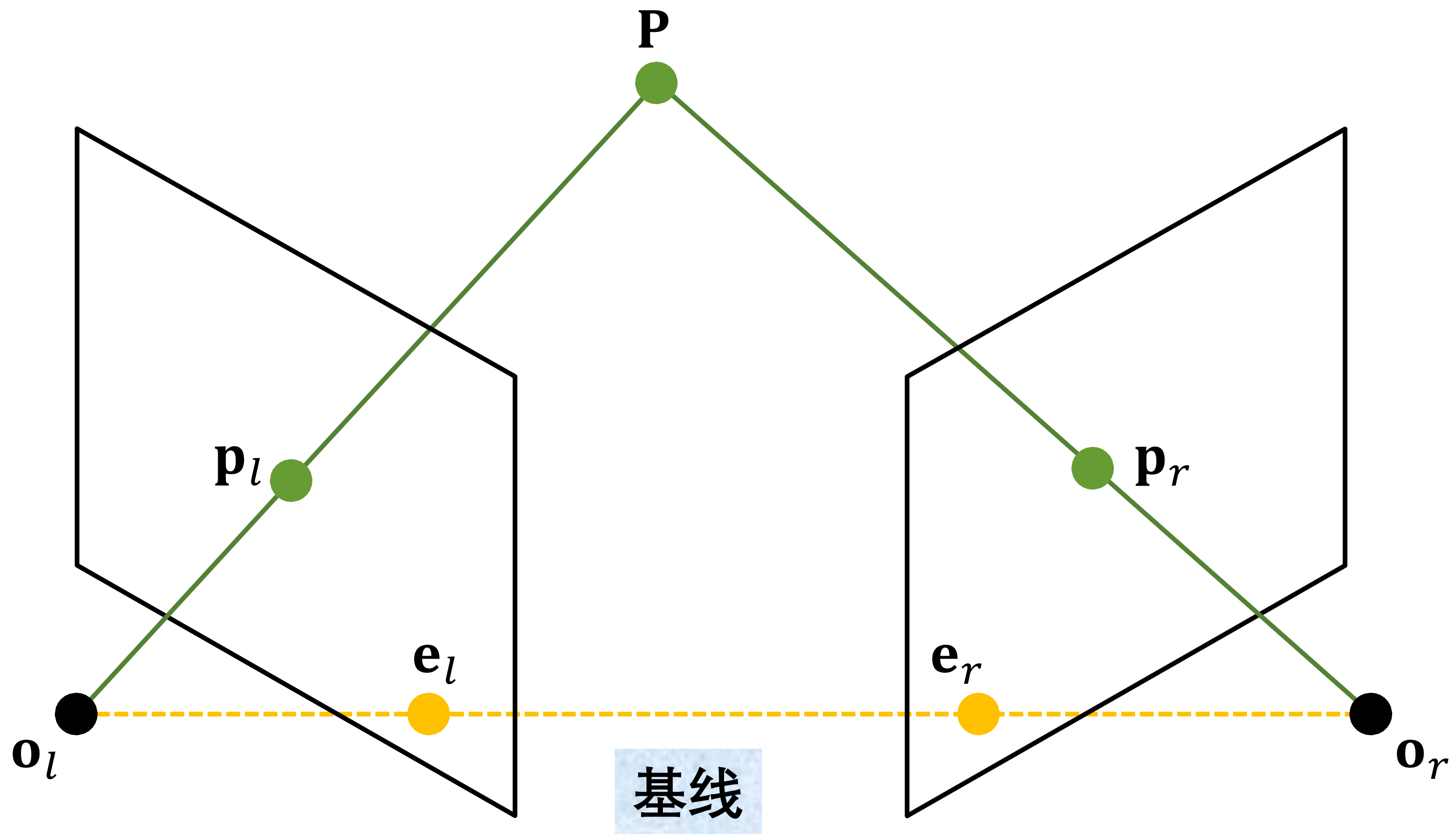
$\mathbf{P}$
$\mathbf{p}_l$
$\mathbf{o}_l$
$\mathbf{o}_r$

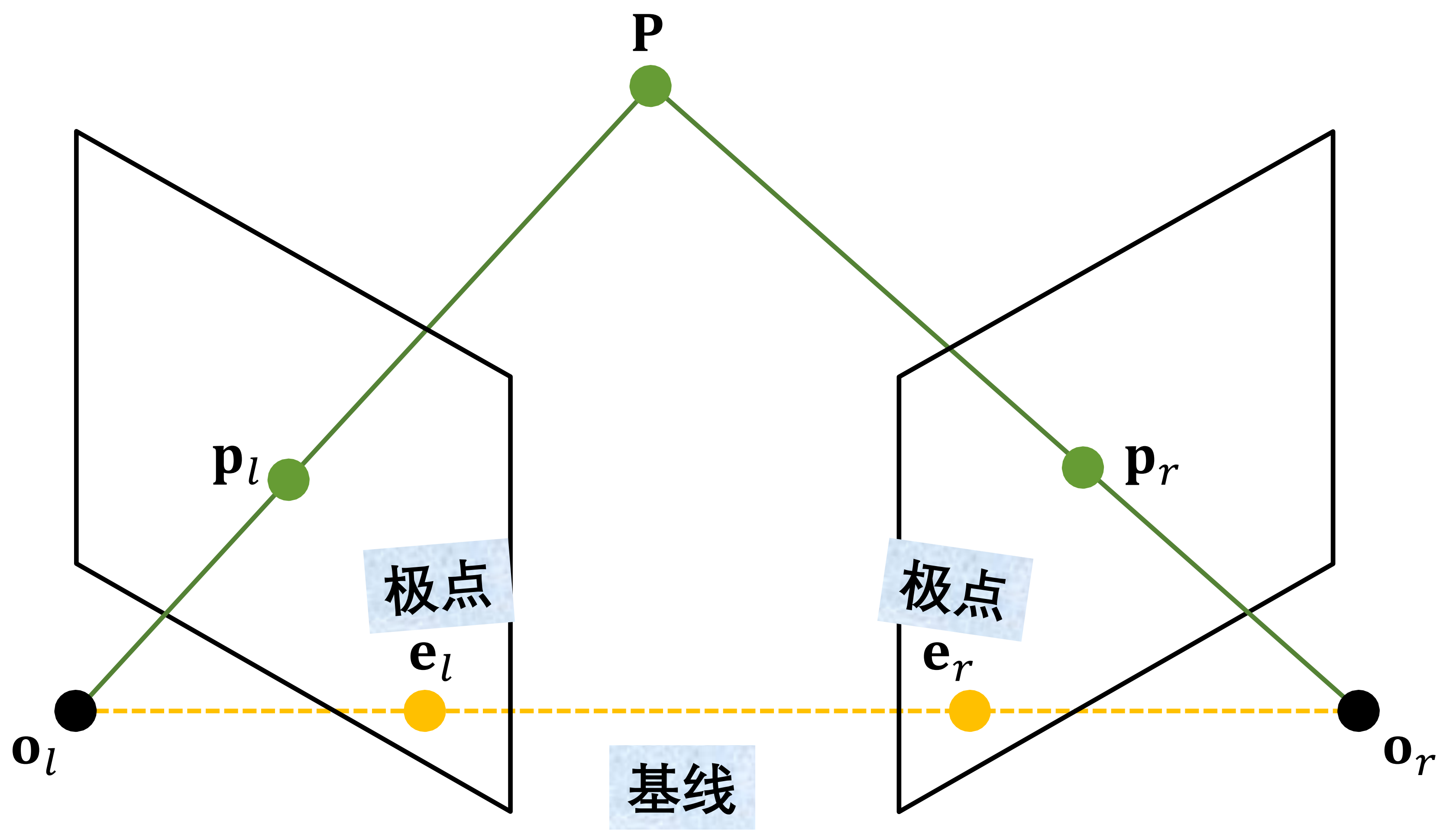

P
p_l
p_r
o_l
o_r



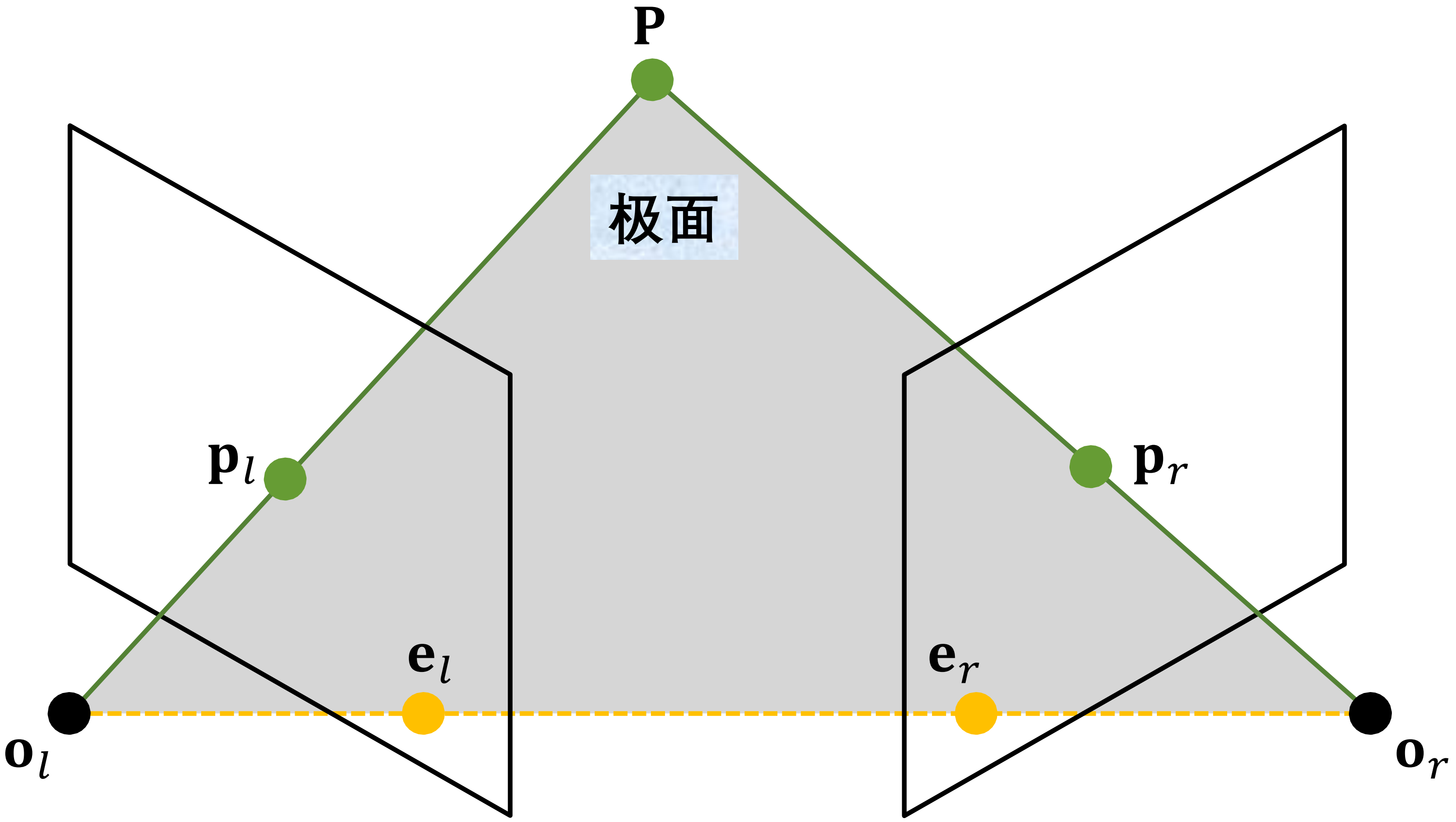
$\mathbf{P}$
$\mathbf{p}_l$
$\mathbf{p}_r$
$\mathbf{o}_l$
$\mathbf{o}_r$
基线

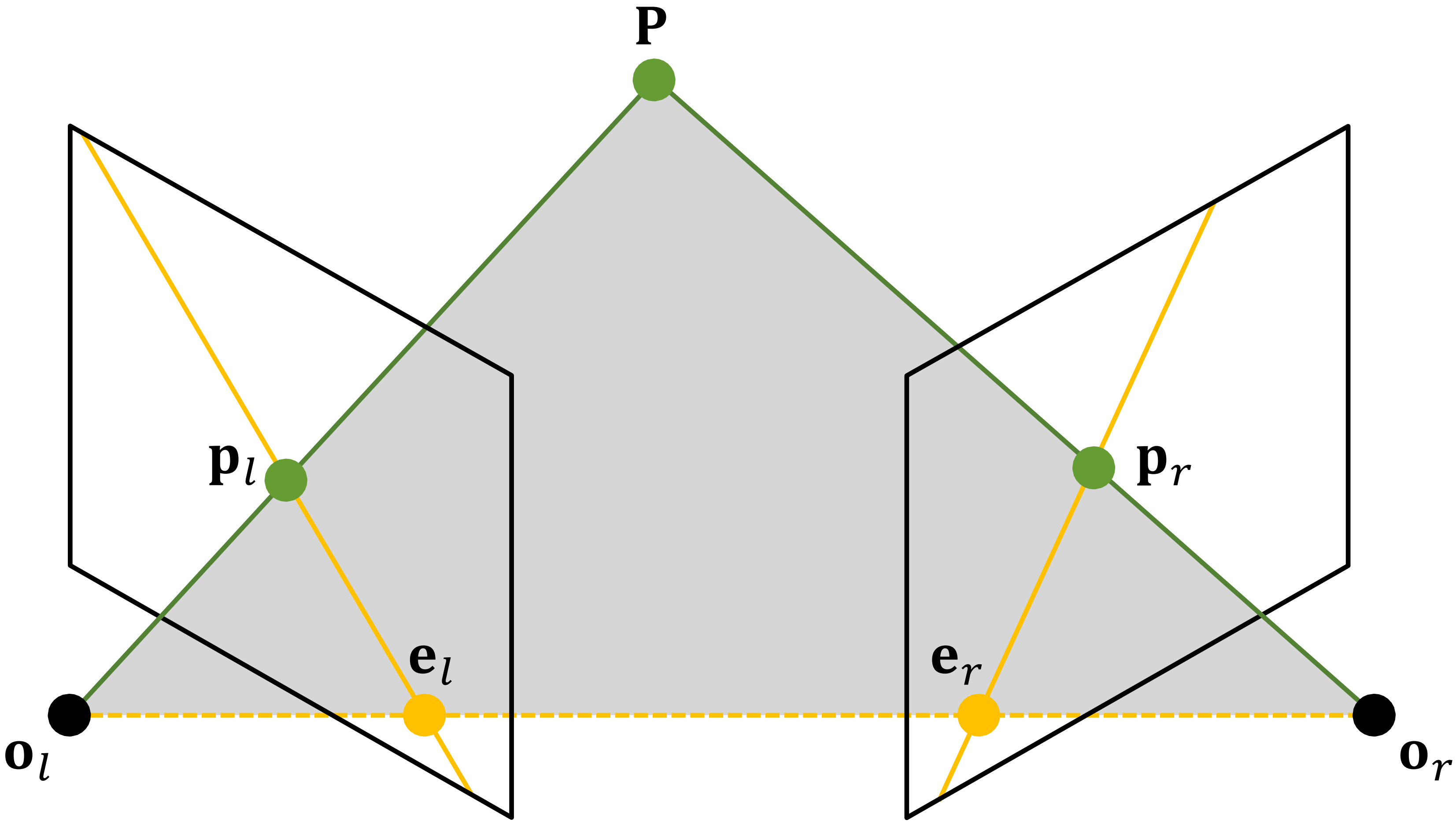

P
p_l
p_r
e_l
o_l
o_r
基线

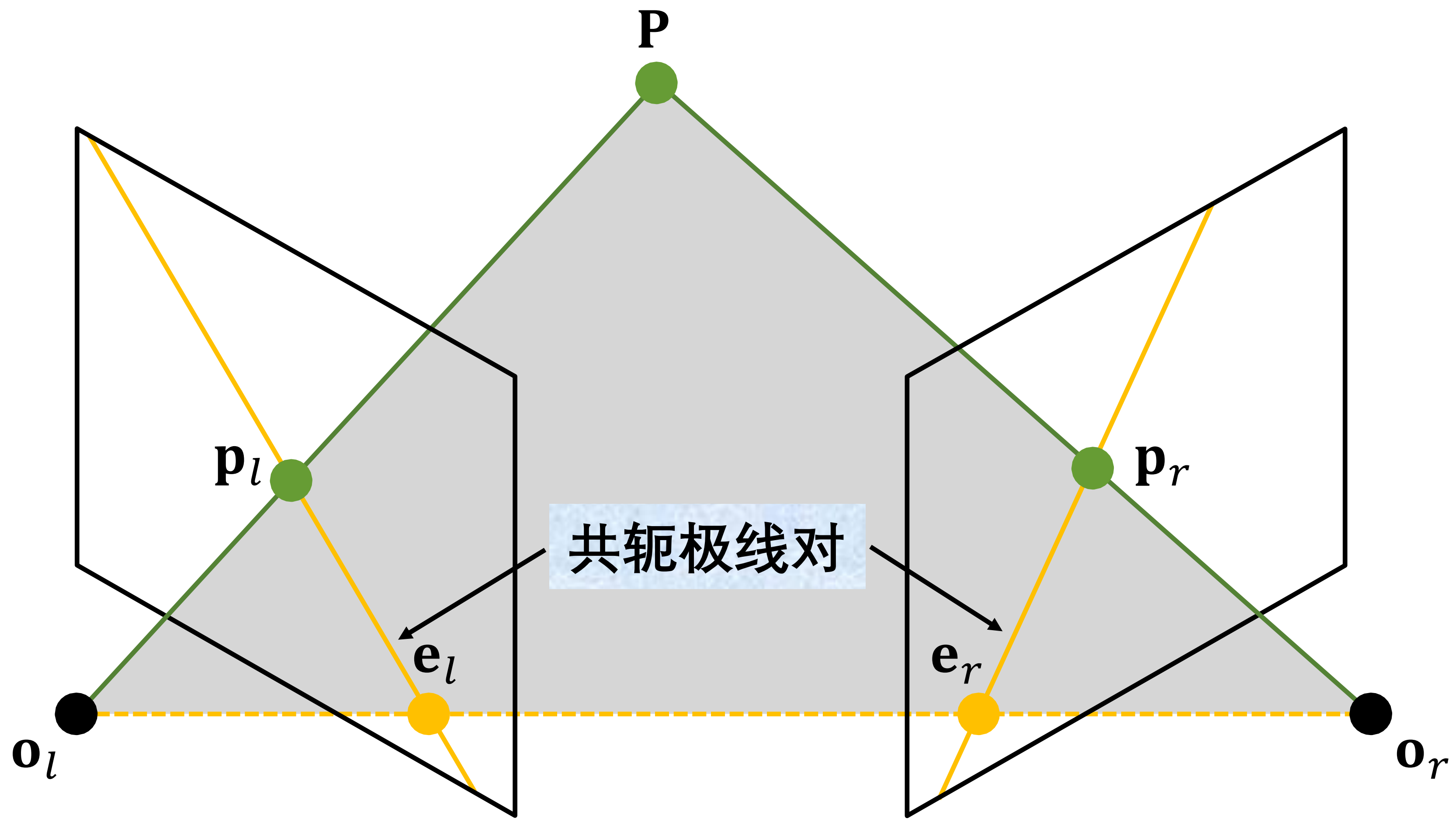

P
p_l
p_r
e_l
e_r
o_l
o_r
基线

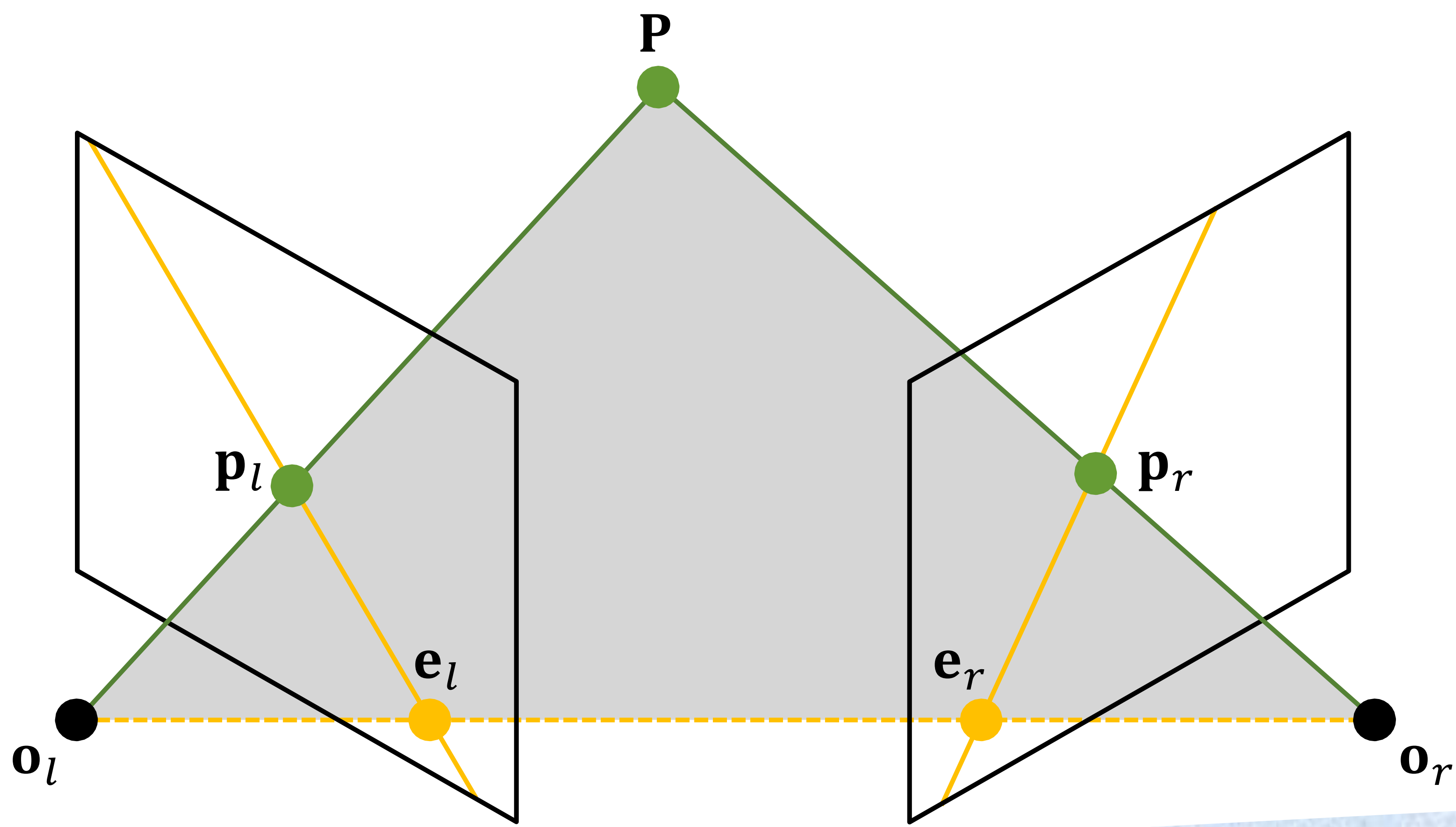


P
p_l
p_r
极点
极点
e_l
e_r
o_l
o_r
基线

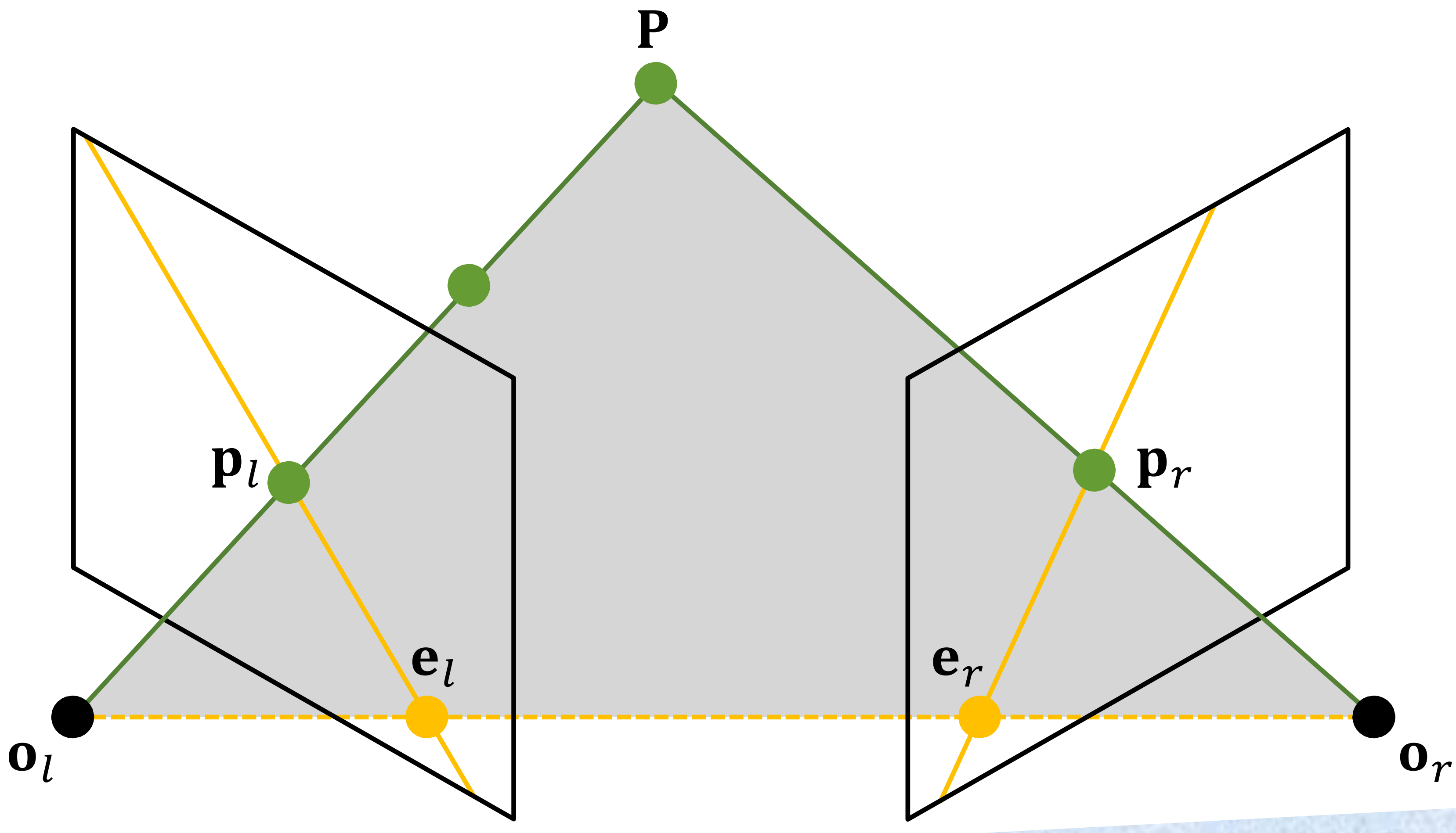

P
极面
$\mathbf{p}_l$
$\mathbf{p}_r$
$\mathbf{e}_l$
$\mathbf{e}_r$
$\mathbf{o}_l$
$\mathbf{o}_r$


P
p_l
p_r
e_l
e_r
o_l
o_r

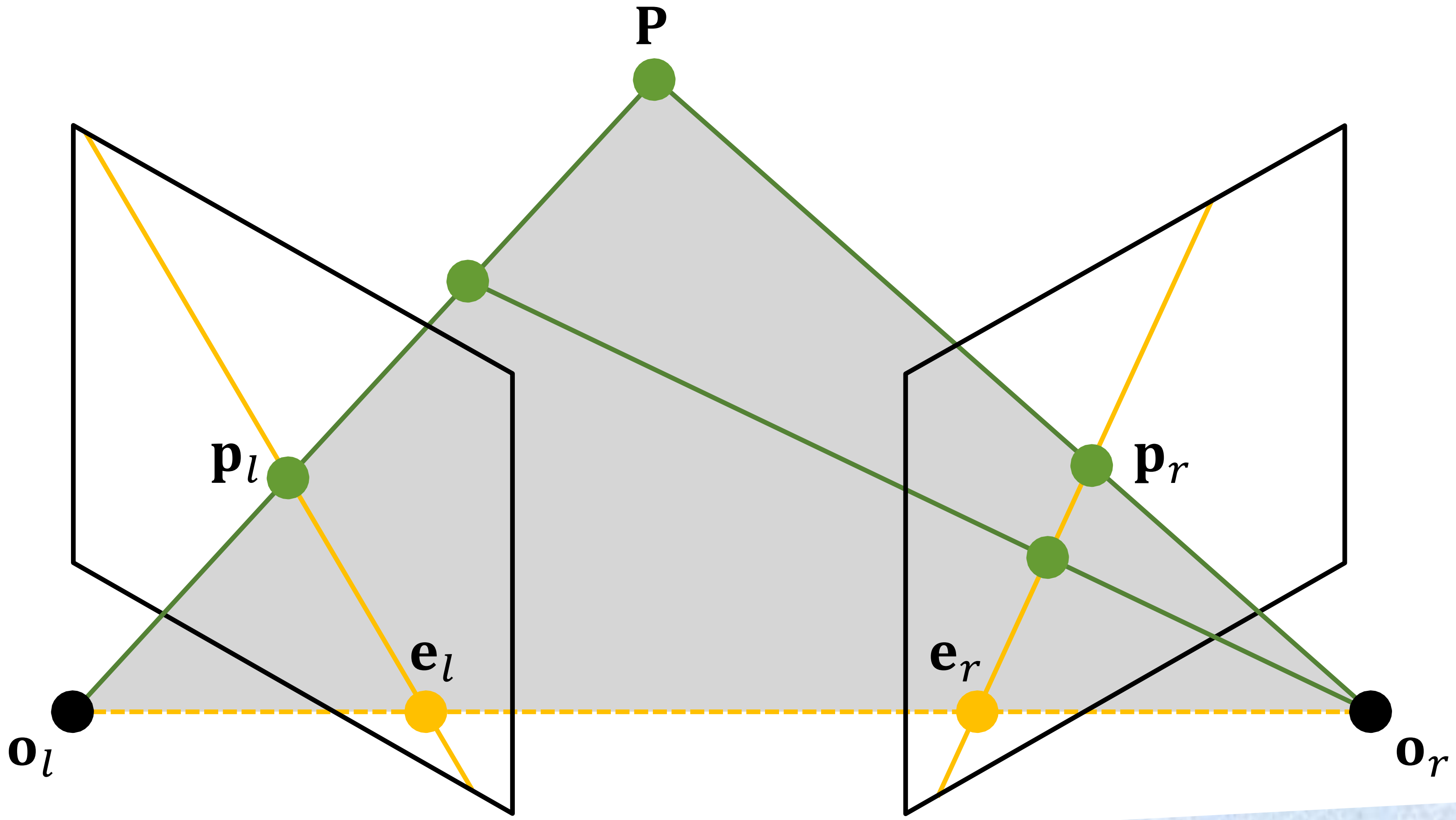

P
$\mathbf{p}_l$
$\mathbf{p}_r$
共轭极线对
$\mathbf{e}_l$
$\mathbf{e}_r$
$\mathbf{o}_l$
$\mathbf{o}_r$



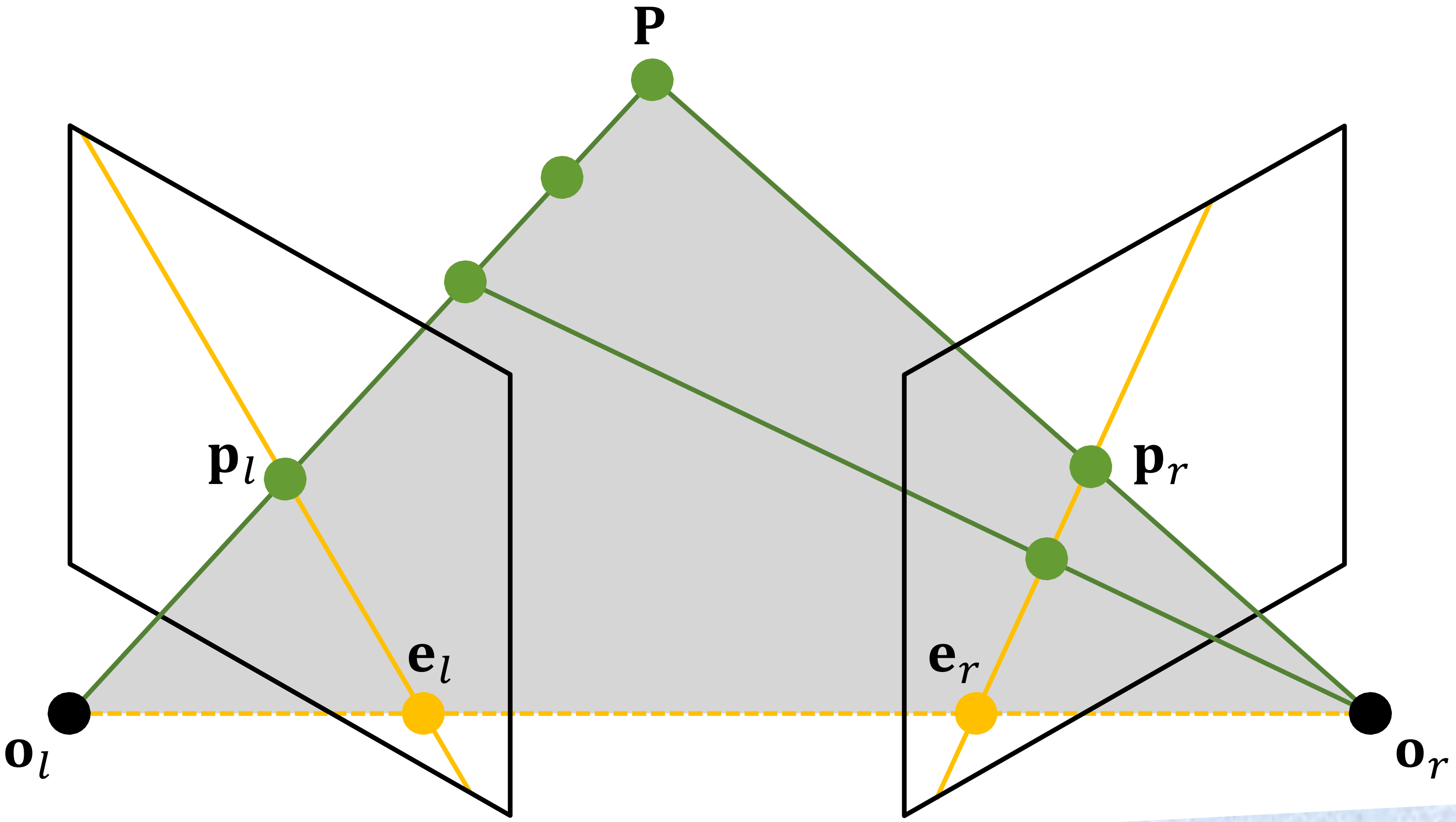
给定$\mathbf{p}_l$，它在右图中的投影$\mathbf{p}_r$在哪里?



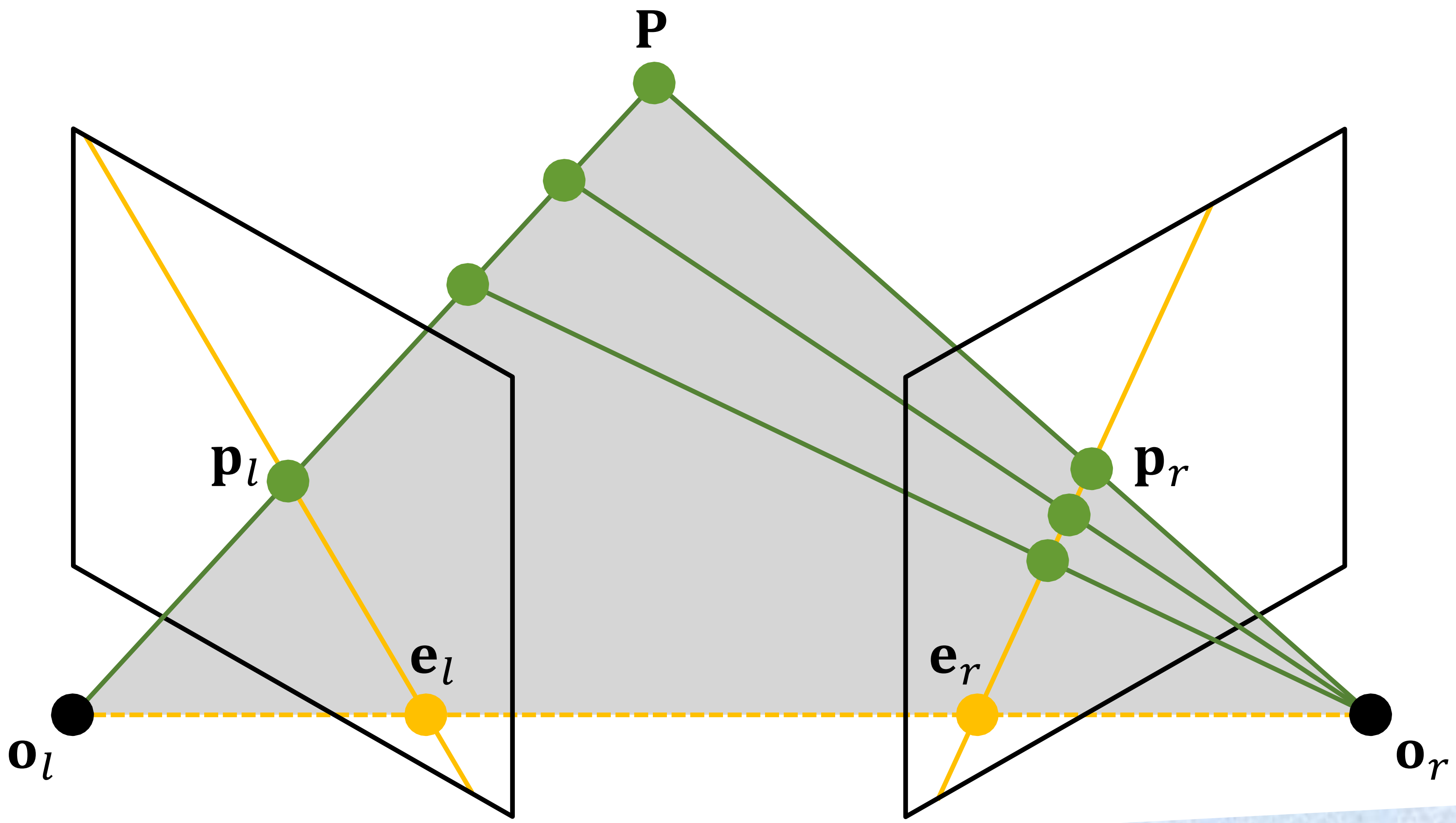
给定$\mathbf{p}_l$，它在右图中的投影$\mathbf{p}_r$在哪里？



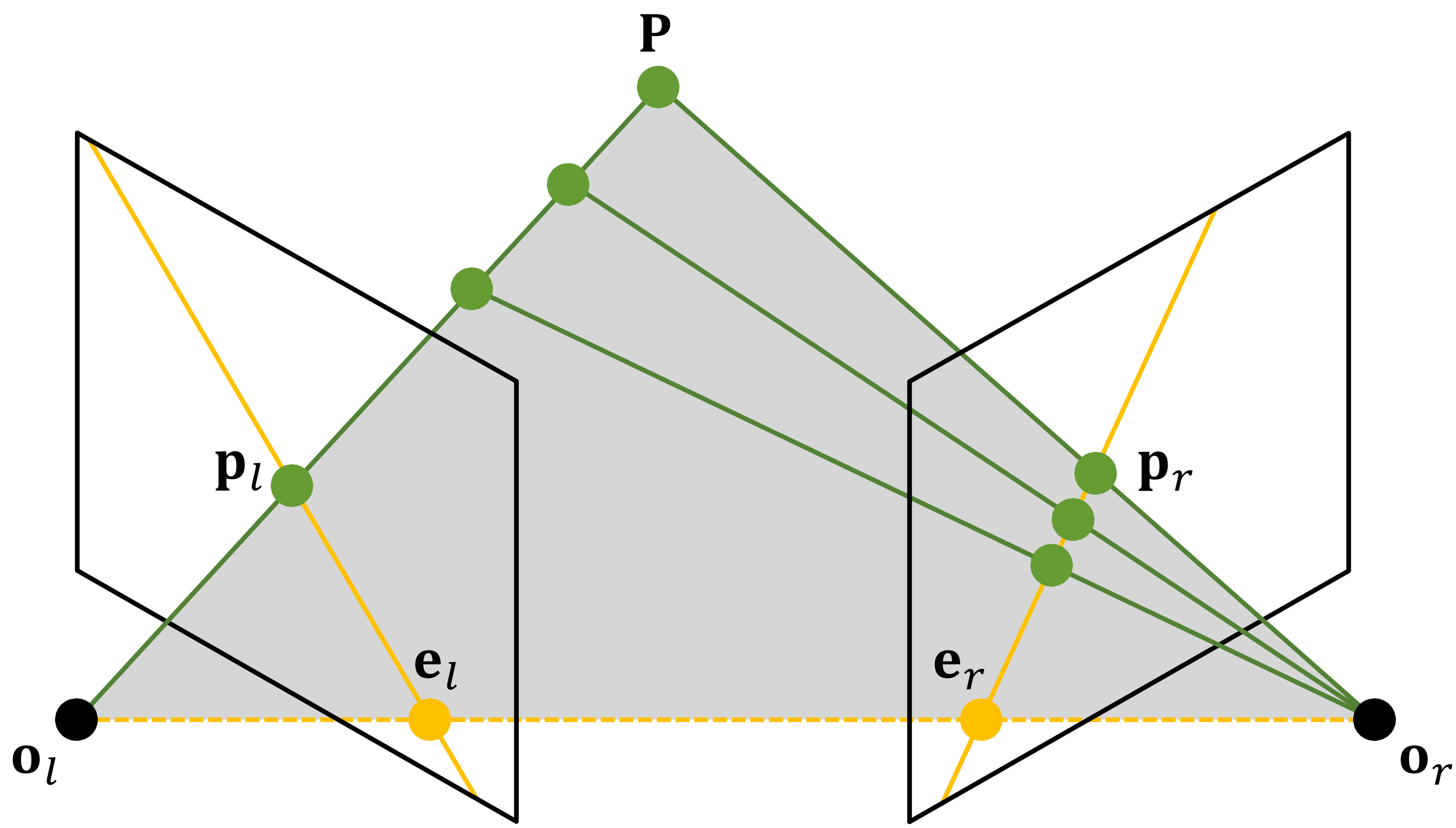
给定$\mathbf{p}_l$，它在右图中的投影$\mathbf{p}_r$在哪里？



给定$\mathbf{p}_l$，它在右图中的投影$\mathbf{p}_r$在哪里？

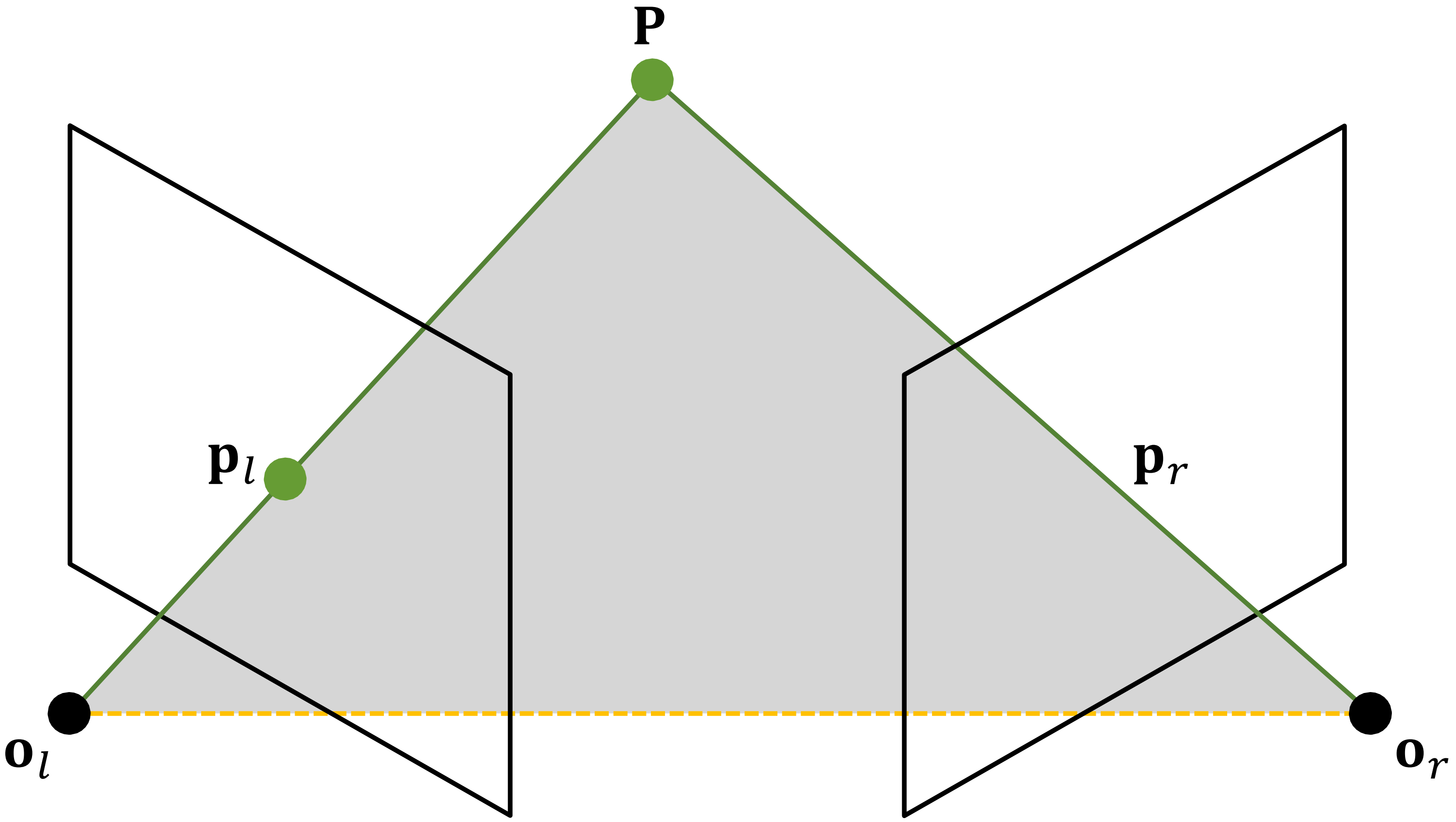


给定$\mathbf{p}_l$，它在右图中的投影$\mathbf{p}_r$在哪里？

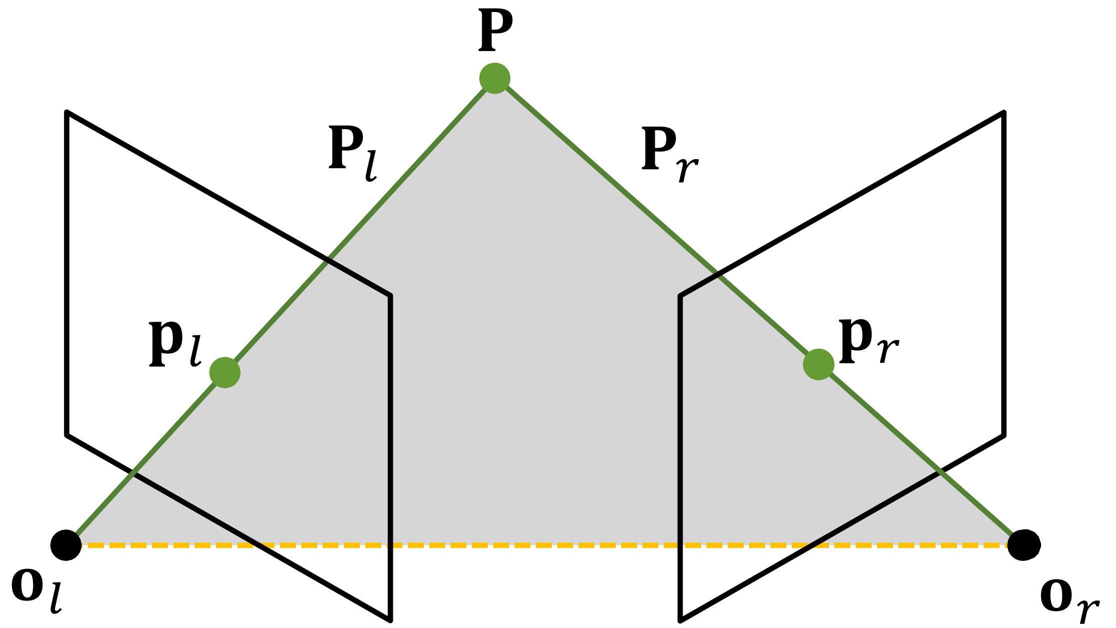


极线约束

对应点必须位于共轭的极线对上

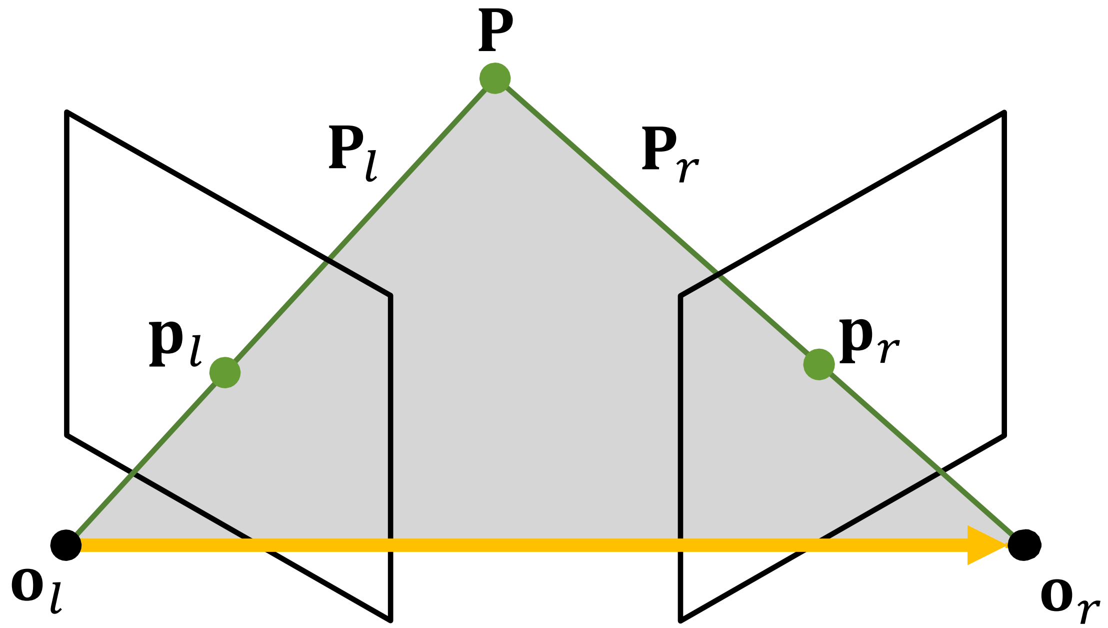


P
p_l
p_r
o_l
o_r



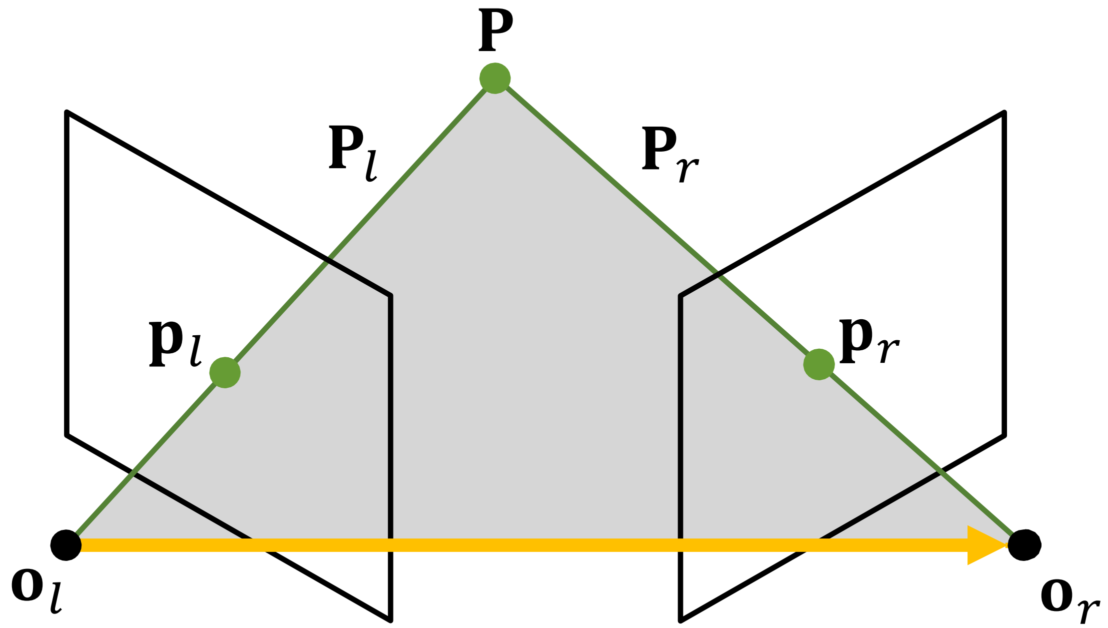
令:

$\mathbf{T} = \boldsymbol{O}_r - \boldsymbol{O}_l$ 定义左相机中心移向右相机中心的平移



令:

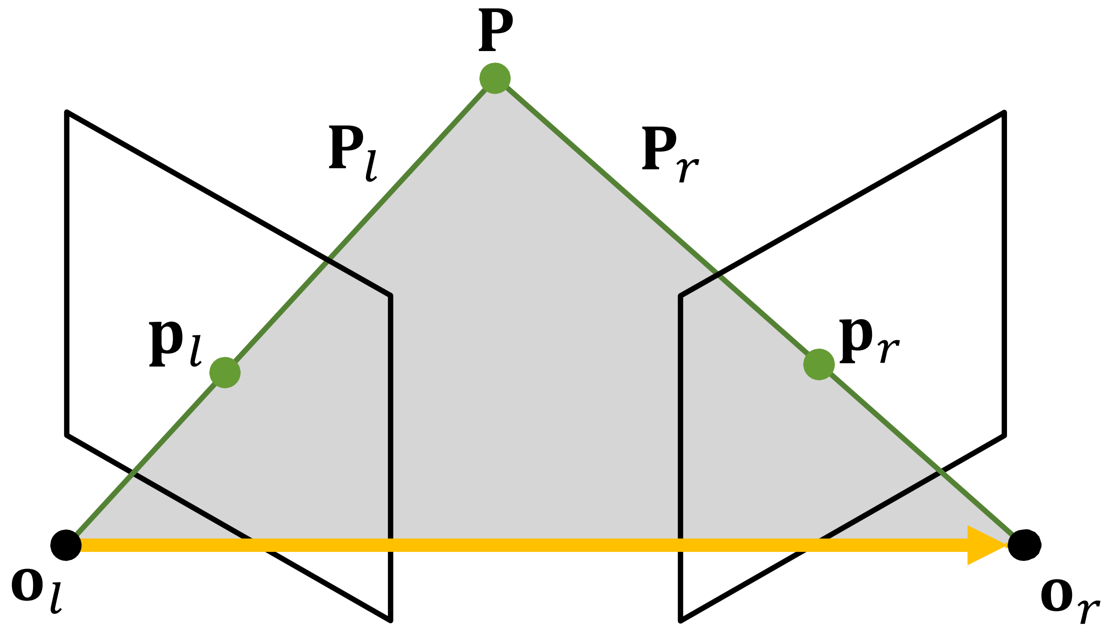
$\mathbf{T} = \boldsymbol{O}_r - \boldsymbol{O}_l$ **定义左相机中心移向右相机中心的平移**



令:

$\mathbf{T} = O_r - O_l$ 定义左相机中心移向右相机中心的平移

R 定义对齐左、右坐标轴的旋转矩阵



令:

$\mathrm{T} = O_r - O_l$ 定义左相机中心移向右相机中心的平移

R 定义对齐左、右坐标轴的旋转矩阵

$\mathbf{P}_r = \mathbf{R}(\mathbf{P}_l - \mathbf{T})$ 定义P在左、右两个视点中的坐标之间的变换

回顾： 线性代数

定义： 一个$m \times n$矩阵$\mathbf{A}$的**列秩**是它的线性无关的纵列的数目。类似地，**行秩**是$\mathbf{A}$的线性无关的横行的数目。

定义： 一个$m \times n$矩阵$\mathbf{A}$的**列秩**是它的线性无关的纵列的数目。类似地，**行秩**是$\mathbf{A}$的线性无关的横行的数目。

矩阵的列秩和行秩总是相等的

定义： 一个$m \times n$矩阵**A**的**列秩**是它的线性无关的纵列的数目。类似地，**行秩**是**A**的线性无关的横行的数目。

矩阵的列秩和行秩总是相等的

$$\begin{pmatrix} 1 & 1 \\ 2 & 2 \end{pmatrix}$$

定义：一个$m \times n$矩阵**A**的列秩是它的线性无关的纵列的数目。类似地，行秩是**A**的线性无关的横行的数目。

矩阵的列秩和行秩总是相等的

$$\begin{pmatrix} 1 & 1 \\ 2 & 2 \end{pmatrix}$$

该矩阵的秩是多少？

定义：一个$m \times n$矩阵A的**零空间**是齐次方程组

$$\mathbf{Ax} = \mathbf{0}$$

所有解的集合。

定义：一个$m \times n$矩阵**A**的**零空间**是齐次方程组

$$\mathbf{Ax} = \mathbf{0}$$

所有解的集合。

$$\mathrm{Null}(\mathbf{A}) = \{\mathbf{x}: \mathbf{x} \in \mathbb{R}^n \text{ and } \mathbf{Ax} = \mathbf{0}\}$$

定义： 一个$m \times n$矩阵$\mathbf{A}$的**零空间**是齐次方程组

$$\mathbf{Ax} = \mathbf{0}$$

所有解的集合。

$$\text{Null}(\mathbf{A}) = \{\mathbf{x}: \mathbf{x} \in \mathbb{R}^n \text{ and } \mathbf{Ax} = \mathbf{0}\}$$

矩阵的秩和零化度之和等于矩阵的列数n

定义：三维空间中两个向量$\mathbf{a}$和$\mathbf{b}$的**叉积**$\mathbf{a} \times \mathbf{b}$是与$\mathbf{a}$和$\mathbf{b}$都垂直的向量，可以定义为：

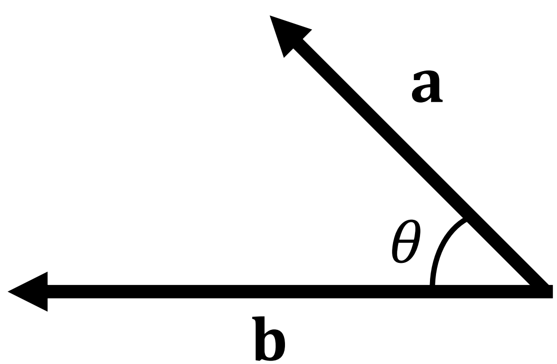
$$\mathbf{a} \times \mathbf{b} = \|\mathbf{a}\| \|\mathbf{b}\| \sin(\theta)\, \mathbf{n}$$

其中θ表示$\mathbf{a}$和$\mathbf{b}$的夹角（$0 \leq \theta \leq \pi$）。$\|\mathbf{a}\|$和$\|\mathbf{b}\|$是向量$\mathbf{a}$和$\mathbf{b}$的模长，而$\mathbf{n}$则是一个与$\mathbf{a}$、$\mathbf{b}$所构成的平面垂直的单位向量，方向由右手定则决定。

定义：三维空间中两个向量**a**和**b**的叉积$\mathbf{a} \times \mathbf{b}$是与**a**和**b**都垂直的向量，可以定义为：

$$\mathbf{a} \times \mathbf{b} = \|\mathbf{a}\| \|\mathbf{b}\| \sin(\theta)\, \mathbf{n}$$

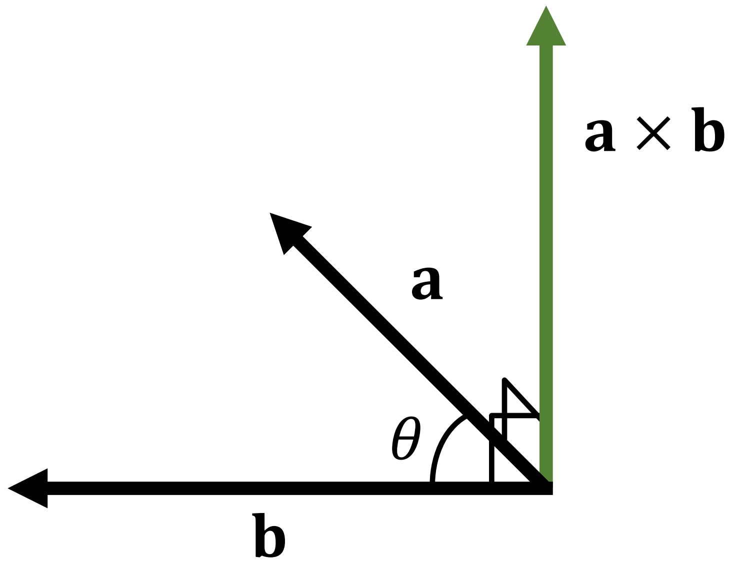
其中θ表示**a**和**b**的夹角（$0 \le \theta \le \pi$）。$\|\mathbf{a}\|$和$\|\mathbf{b}\|$是向量**a**和**b**的模长，而**n**则是一个与**a**、**b**所构成的平面垂直的单位向量，方向由右手定则决定。



定义：三维空间中两个向量$\mathbf{a}$和$\mathbf{b}$的**叉积**$\mathbf{a} \times \mathbf{b}$是与$\mathbf{a}$和$\mathbf{b}$都垂直的向量，可以定义为：

$$\mathbf{a} \times \mathbf{b} = \|\mathbf{a}\| \|\mathbf{b}\| \sin(\theta)\, \mathbf{n}$$

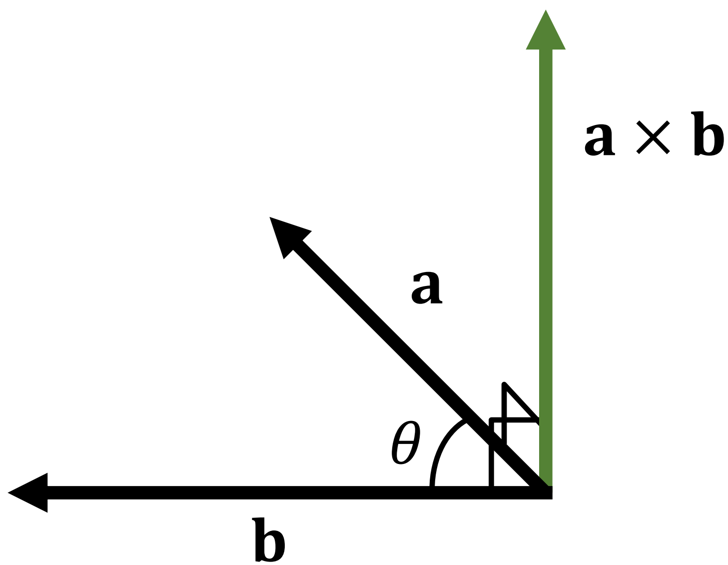
其中θ表示$\mathbf{a}$和$\mathbf{b}$的夹角（$0 \le \theta \le \pi$）。$\|\mathbf{a}\|$和$\|\mathbf{b}\|$是向量$\mathbf{a}$和$\mathbf{b}$的模长，而$\mathbf{n}$则是一个与$\mathbf{a}$、$\mathbf{b}$所构成的平面垂直的单位向量，方向由右手定则决定。

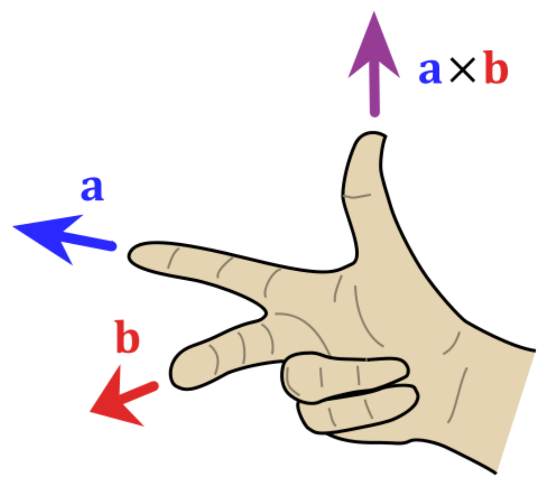


定义：三维空间中两个向量**a**和**b**的**叉积** $\mathbf{a} \times \mathbf{b}$ 是与**a**和**b**都垂直的向量，可以定义为：

$$\mathbf{a} \times \mathbf{b} = \|\mathbf{a}\| \|\mathbf{b}\| \sin(\theta)\, \mathbf{n}$$

其中θ表示**a**和**b**的夹角（$0 \le \theta \le \pi$）。$\|\mathbf{a}\|$和$\|\mathbf{b}\|$是向量**a**和**b**的模长，而**n**则是一个与**a**、**b**所构成的平面垂直的单位向量，方向由右手定则决定。





给定3×1**向量**$\mathbf{a} = (a_1, a_2, a_3)^{\mathrm{T}}$**和**$\mathbf{b} = (b_1, b_2, b_3)^{\mathrm{T}}$

$$\mathbf{a} \times \mathbf{b} =$$

给定3×1**向量**$\mathbf{a}=(a_1,a_2,a_3)^{\mathrm{T}}$**和**$\mathbf{b}=(b_1,b_2,b_3)^{\mathrm{T}}$

$$\mathbf{a}\times\mathbf{b}=\begin{pmatrix}-a_3b_2+a_2b_3\\ a_3b_1-a_1b_3\\ -a_2b_1+a_1b_2\end{pmatrix}$$

给定3×1**向量**$\mathbf{a}=(a_1,a_2,a_3)^{\mathrm{T}}$**和**$\mathbf{b}=(b_1,b_2,b_3)^{\mathrm{T}}$

$$\mathbf{a}\times\mathbf{b}=\begin{pmatrix}-a_3b_2+a_2b_3\\ a_3b_1-a_1b_3\\ -a_2b_1+a_1b_2\end{pmatrix}$$

$$=\begin{pmatrix}0 & -a_3 & a_2\\ a_3 & 0 & -a_1\\ -a_2 & a_1 & 0\end{pmatrix}\begin{pmatrix}b_1\\ b_2\\ b_3\end{pmatrix}$$

给定3×1**向量**$\mathbf{a}=(a_1,a_2,a_3)^{\mathrm{T}}$**和**$\mathbf{b}=(b_1,b_2,b_3)^{\mathrm{T}}$

$$\mathbf{a}\times\mathbf{b}=\begin{pmatrix}-a_3b_2+a_2b_3\\ a_3b_1-a_1b_3\\ -a_2b_1+a_1b_2\end{pmatrix}$$
$$=\begin{pmatrix}0&-a_3&a_2\\ a_3&0&-a_1\\ -a_2&a_1&0\end{pmatrix}\begin{pmatrix}b_1\\ b_2\\ b_3\end{pmatrix}$$

矩阵乘法

给定3×1向量$\mathbf{a} = (a_1, a_2, a_3)^{\mathrm{T}}$和$\mathbf{b} = (b_1, b_2, b_3)^{\mathrm{T}}$

$$\mathbf{a} \times \mathbf{b} = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

给定3×1向量$\mathbf{a} = (a_1, a_2, a_3)^{\mathrm{T}}$和$\mathbf{b} = (b_1, b_2, b_3)^{\mathrm{T}}$

$$\mathbf{a} \times \mathbf{b} = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

反对称矩阵

$$\boldsymbol{S} = -\boldsymbol{S}^{\mathrm{T}}$$

给定3×1向量$\mathbf{a} = (a_1, a_2, a_3)^{\mathrm{T}}$和$\mathbf{b} = (b_1, b_2, b_3)^{\mathrm{T}}$

$$\mathbf{a} \times \mathbf{b} = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

反对称矩阵

$\boldsymbol{S} = -\boldsymbol{S}^{\mathrm{T}}$

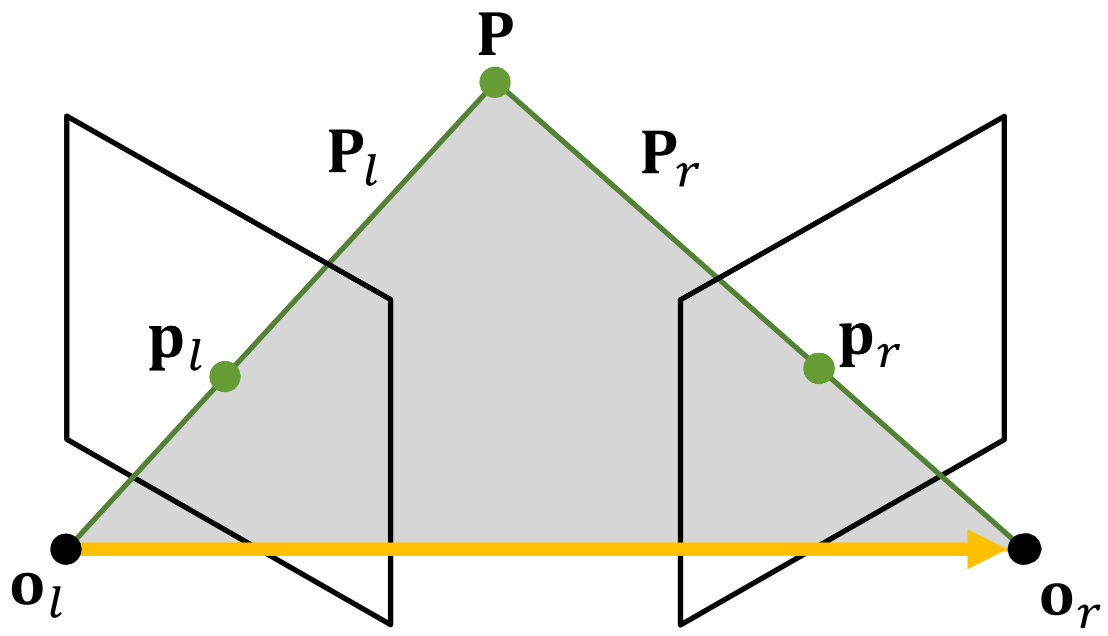
秩为2

给定3×1向量$\mathbf{a} = (a_1, a_2, a_3)^{\mathrm{T}}$和$\mathbf{b} = (b_1, b_2, b_3)^{\mathrm{T}}$

$$\mathbf{a} \times \mathbf{b} = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

$$= [\mathbf{a}_\times]\mathbf{b}$$

回顾：线性代数

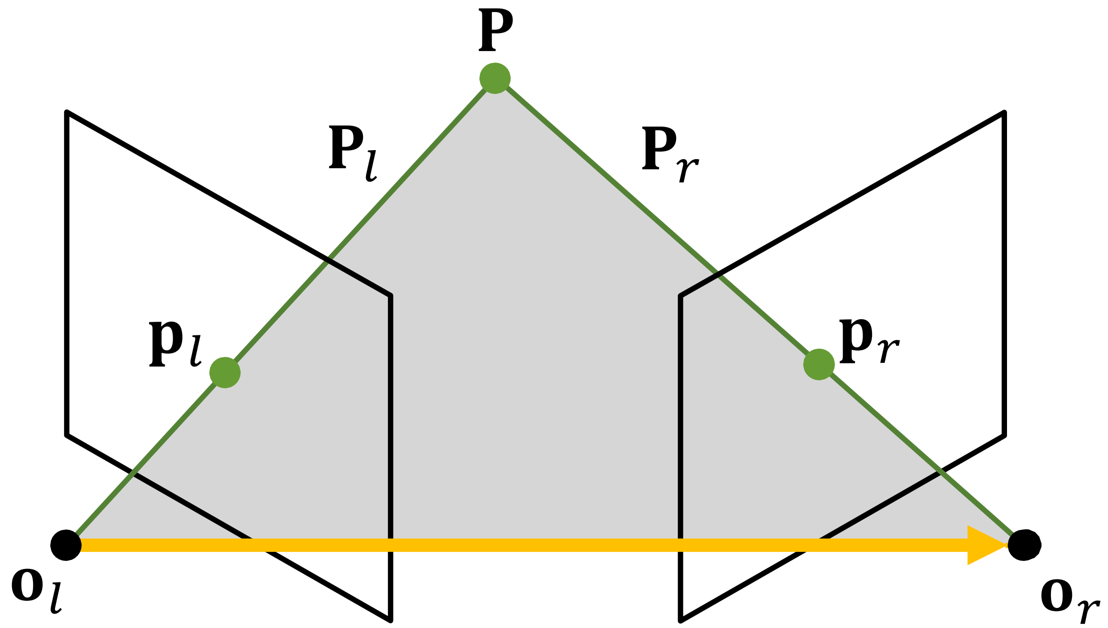


令:

$\mathrm{T} = O_r - O_l$ 定义左相机中心移向右相机中心的平移

R 定义对齐左、右坐标轴的旋转矩阵

$\mathbf{P}_r = \mathbf{R}(\mathbf{P}_l - \mathbf{T})$ **定义P在左、右两个视点中的坐标之间的变换**

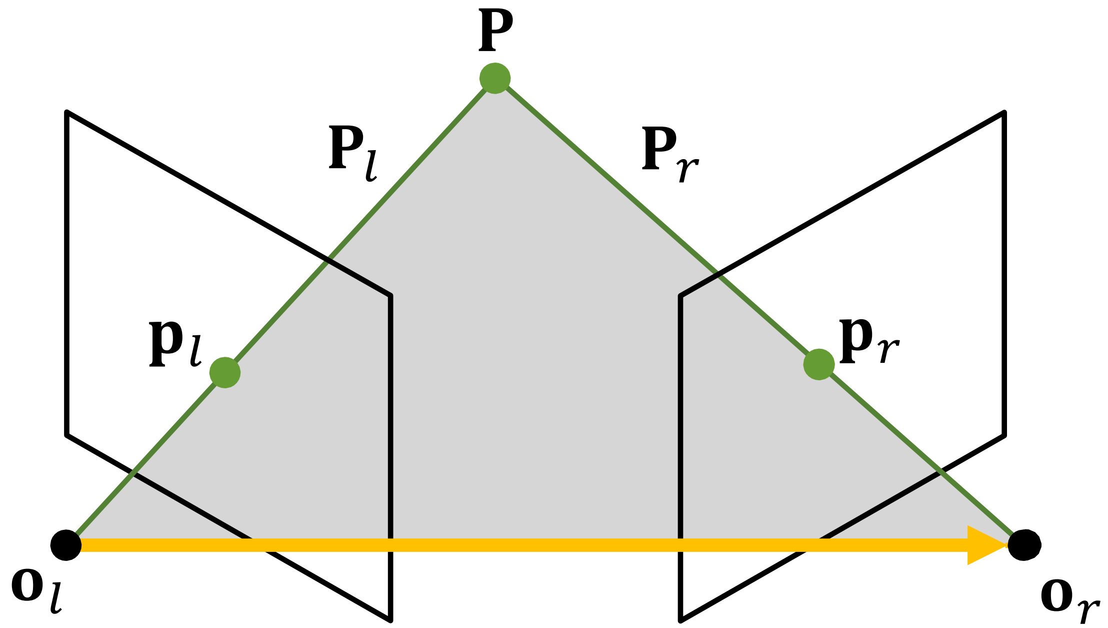


令:

$\mathbf{P}_r = \mathbf{R}(\mathbf{P}_l - \mathbf{T})$ 定义P在左、右两个视点中的坐标之间的变换

共面条件:

$$(\mathbf{P}_l - \mathbf{T}) \cdot (\mathbf{T} \times \mathbf{P}_l) = 0$$



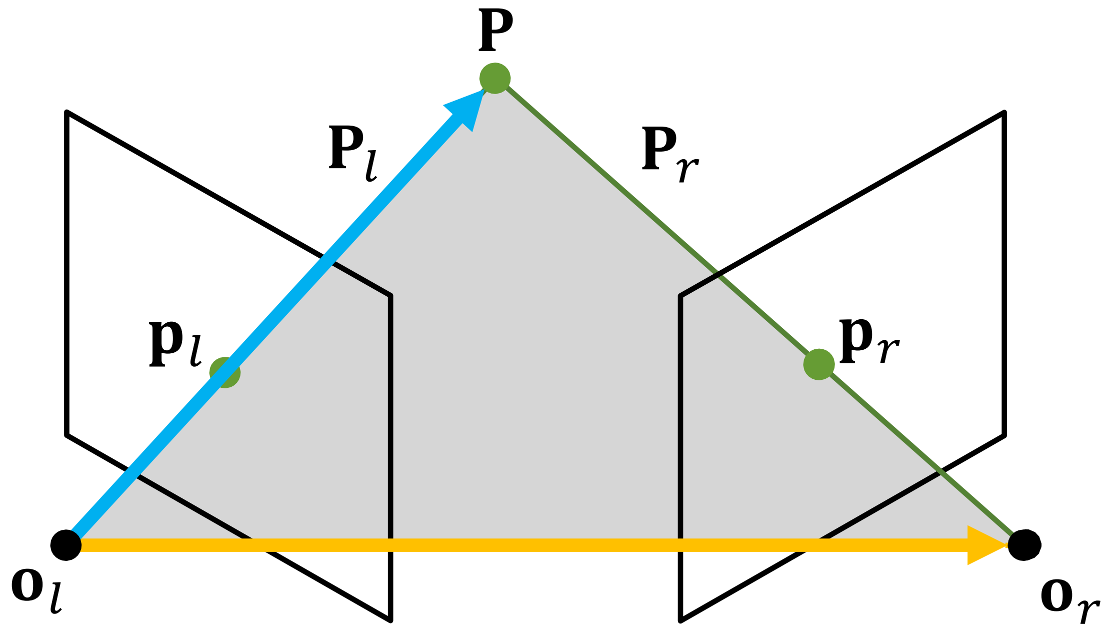
令:

$\mathbf{P}_r = \mathbf{R}(\mathbf{P}_l - \mathbf{T})$ 定义P在左、右两个视点中的坐标之间的变换

共面条件:

$$(\mathbf{P}_l - \mathbf{T}) \cdot (\mathbf{T} \times \mathbf{P}_l) = 0$$

垂直于平面



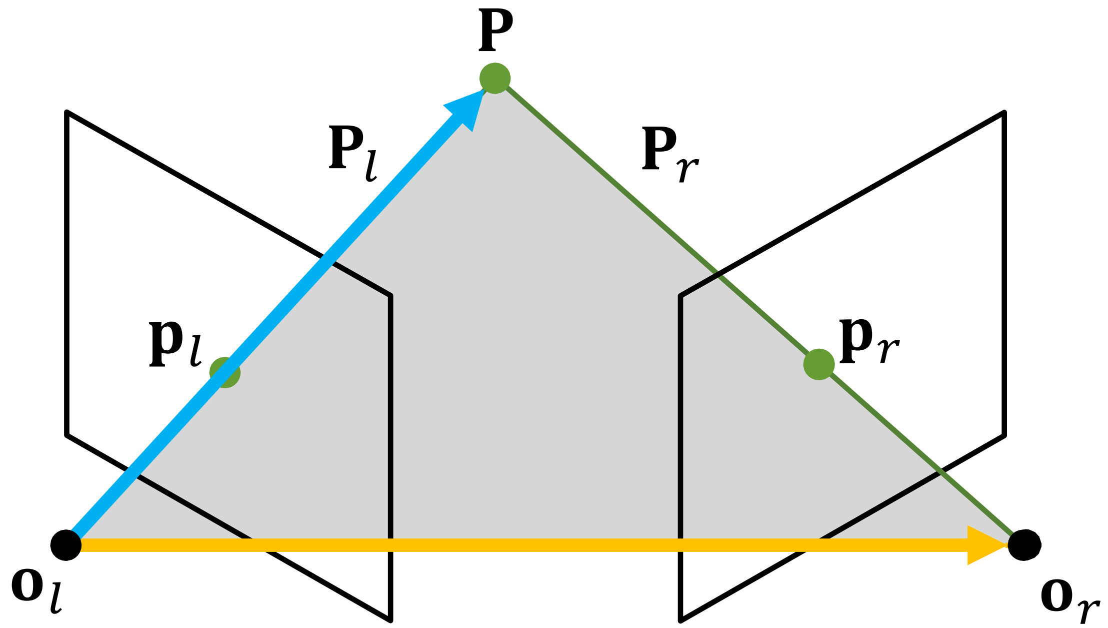
令:

$\mathbf{P}_r = \mathbf{R}(\mathbf{P}_l - \mathbf{T})$ 定义P在左、右两个视点中的坐标之间的变换

共面条件:

$$(\mathbf{P}_l - \mathbf{T}) \cdot (\mathbf{T} \times \mathbf{P}_l) = 0$$

垂直于平面

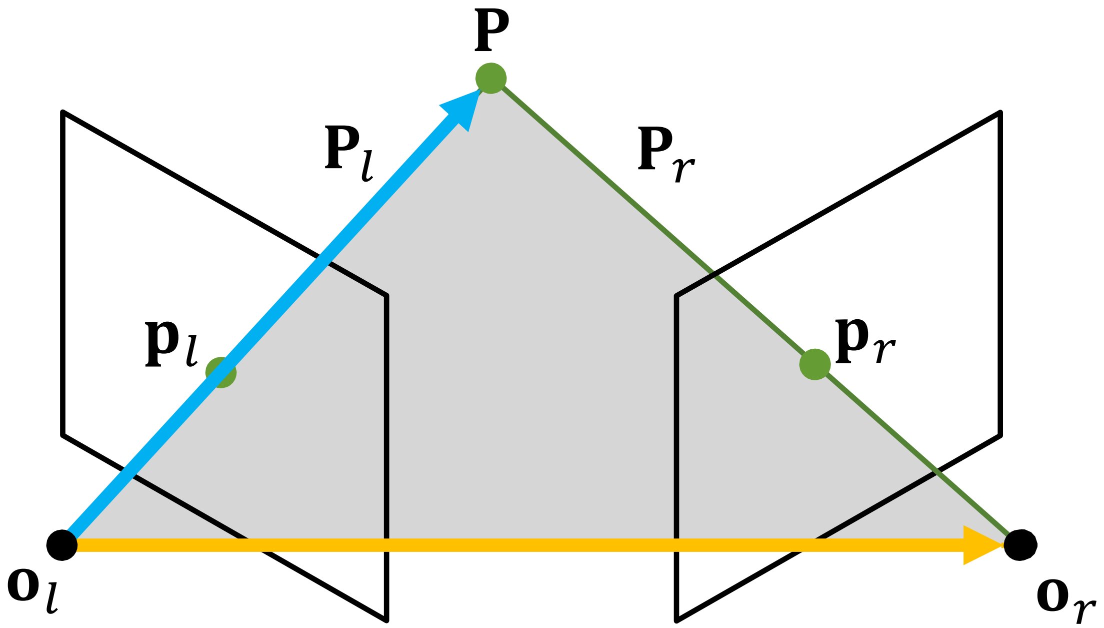


令:

$\mathbf{P}_r = \mathbf{R}(\mathbf{P}_l - \mathbf{T})$ **定义P在左、右两个视点中的坐标之间的变换**

共面条件:

$$(\mathbf{P}_l - \mathbf{T}) \cdot (\mathbf{T} \times \mathbf{P}_l) = 0$$



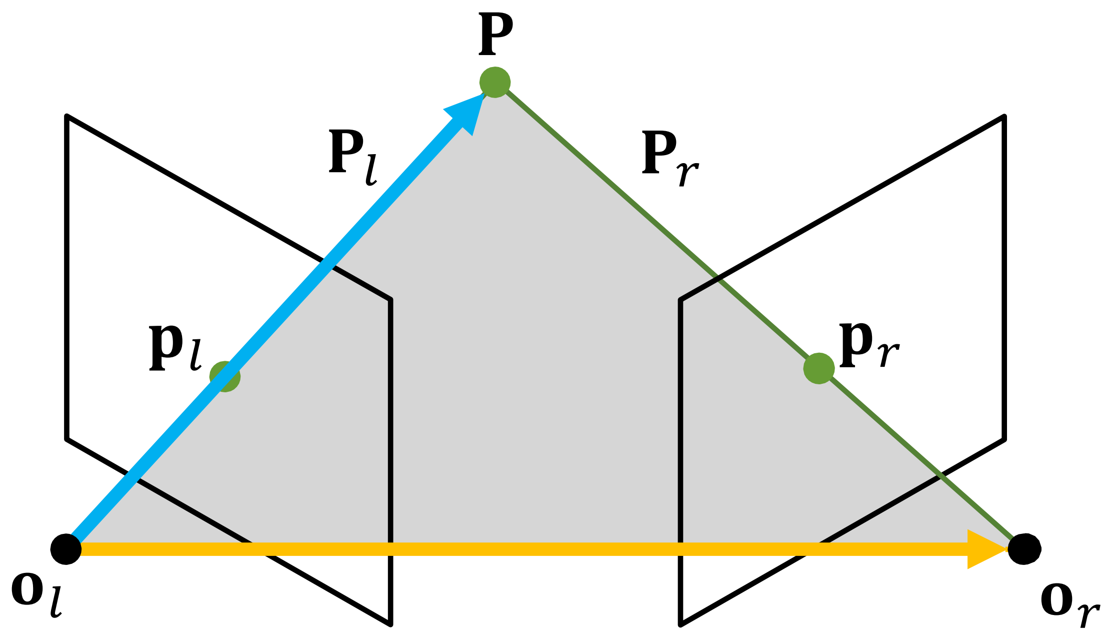
令:

$\mathbf{P}_r = \mathbf{R}(\mathbf{P}_l - \mathbf{T})$ 定义P在左、右两个视点中的坐标之间的变换

共面条件:

$$(\mathbf{P}_l - \mathbf{T}) \cdot (\mathbf{T} \times \mathbf{P}_l) = 0$$

代入 $\mathbf{P}_l = \mathbf{R}^{\mathrm{T}}\mathbf{P}_r + \mathbf{T}$



令:

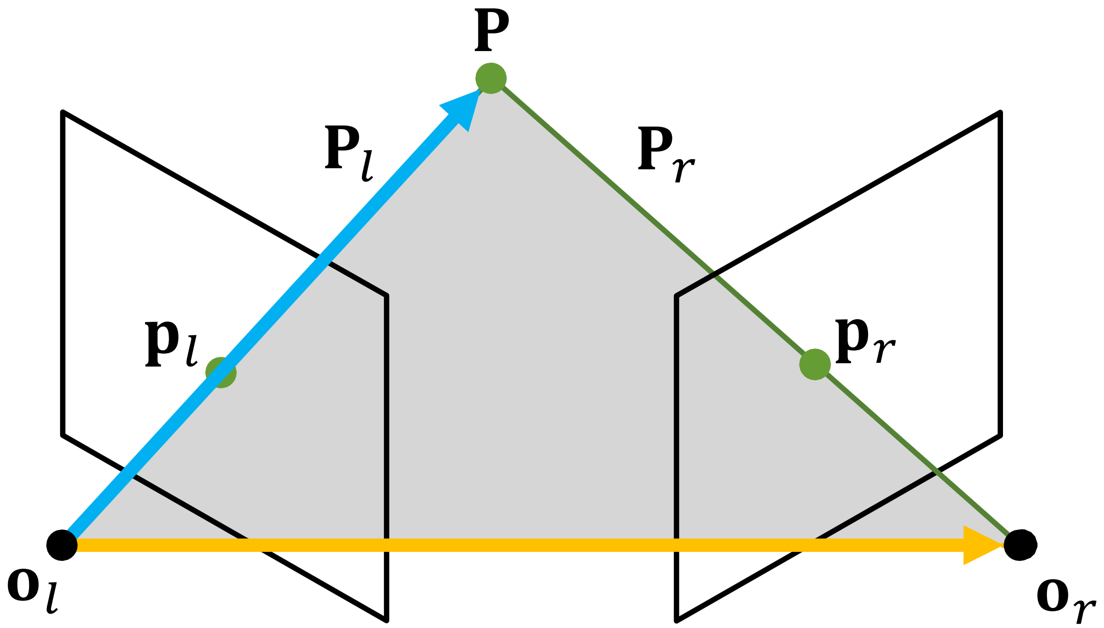
$\mathbf{P}_r = \mathbf{R}(\mathbf{P}_l - \mathbf{T})$ 定义P在左、右两个视点中的坐标之间的变换

共面条件:

$$(\mathbf{P}_l - \mathbf{T}) \cdot (\mathbf{T} \times \mathbf{P}_l) = 0$$

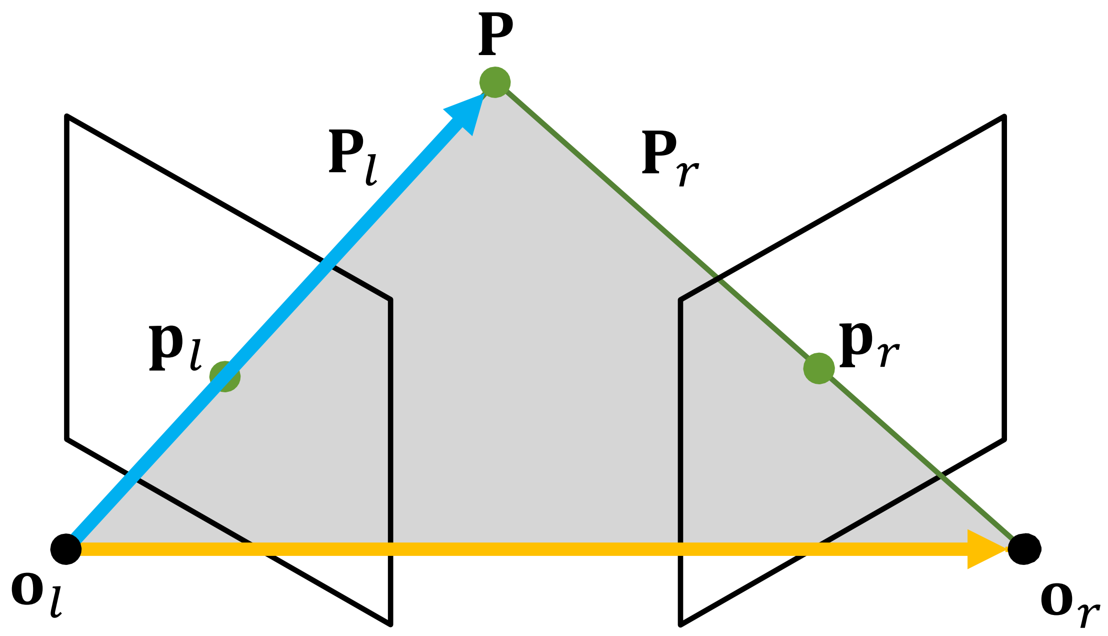
代入 $\mathbf{P}_l = \mathbf{R}^{\mathrm{T}}\mathbf{P}_r + \mathbf{T}$

$$\left(\mathbf{R}^{\mathrm{T}}\mathbf{P}_r\right) \cdot (\mathbf{T} \times \mathbf{P}_l) = 0$$



共面条件:

$$\left(\mathbf{R}^{\mathrm{T}}\mathbf{P}_r\right) \cdot (\mathbf{T} \times \mathbf{P}_l) = 0$$



共面条件:

$$\left(\mathbf{R}^{\mathrm{T}}\mathbf{P}_r\right)\cdot\left(\mathbf{T}\times\mathbf{P}_l\right)=0$$

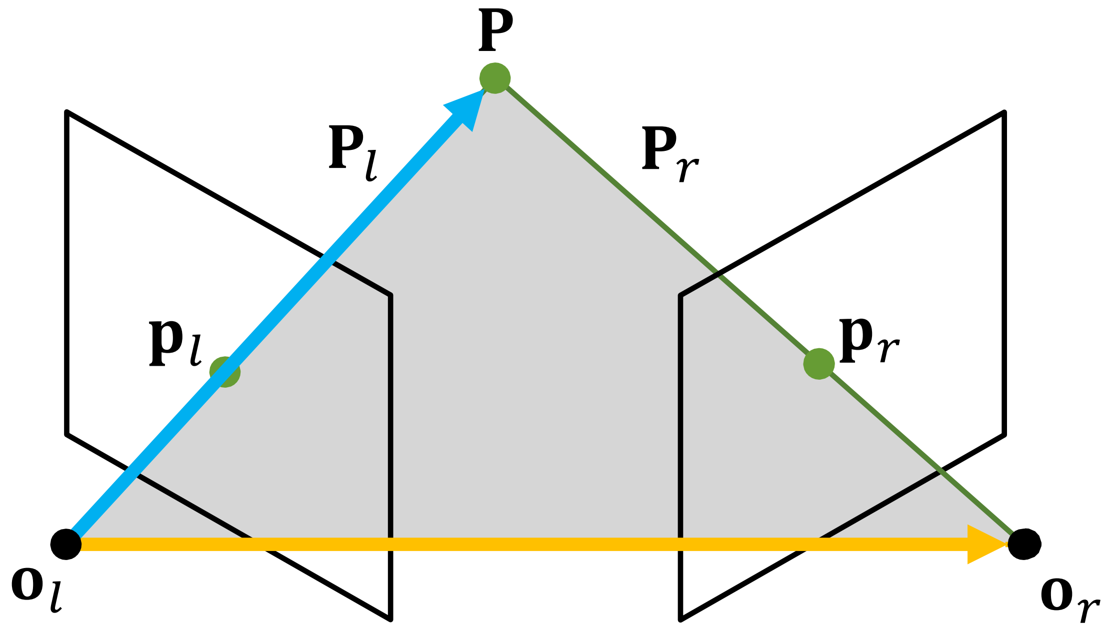
叉积写成矩阵乘法

给定3×1**向量**$\mathbf{a} = (a_1, a_2, a_3)^{\mathrm{T}}$**和**$\mathbf{b} = (b_1, b_2, b_3)^{\mathrm{T}}$

$$\mathbf{a} \times \mathbf{b} = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

$$= [\mathbf{a}_\times]\mathbf{b}$$

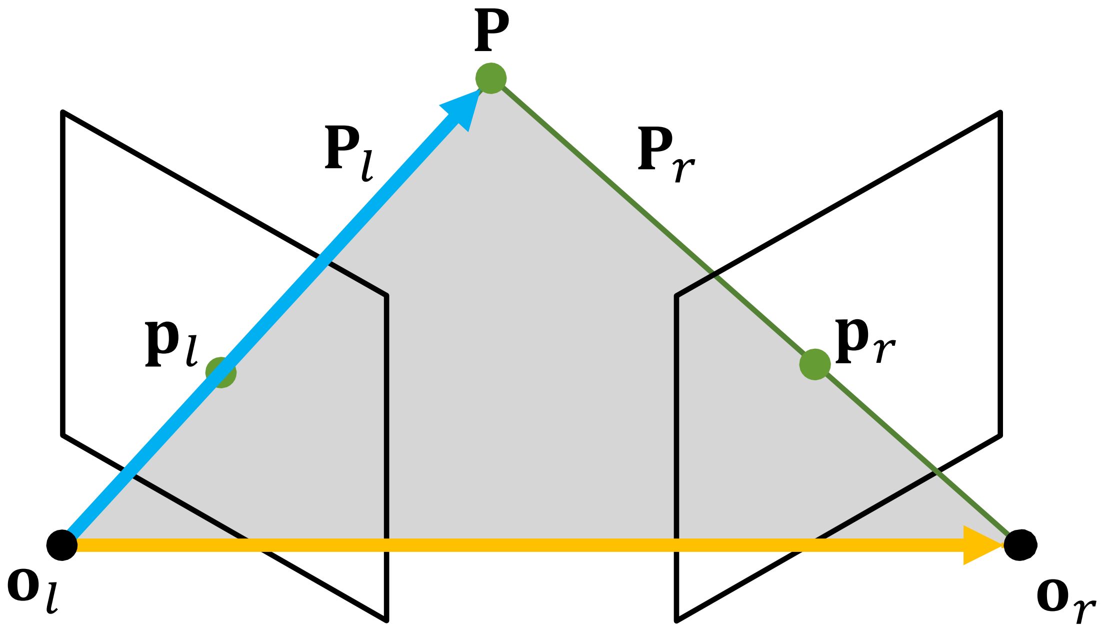
叉积写作矩阵乘法



共面条件:

$$\left(\mathbf{R}^{\mathrm{T}}\mathbf{P}_r\right) \cdot \left(\mathbf{T} \times \mathbf{P}_l\right) = 0$$

叉积写成矩阵乘法

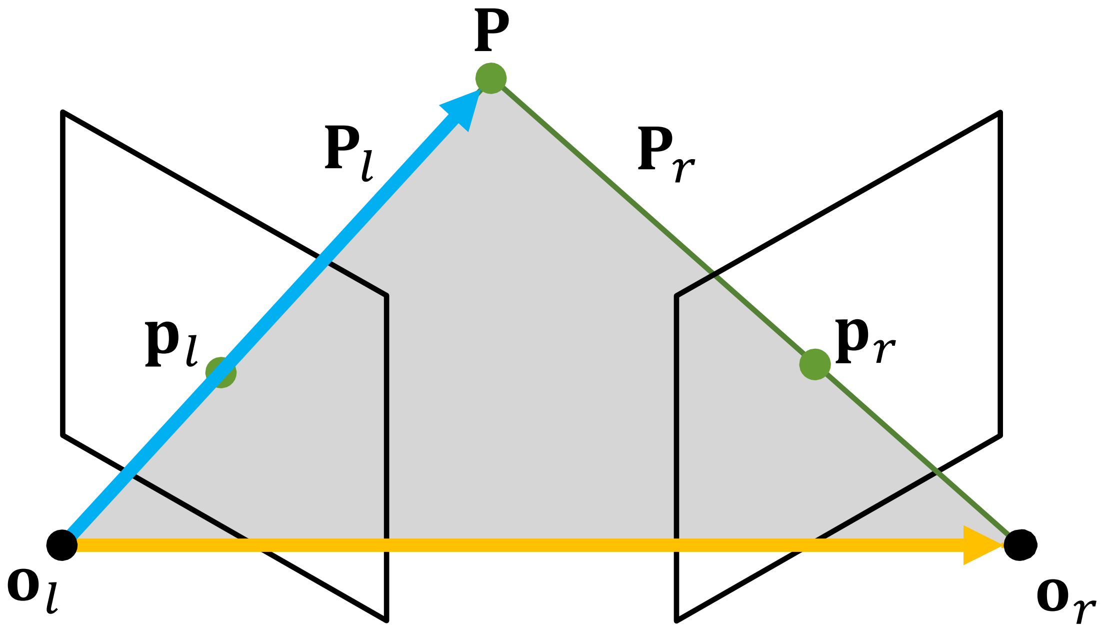


共面条件:

$$\left(\mathbf{R}^{\mathrm{T}}\mathbf{P}_r\right)\cdot(\mathbf{T}\times\mathbf{P}_l)=0$$

叉积写成矩阵乘法

$$\left(\mathbf{R}^{\mathrm{T}}\mathbf{P}_r\right)^{\mathrm{T}}[\mathbf{T}_\times]\mathbf{P}_l=0$$



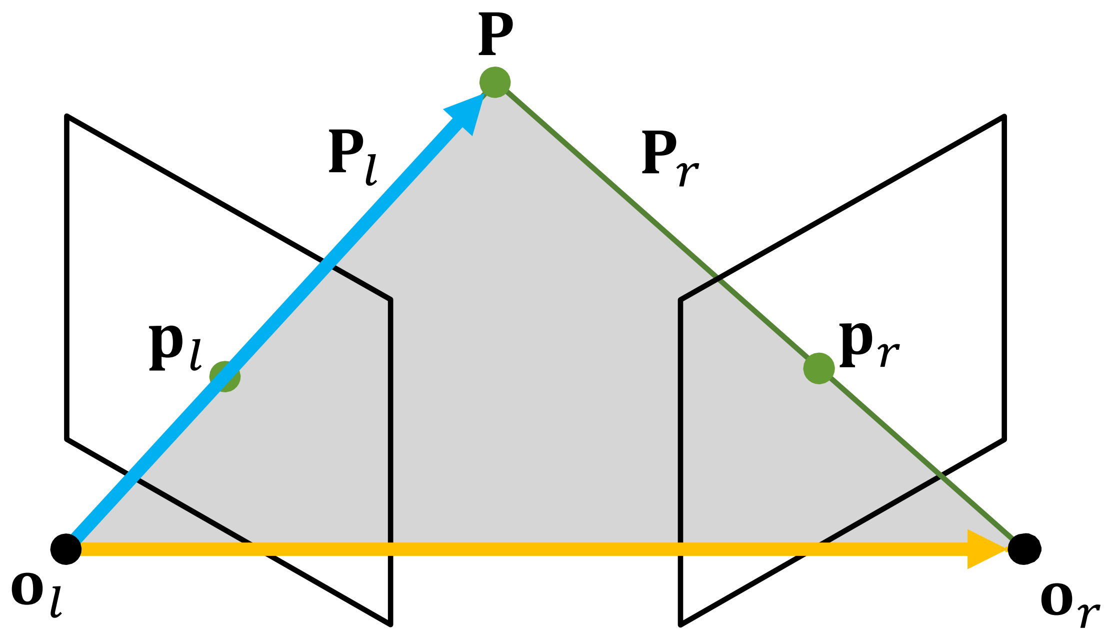
共面条件:

$$\left(\mathbf{R}^{\mathrm{T}}\mathbf{P}_r\right)\cdot(\mathbf{T}\times\mathbf{P}_l)=0$$

叉积写成矩阵乘法

$$\left(\mathbf{R}^{\mathrm{T}}\mathbf{P}_r\right)^{\mathrm{T}}[\mathbf{T}_\times]\mathbf{P}_l=0$$

改写



共面条件:

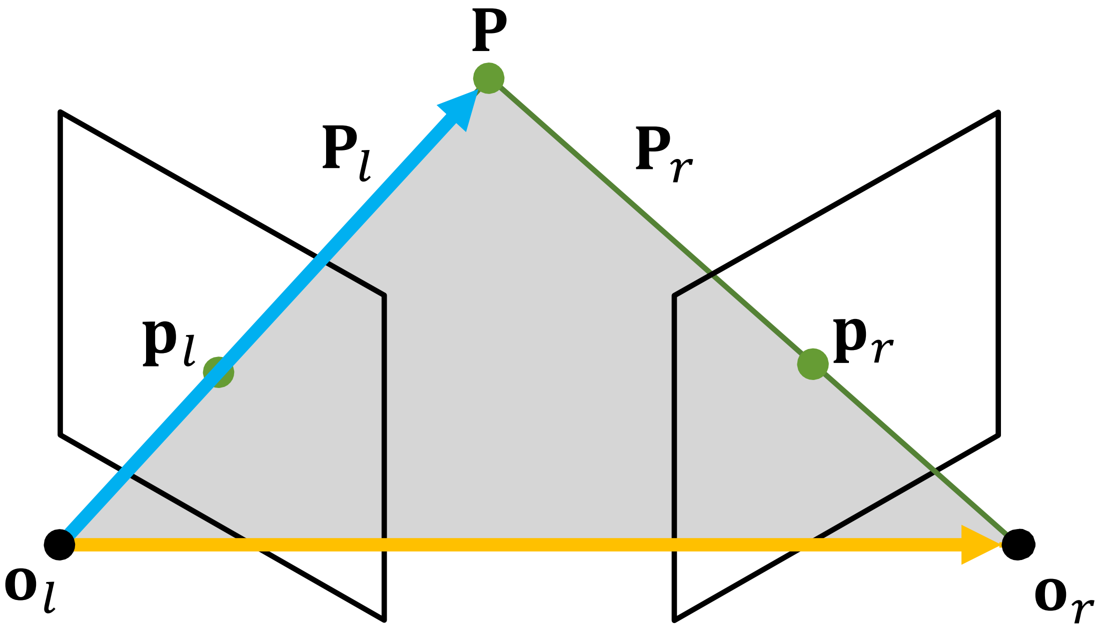
$$\left(\mathbf{R}^{\mathrm{T}}\mathbf{P}_r\right)\cdot(\mathbf{T}\times\mathbf{P}_l)=0$$

叉积写成矩阵乘法

$$\left(\mathbf{R}^{\mathrm{T}}\mathbf{P}_r\right)^{\mathrm{T}}[\mathbf{T}_\times]\mathbf{P}_l=0$$

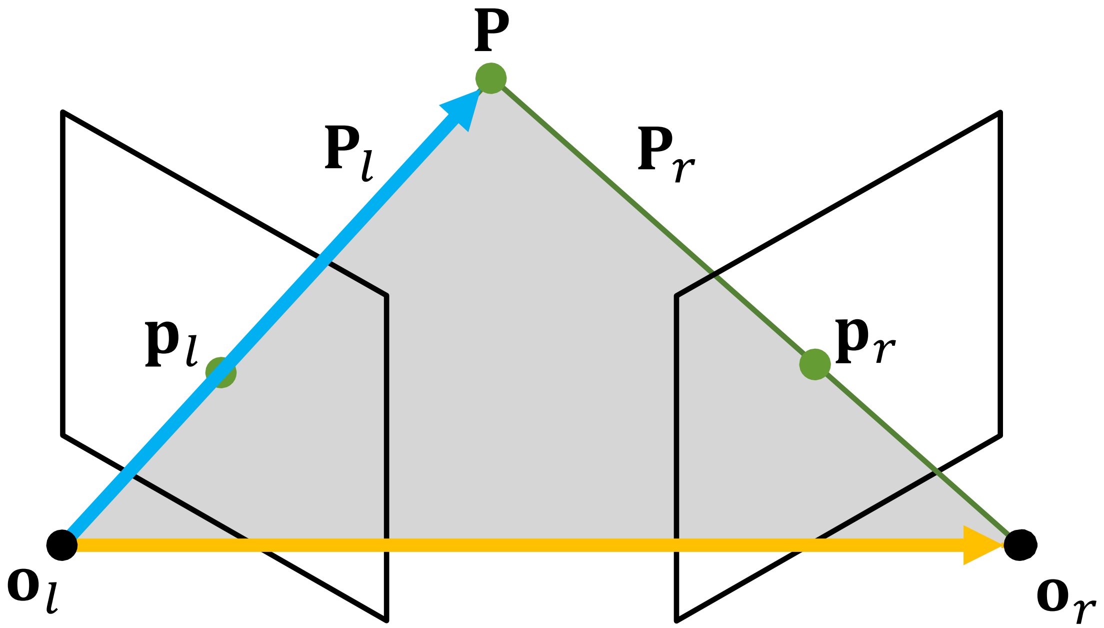
改写

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{R}[\mathbf{T}_\times]\mathbf{P}_l=0$$



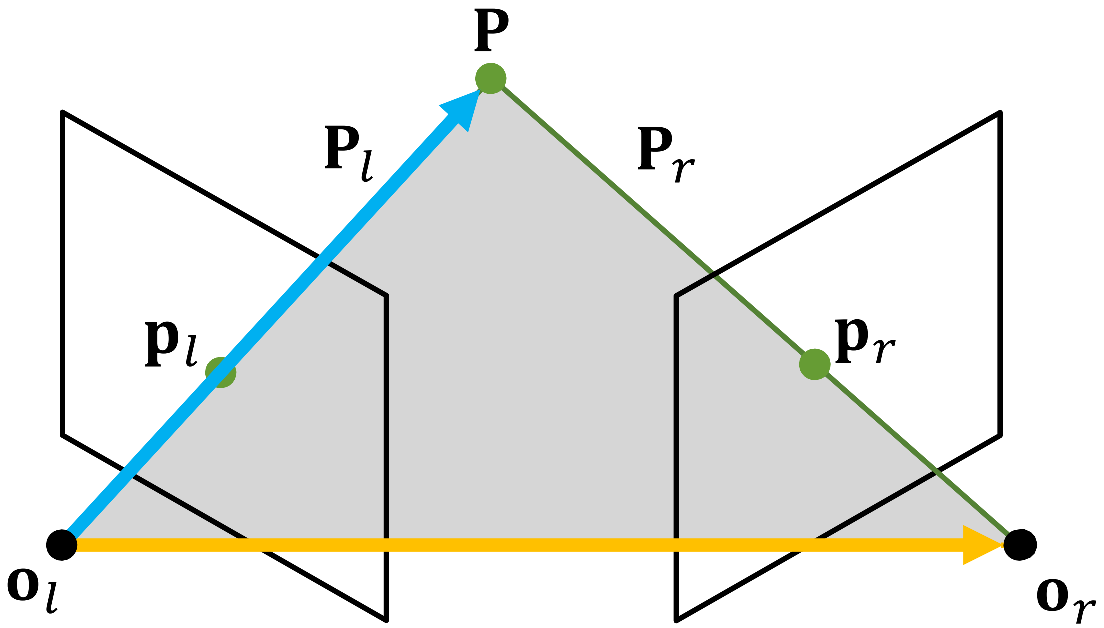
共面条件:

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{R}[\mathbf{T}_{\times}]\mathbf{P}_l = 0$$



共面条件:

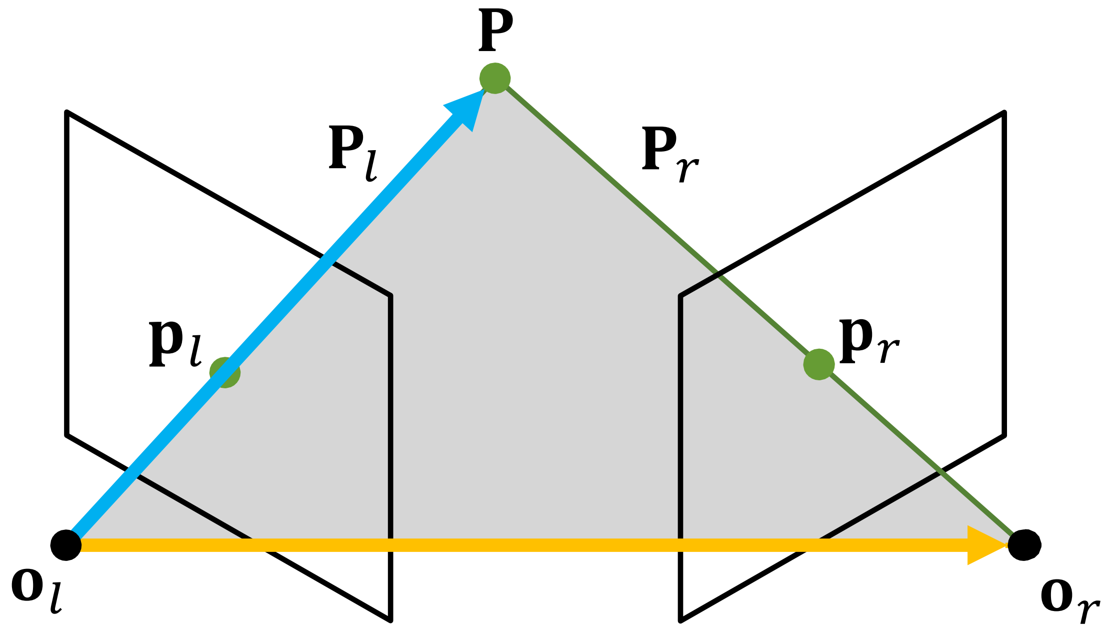
$$\mathbf{P}_r^{\mathrm{T}}\mathbf{R}[\mathbf{T}_{\times}]\mathbf{P}_l = 0$$



共面条件:

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{R}[\mathbf{T}_\times]\mathbf{P}_l = 0$$

令 $\mathbf{E} = \mathbf{R}[\mathbf{T}_\times]$

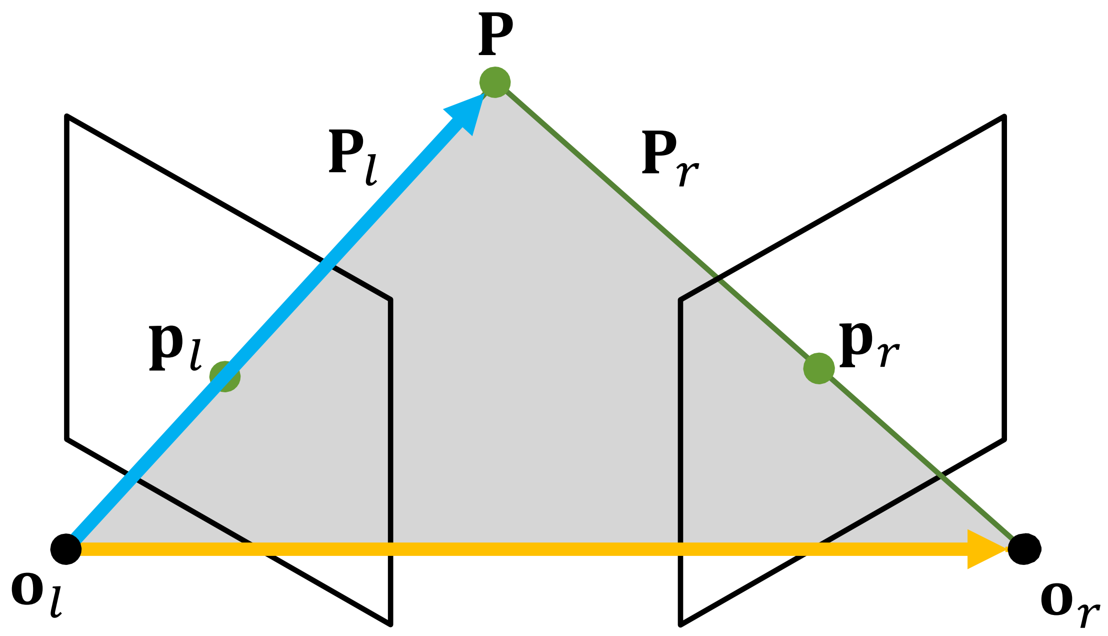


共面条件:

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{R}[\mathbf{T}_\times]\mathbf{P}_l = 0$$

令 $\mathbf{E} = \mathbf{R}[\mathbf{T}_\times]$

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{E}\mathbf{P}_l = 0$$



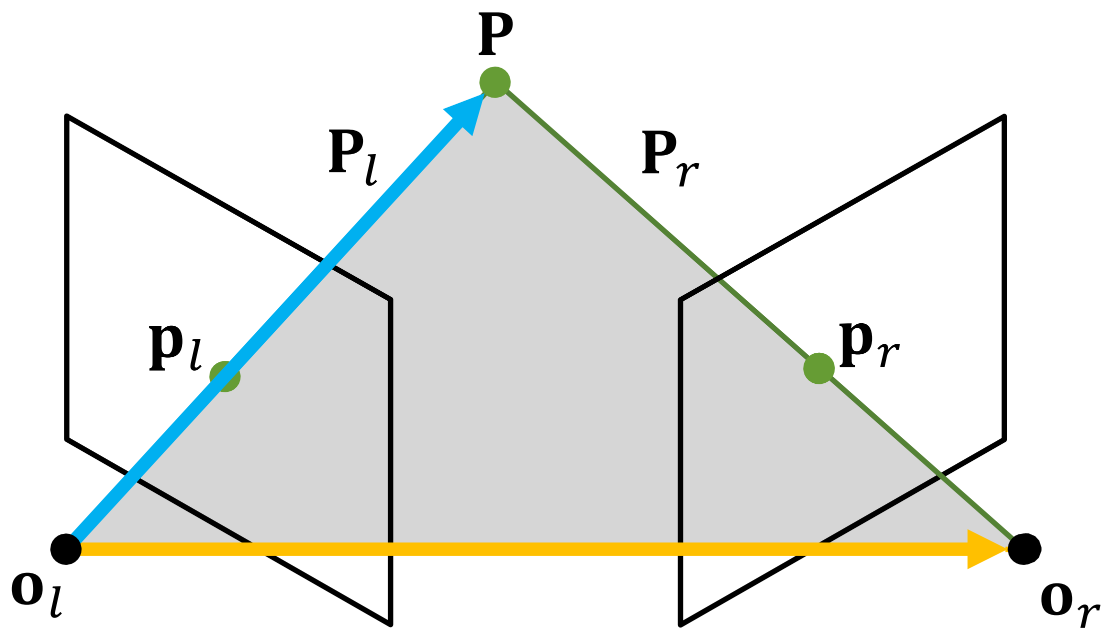
共面条件:

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{R}[\mathbf{T}_\times]\mathbf{P}_l = 0$$

令 $\mathbf{E} = \mathbf{R}[\mathbf{T}_\times]$

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{E}\mathbf{P}_l = 0$$

本质矩阵



共面条件:

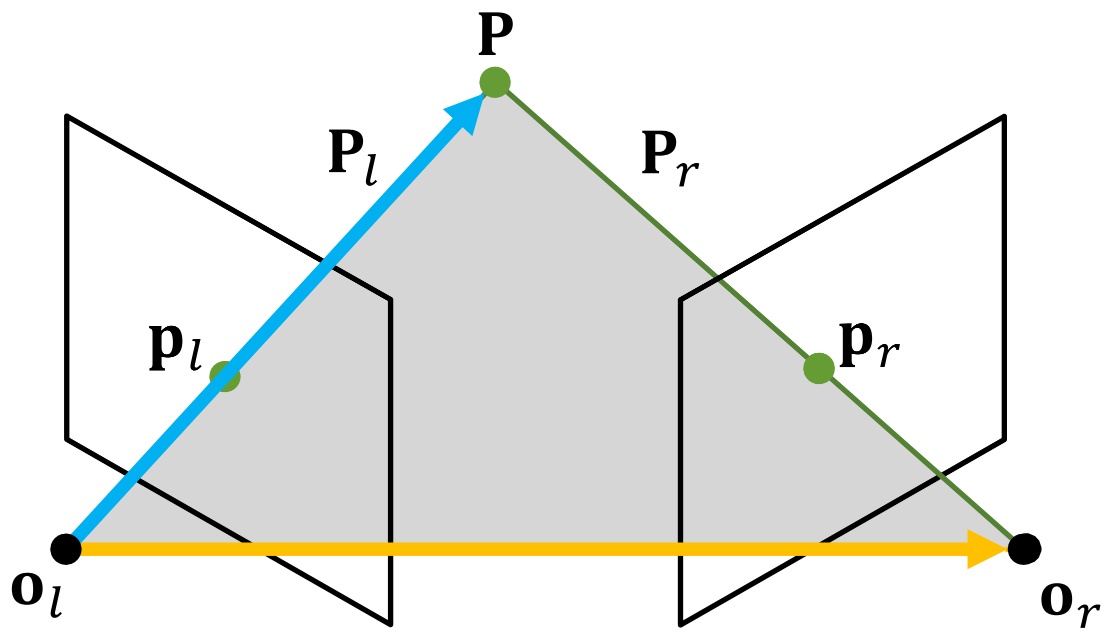
$$\mathbf{P}_r^{\mathrm{T}}\mathbf{R}[\mathbf{T}_\times]\mathbf{P}_l = 0$$

令 $\mathbf{E} = \mathbf{R}[\mathbf{T}_\times]$

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{E}\mathbf{P}_l = 0$$

本质矩阵

有多少个自由度?



共面条件:

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{R}[\mathbf{T}_\times]\mathbf{P}_l = 0$$

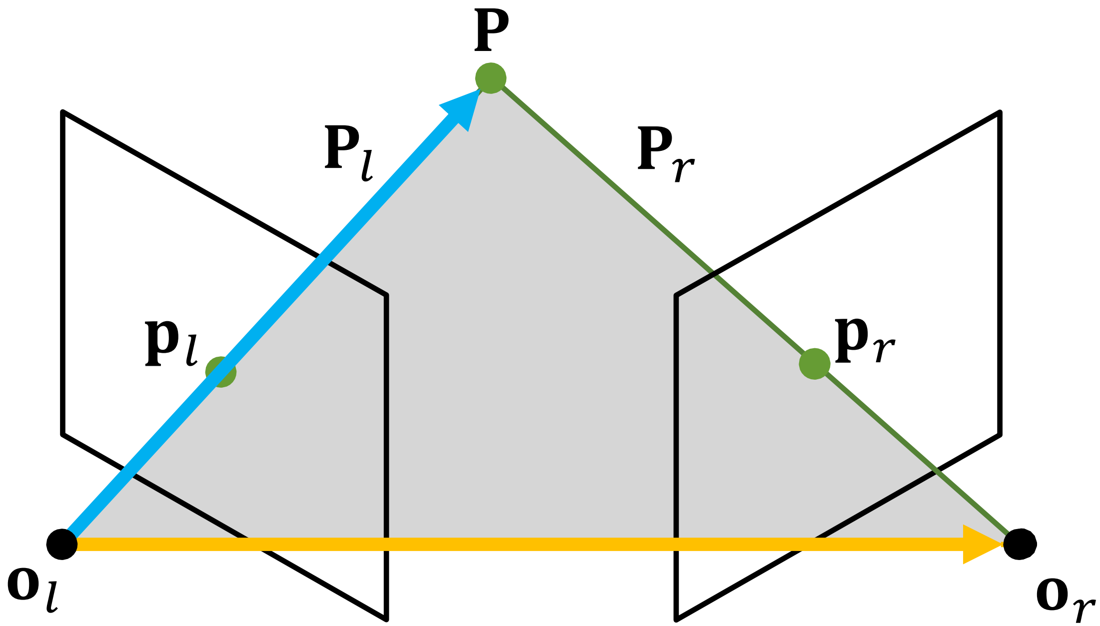
令 $\mathbf{E} = \mathbf{R}[\mathbf{T}_\times]$

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{E}\mathbf{P}_l = 0$$

本质矩阵

有多少个自由度?

5自由度



共面条件:

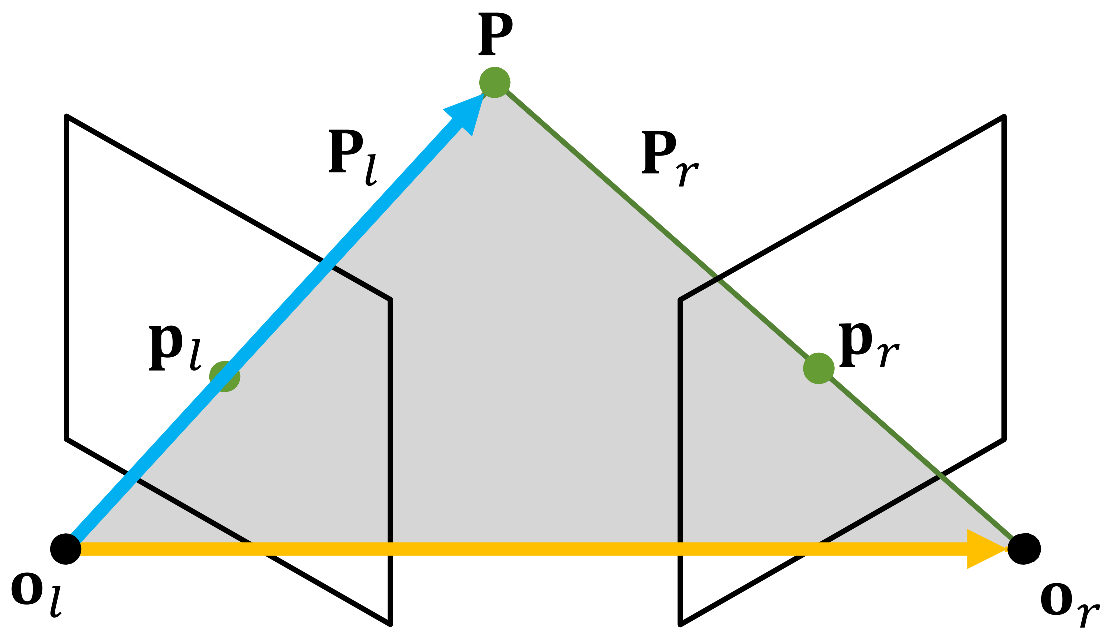
$$\mathbf{P}_r^{\mathrm{T}}\mathbf{R}[\mathbf{T}_\times]\mathbf{P}_l = 0$$

令 $\mathbf{E} = \mathbf{R}[\mathbf{T}_\times]$

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{E}\mathbf{P}_l = 0$$

本质矩阵

P在左、右两个视点中的坐标之间的约束关系



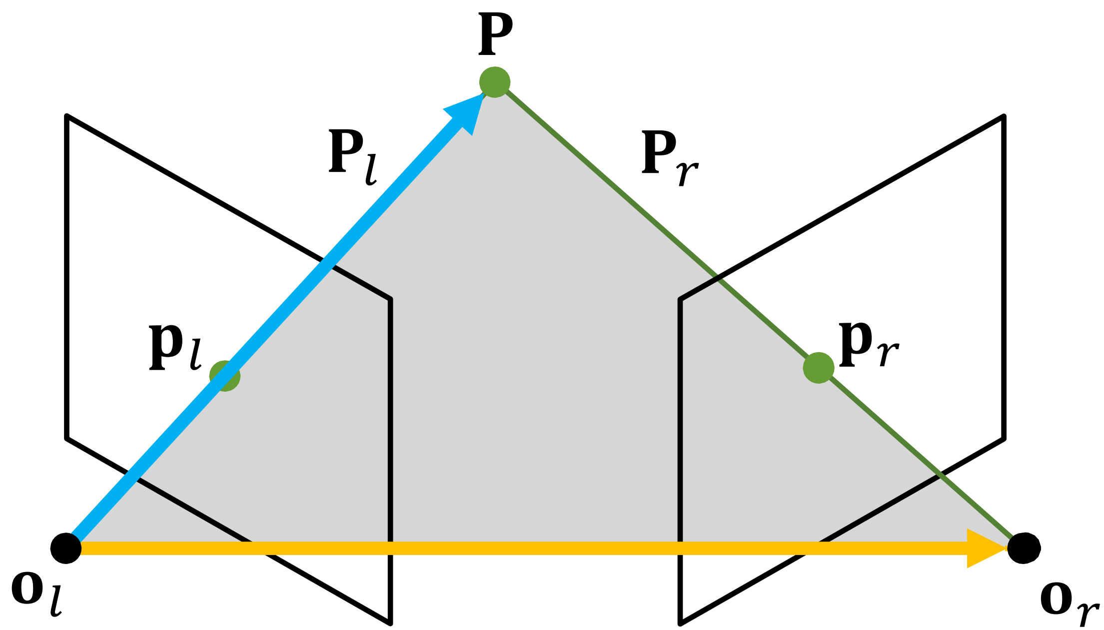
共面条件:

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{R}[\mathbf{T}_\times]\mathbf{P}_l = 0$$

令 $\mathbf{E} = \mathbf{R}[\mathbf{T}_\times]$

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{E}\mathbf{P}_l = 0$$

我们如何将这个约束关系修改成
图像点之间的关系？（在相机坐标系中）



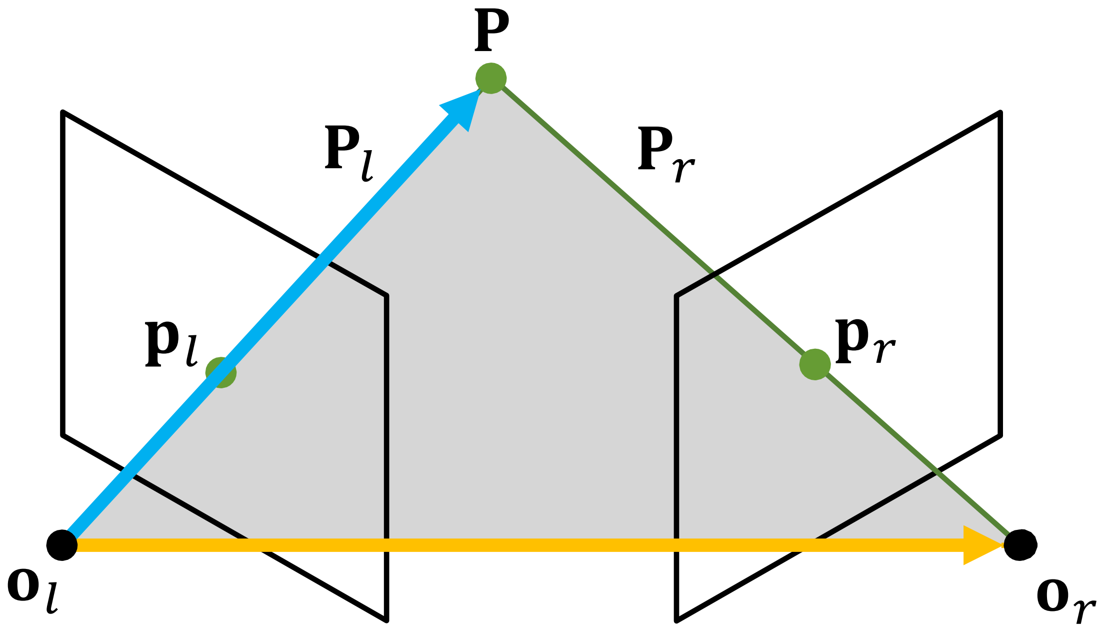
共面条件:

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{R}[\mathbf{T}_\times]\mathbf{P}_l = 0$$

令 $\mathbf{E} = \mathbf{R}[\mathbf{T}_\times]$

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{E}\mathbf{P}_l = 0$$

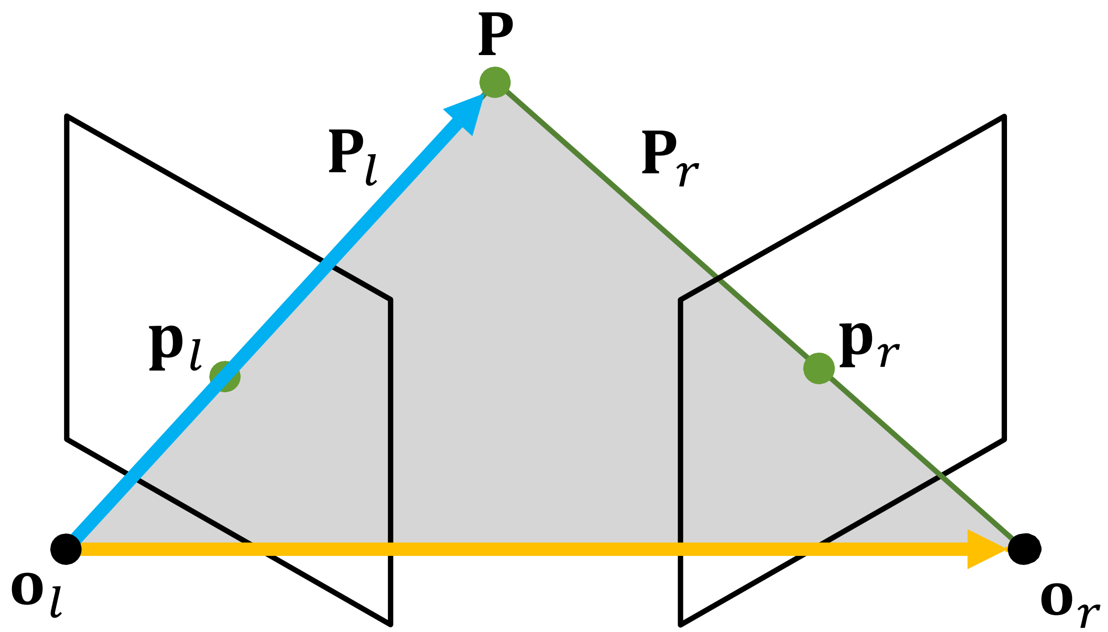
回顾: $\mathbf{p} = f\dfrac{\mathbf{P}}{Z}$



共面条件:

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{E}\mathbf{P}_l = 0$$

回顾：$\mathbf{p} = f\dfrac{\mathbf{P}}{Z}$

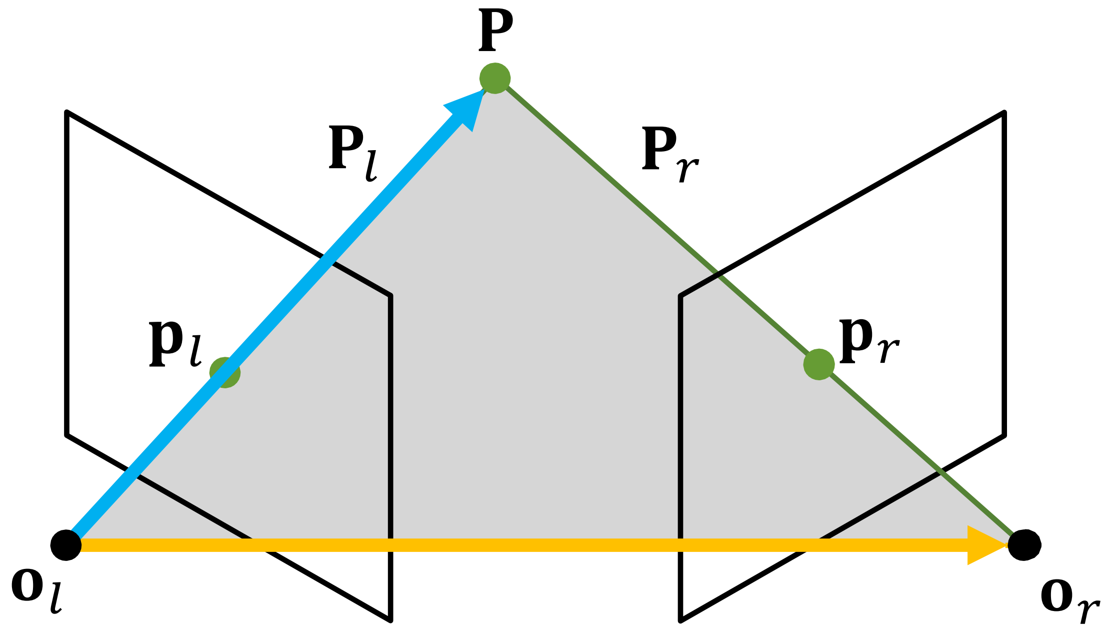


共面条件:

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{E}\mathbf{P}_l = 0$$

回顾： $\mathbf{p} = f\frac{\mathbf{P}}{Z}$

$$\left(\frac{\mathbf{p}_r Z_r}{f_r}\right)^{\mathrm{T}}\mathbf{E}\left(\frac{\mathbf{p}_l Z_l}{f_l}\right) = 0$$



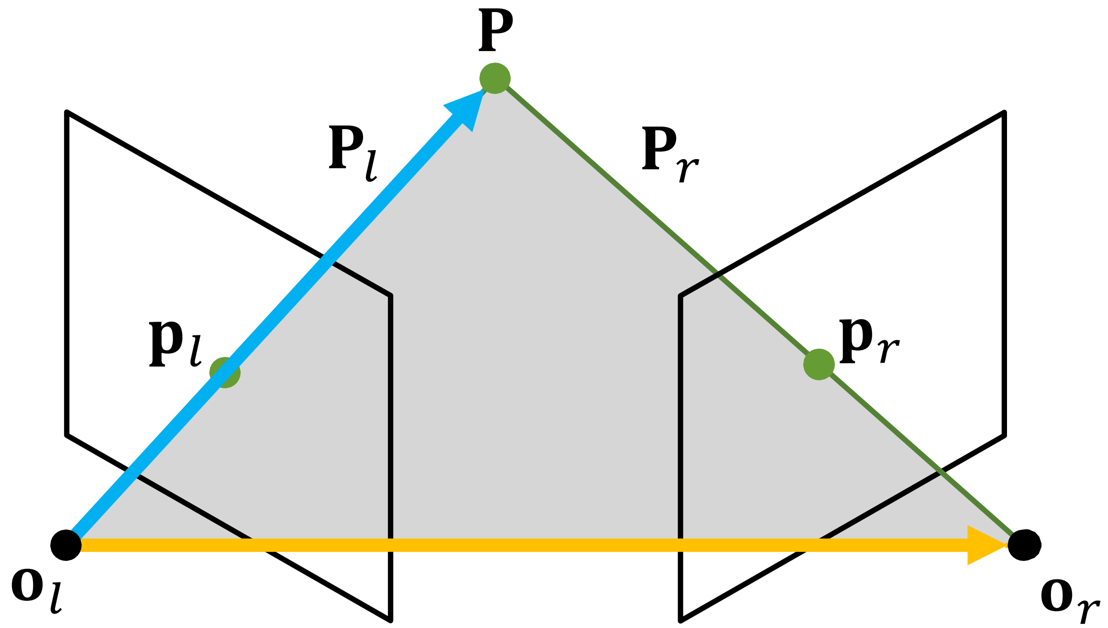
共面条件:

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{E}\mathbf{P}_l = 0$$

回顾: $\mathbf{p} = f\dfrac{\mathbf{P}}{Z}$

$$\left(\frac{\mathbf{p}_r Z_r}{f_r}\right)^{\mathrm{T}} \mathbf{E} \left(\frac{\mathbf{p}_l Z_l}{f_l}\right) = 0$$

化简



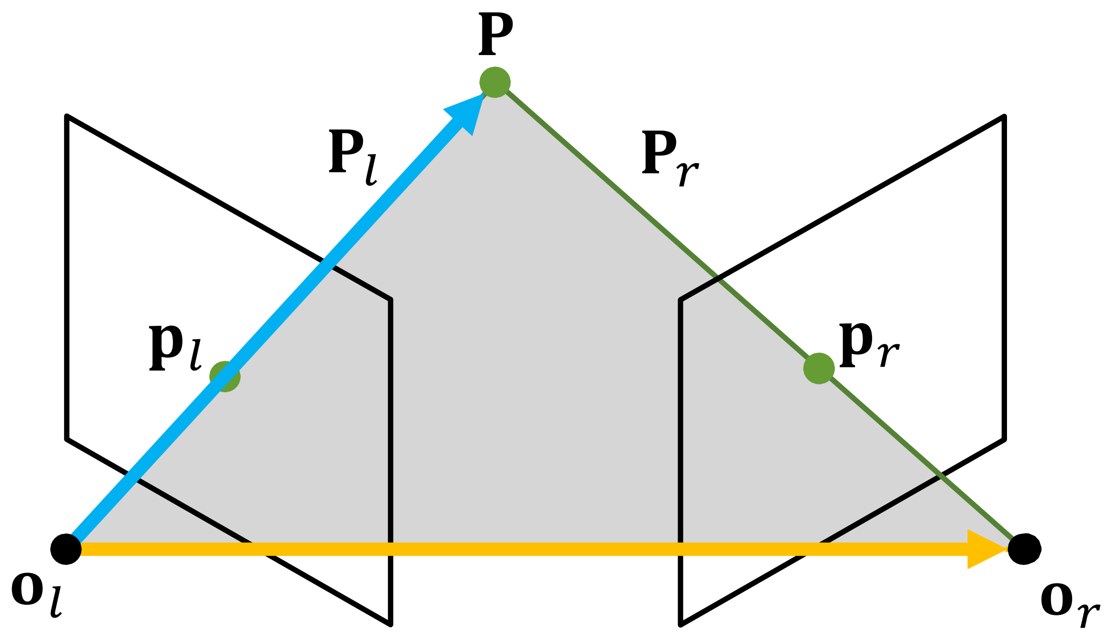
共面条件:

$$\mathbf{P}_r^{\mathrm{T}}\mathbf{E}\mathbf{P}_l = 0$$

回顾： $\mathbf{p} = f\dfrac{\mathbf{P}}{Z}$

$$\left(\frac{\mathbf{p}_r Z_r}{f_r}\right)^{\mathrm{T}} \mathbf{E} \left(\frac{\mathbf{p}_l Z_l}{f_l}\right) = 0$$

化简



共面条件:

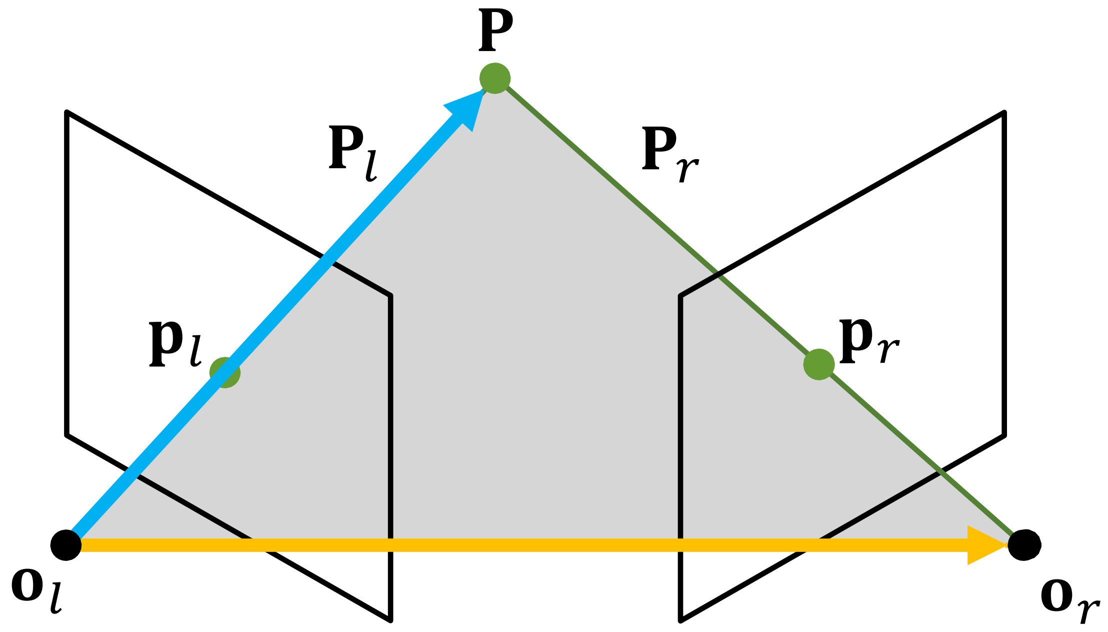
$$\mathbf{P}_r^{\mathrm{T}}\mathbf{E}\mathbf{P}_l = 0$$

回顾：$\mathbf{p} = f\dfrac{\mathbf{P}}{Z}$

$$\left(\frac{\mathbf{p}_r Z_r}{f_r}\right)^{\mathrm{T}}\mathbf{E}\left(\frac{\mathbf{p}_l Z_l}{f_l}\right) = 0$$

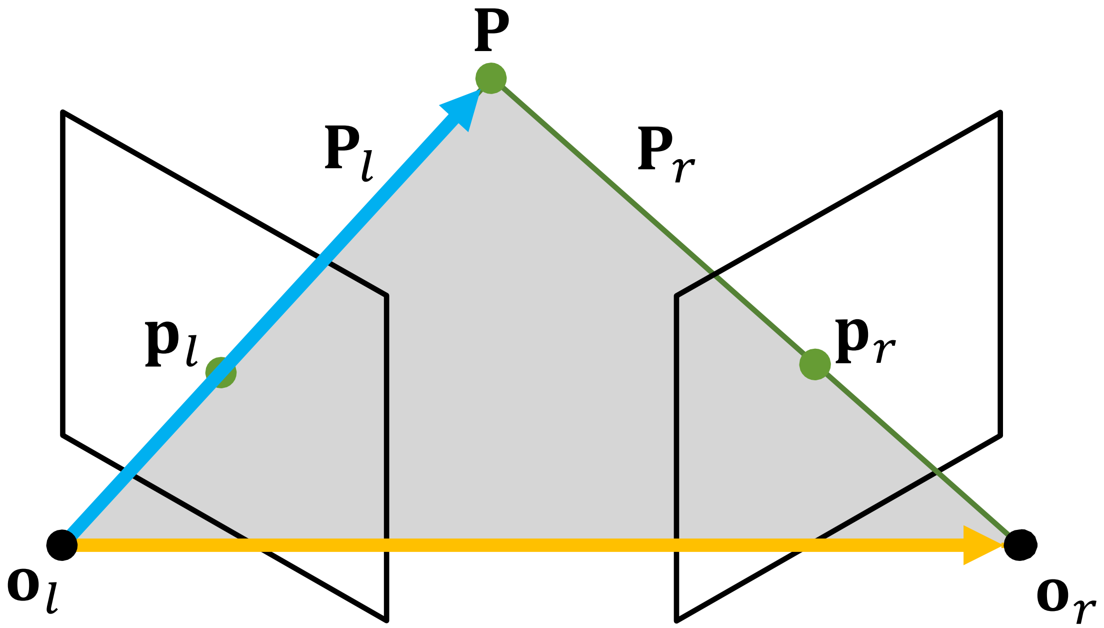
化简

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{E}\mathbf{p}_l = 0$$



共面条件:

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{E}\mathbf{p}_l = 0$$



共面条件:

$$\mathbf{p}_r^\top \mathbf{E} \mathbf{p}_l = 0$$

假设图像点是在相机坐标系中测量
而不是像素坐标系

回顾：　投影流程

$$\begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} & -\mathbf{R}_1^{\mathrm{T}}\mathbf{T} \\ \mathbf{R} & -\mathbf{R}_2^{\mathrm{T}}\mathbf{T} \\ & -\mathbf{R}_3^{\mathrm{T}}\mathbf{T} \end{pmatrix} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix}$$

外参矩阵

$\mathbf{M}_{\mathrm{ext}}$

$$
\begin{pmatrix} -f/s_x & 0 & o_x \\ 0 & -f/s_y & o_y \\ 0 & 0 & 1 \end{pmatrix}
\begin{pmatrix} & -\mathbf{R}_1^{\mathrm{T}}\mathbf{T} \\ \mathbf{R} & -\mathbf{R}_2^{\mathrm{T}}\mathbf{T} \\ & -\mathbf{R}_3^{\mathrm{T}}\mathbf{T} \end{pmatrix}
\begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix}
$$

内参矩阵 $\mathbf{M}_{\text{int}}$

外参矩阵 $\mathbf{M}_{\text{ext}}$

$$\begin{pmatrix} -f/s_x & 0 & o_x \\ 0 & -f/s_y & o_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} & -\mathbf{R}_1^{\mathrm{T}}\mathbf{T} \\ \mathbf{R} & -\mathbf{R}_2^{\mathrm{T}}\mathbf{T} \\ & -\mathbf{R}_3^{\mathrm{T}}\mathbf{T} \end{pmatrix} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix}$$

内参矩阵 $\mathbf{M}_{\mathrm{int}}$

外参矩阵 $\mathbf{M}_{\mathrm{ext}}$

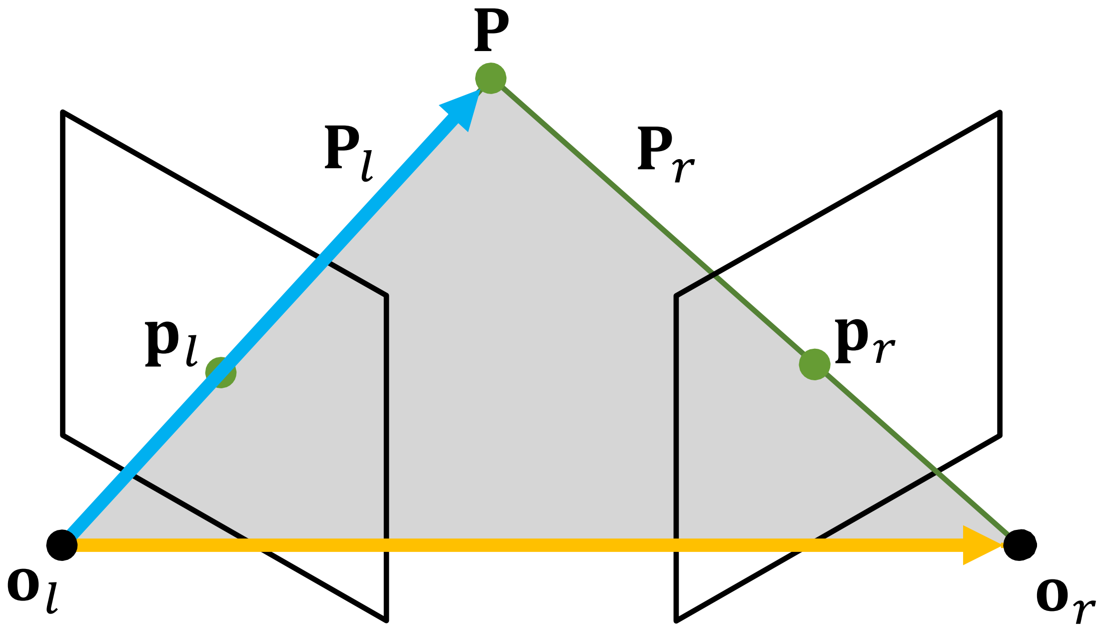
内参矩阵是可逆的

$$\begin{pmatrix} \alpha x_{im} \\ \alpha y_{im} \\ \alpha \end{pmatrix} = \underbrace{\begin{pmatrix} -f/s_x & 0 & o_x \\ 0 & -f/s_y & o_y \\ 0 & 0 & 1 \end{pmatrix}}_{\substack{\text{内参矩阵} \\ \mathbf{M}_{\text{int}}}} \underbrace{\begin{pmatrix} \mathbf{R} & \begin{matrix} -\mathbf{R}_1^{\mathrm{T}}\mathbf{T} \\ -\mathbf{R}_2^{\mathrm{T}}\mathbf{T} \\ -\mathbf{R}_3^{\mathrm{T}}\mathbf{T} \end{matrix} \end{pmatrix}}_{\substack{\text{外参矩阵} \\ \mathbf{M}_{\text{ext}}}} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix}$$

$$
\begin{pmatrix} \alpha x_{im} \\ \alpha y_{im} \\ \alpha \end{pmatrix} = \underbrace{\begin{pmatrix} -f/s_x & 0 & o_x \\ 0 & -f/s_y & o_y \\ 0 & 0 & 1 \end{pmatrix}}_{\substack{\text{内参矩阵} \\ \mathbf{M}_{\text{int}}}} \underbrace{\begin{pmatrix} & -\mathbf{R}_1^{\mathrm{T}}\mathbf{T} \\ \mathbf{R} & -\mathbf{R}_2^{\mathrm{T}}\mathbf{T} \\ & -\mathbf{R}_3^{\mathrm{T}}\mathbf{T} \end{pmatrix}}_{\substack{\text{外参矩阵} \\ \mathbf{M}_{\text{ext}}}} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix}
$$

除以α

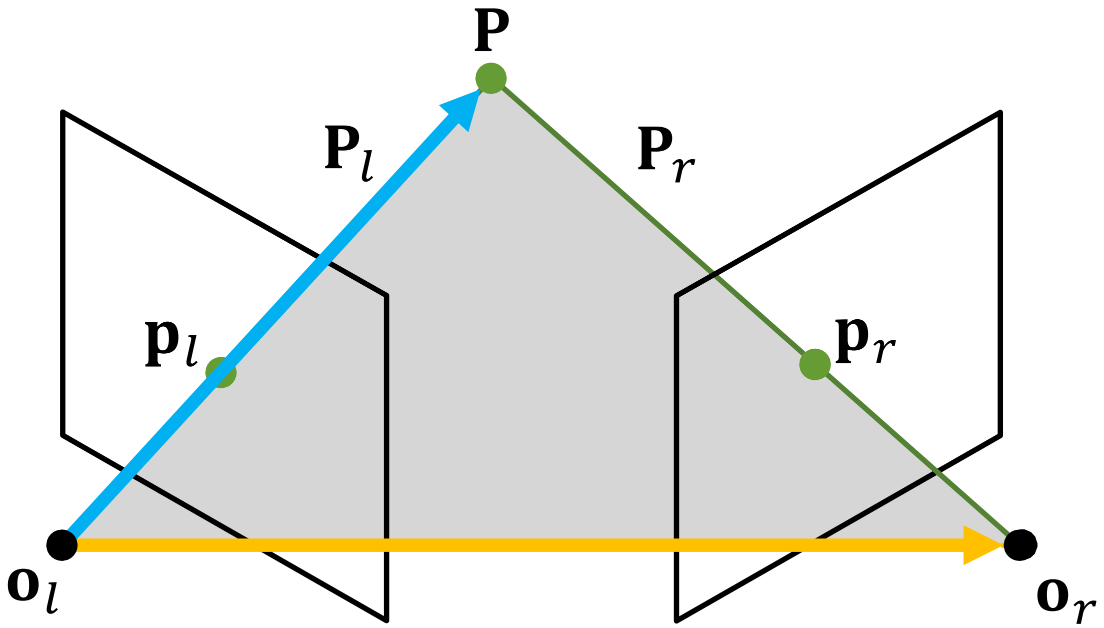
回顾：投影流程



共面条件:

$$\mathbf{p}_r^{\mathsf{T}}\mathbf{E}\mathbf{p}_l = 0$$

假设图像点是在相机坐标系中测量
而不是图像坐标系

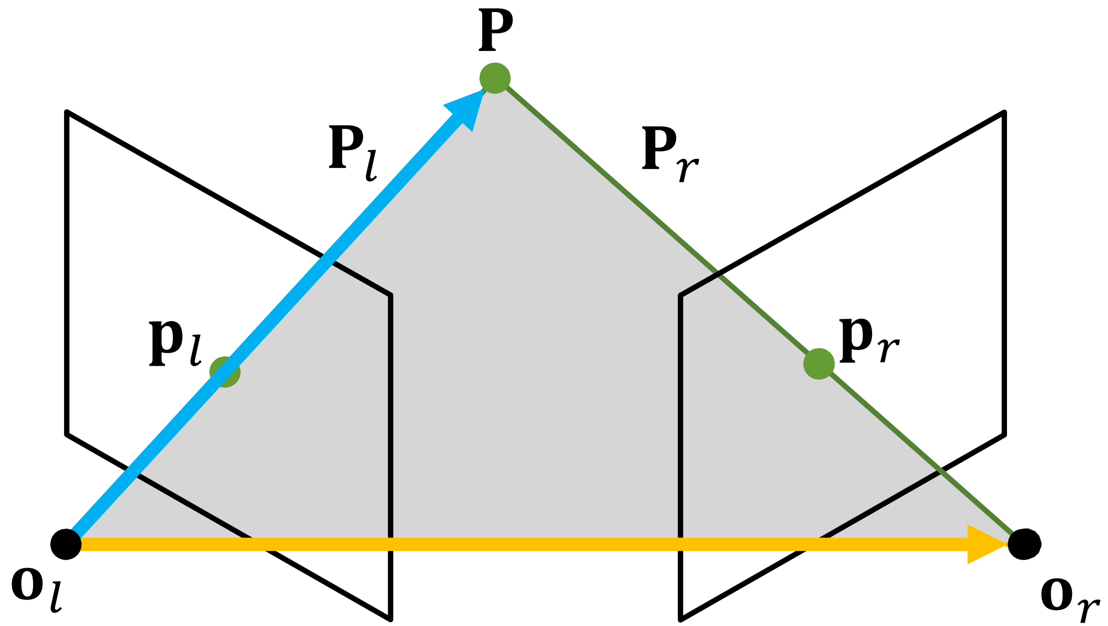


共面条件:

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{E}\mathbf{p}_l = 0$$

我们有:

$$\mathbf{p}_l = \mathbf{M}_{\mathrm{int},l}^{-1}\widetilde{\mathbf{p}}_l, \quad \mathbf{p}_r = \mathbf{M}_{\mathrm{int},r}^{-1}\widetilde{\mathbf{p}}_r$$



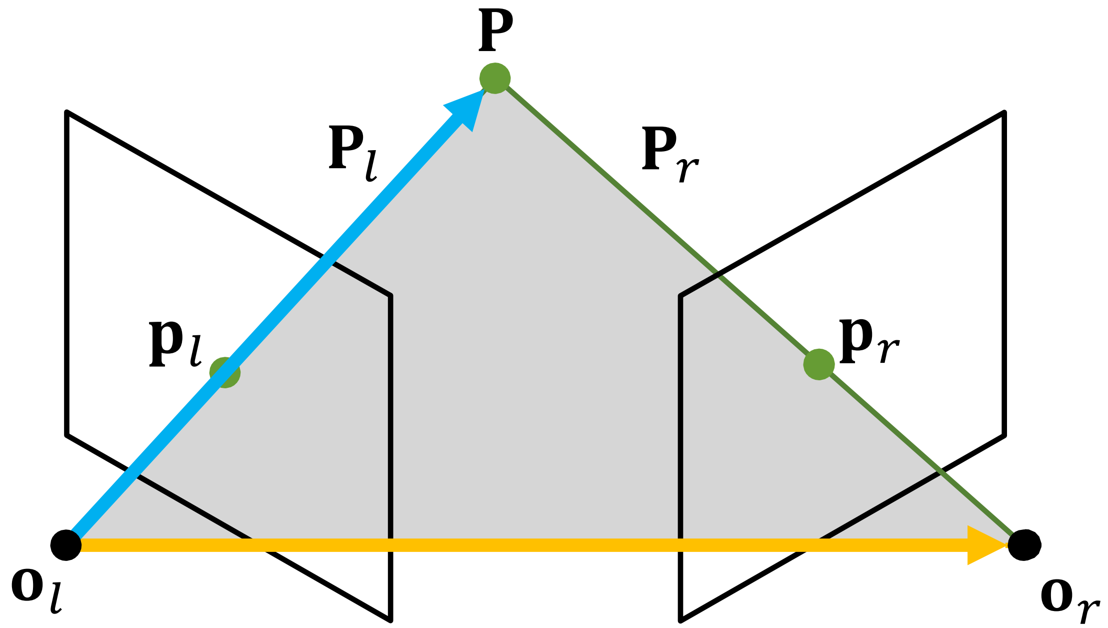
共面条件:

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{E}\mathbf{p}_l = 0$$

我们有:

$$\mathbf{p}_l = \mathbf{M}_{\mathrm{int},l}^{-1}\tilde{\mathbf{p}}_l, \quad \mathbf{p}_r = \mathbf{M}_{\mathrm{int},r}^{-1}\tilde{\mathbf{p}}_r$$

图像坐标



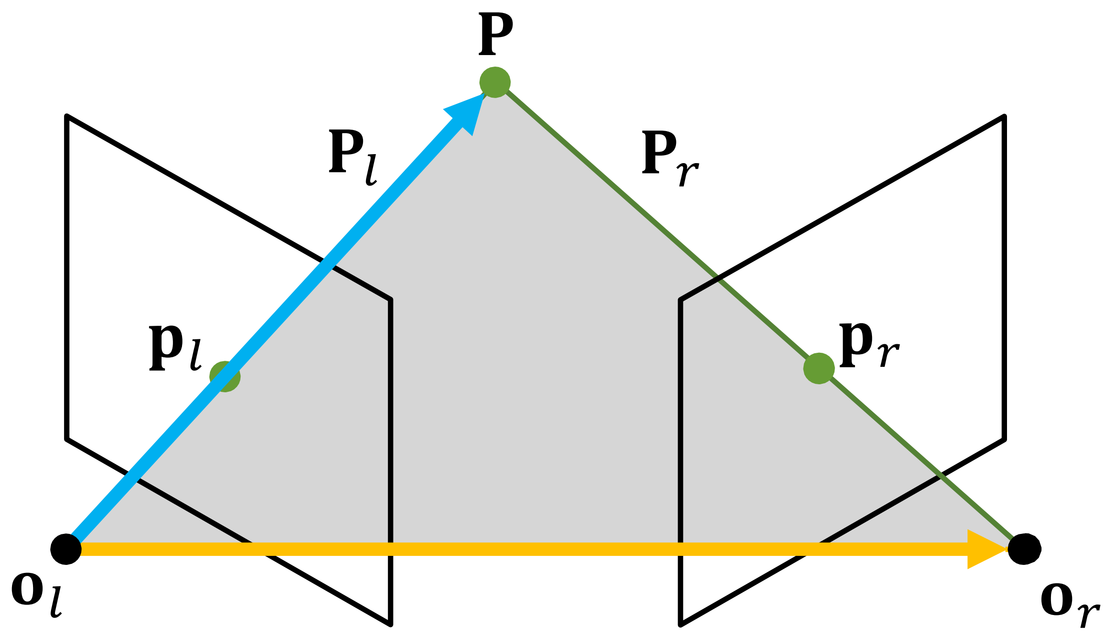
共面条件:

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{E}\mathbf{p}_l = 0$$

我们有:

$$\mathbf{p}_l = \mathbf{M}_{\mathrm{int},l}^{-1}\widetilde{\mathbf{p}}_l, \quad \mathbf{p}_r = \mathbf{M}_{\mathrm{int},r}^{-1}\widetilde{\mathbf{p}}_r$$

代入



共面条件:

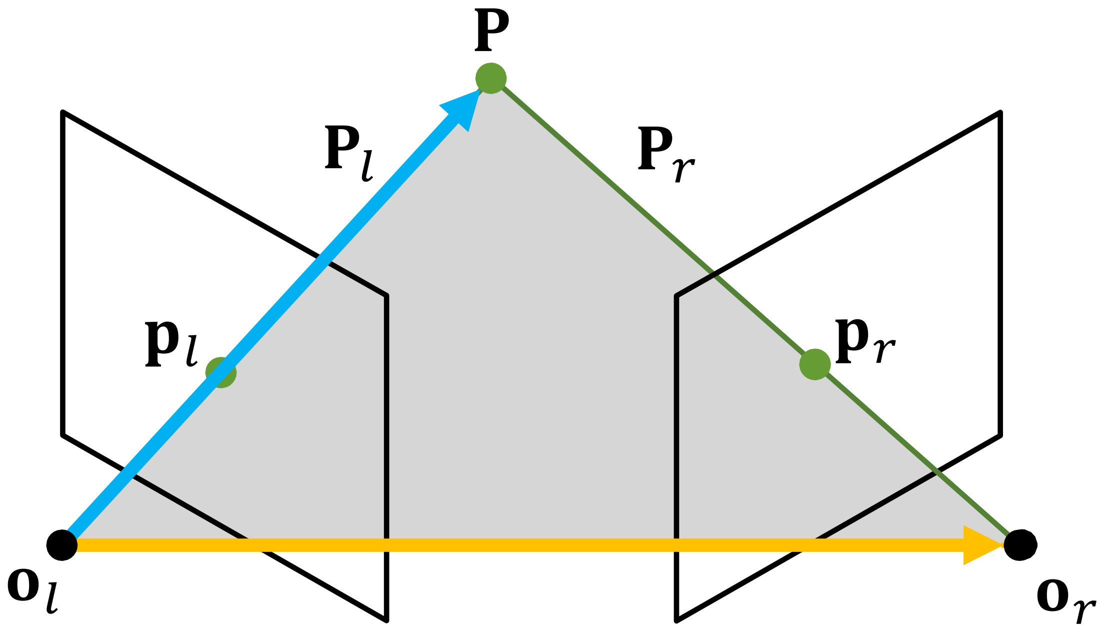
$$\mathbf{p}_r^{\mathrm{T}}\mathbf{E}\mathbf{p}_l = 0$$

我们有:

$$\mathbf{p}_l = \mathbf{M}_{\mathrm{int},l}^{-1}\widetilde{\mathbf{p}}_l, \quad \mathbf{p}_r = \mathbf{M}_{\mathrm{int},r}^{-1}\widetilde{\mathbf{p}}_r$$

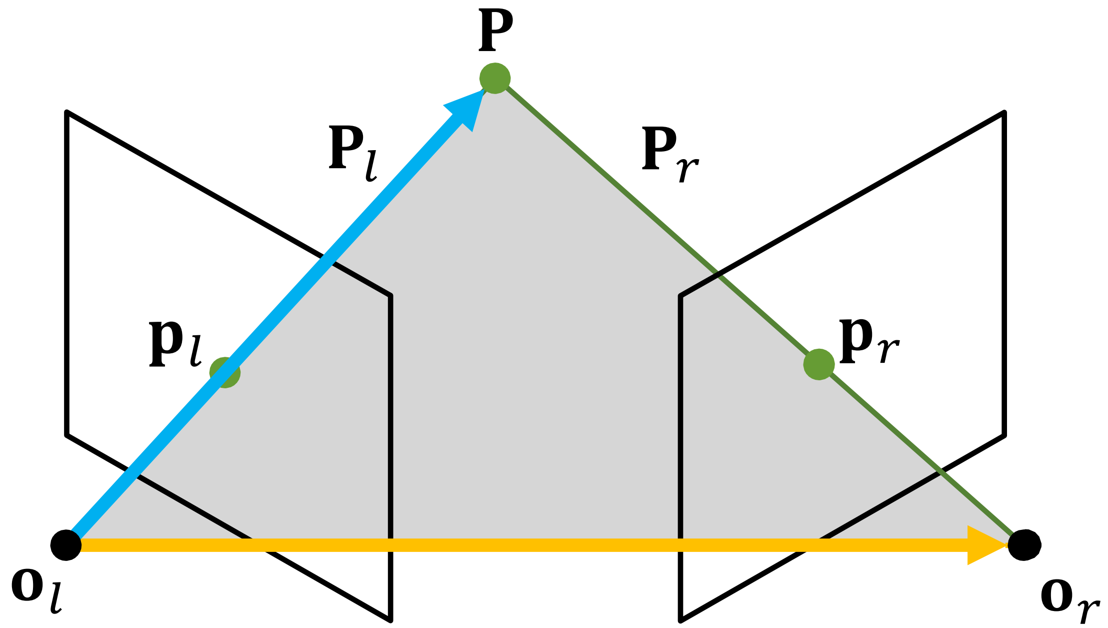
代入

$$\widetilde{\mathbf{p}}_r^{\mathrm{T}}\mathbf{M}_{\mathrm{int},r}^{-\mathrm{T}}\mathbf{E}\mathbf{M}_{\mathrm{int},l}^{-1}\widetilde{\mathbf{p}}_l = 0$$



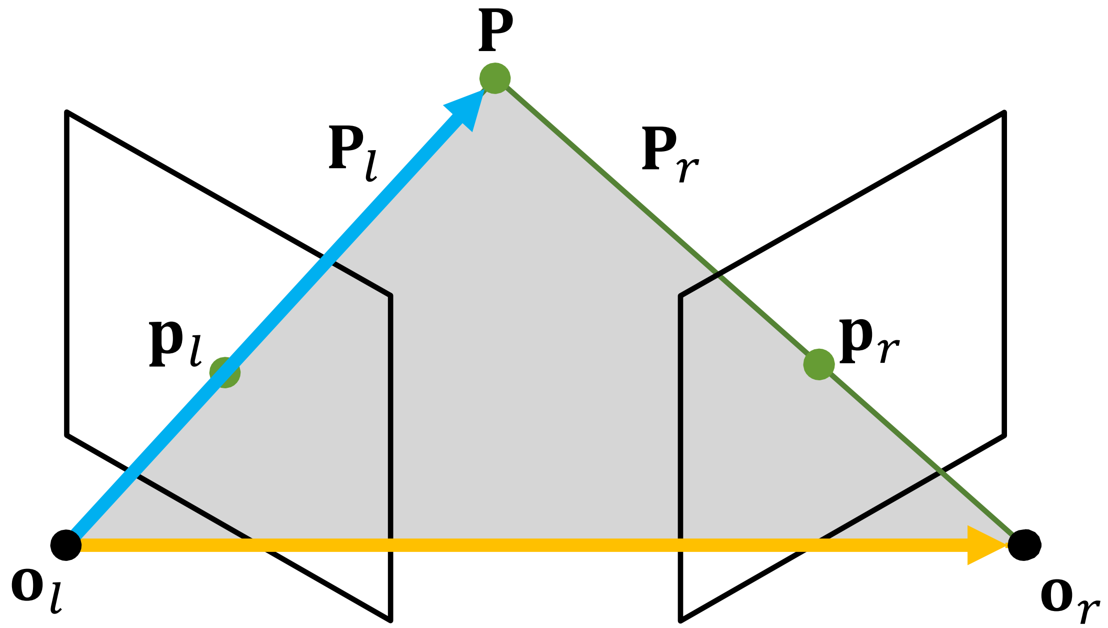
共面条件:

$$\tilde{\mathbf{p}}_r^{\mathrm{T}}\mathbf{M}_{\mathrm{int},r}^{-\mathrm{T}}\mathbf{E}\mathbf{M}_{\mathrm{int},l}^{-1}\tilde{\mathbf{p}}_l = 0$$



共面条件:

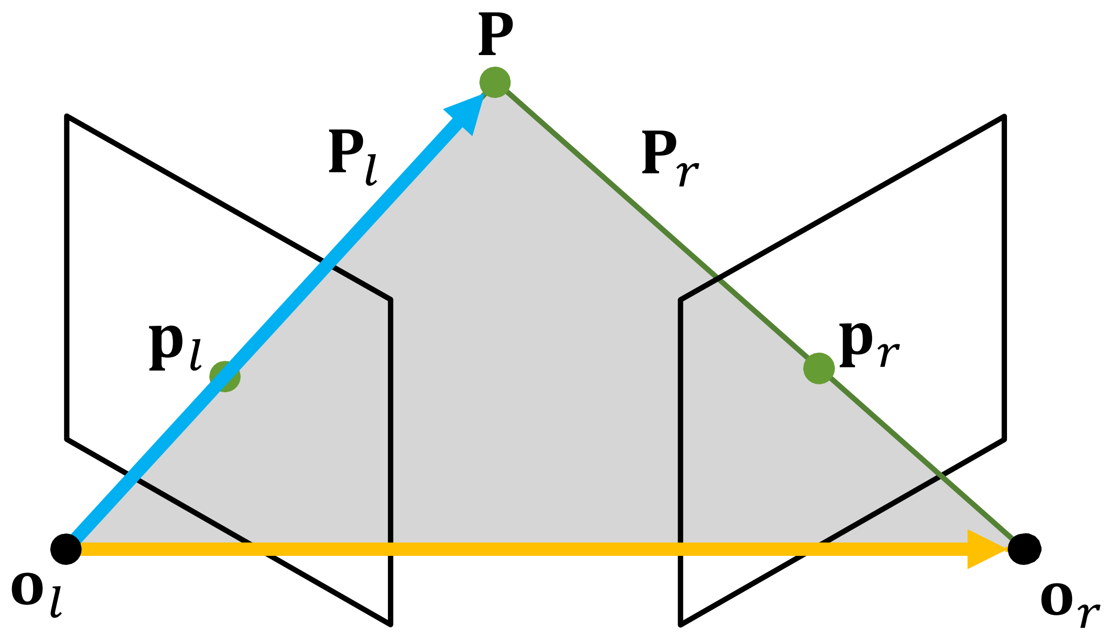
$$\tilde{\mathbf{p}}_r^{\mathrm{T}} \mathbf{M}_{\mathrm{int},r}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\mathrm{int},l}^{-1} \tilde{\mathbf{p}}_l = 0$$



共面条件:

$$\tilde{\mathbf{p}}_r^{\mathrm{T}} \mathbf{M}_{\mathrm{int},r}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\mathrm{int},l}^{-1} \tilde{\mathbf{p}}_l = 0$$

令 $\mathbf{F} = \mathbf{M}_{\mathrm{int},\boldsymbol{r}}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\mathrm{int},\boldsymbol{l}}^{-\mathbf{1}}$

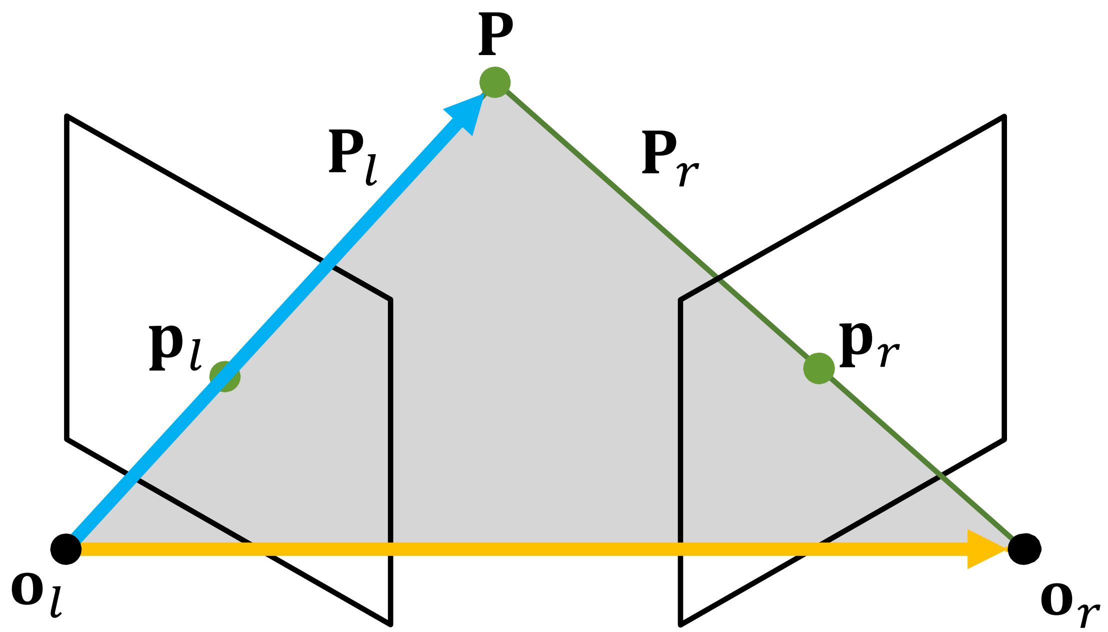


共面条件:

$$\tilde{\mathbf{p}}_r^{\mathrm{T}}\mathbf{M}_{\mathrm{int},r}^{-\mathrm{T}}\mathbf{E}\mathbf{M}_{\mathrm{int},l}^{-1}\tilde{\mathbf{p}}_l = 0$$

令 $\mathbf{F} = \mathbf{M}_{\mathrm{int},\boldsymbol{r}}^{-\mathrm{T}}\mathbf{E}\mathbf{M}_{\mathrm{int},\boldsymbol{l}}^{-\mathbf{1}}$

基础矩阵



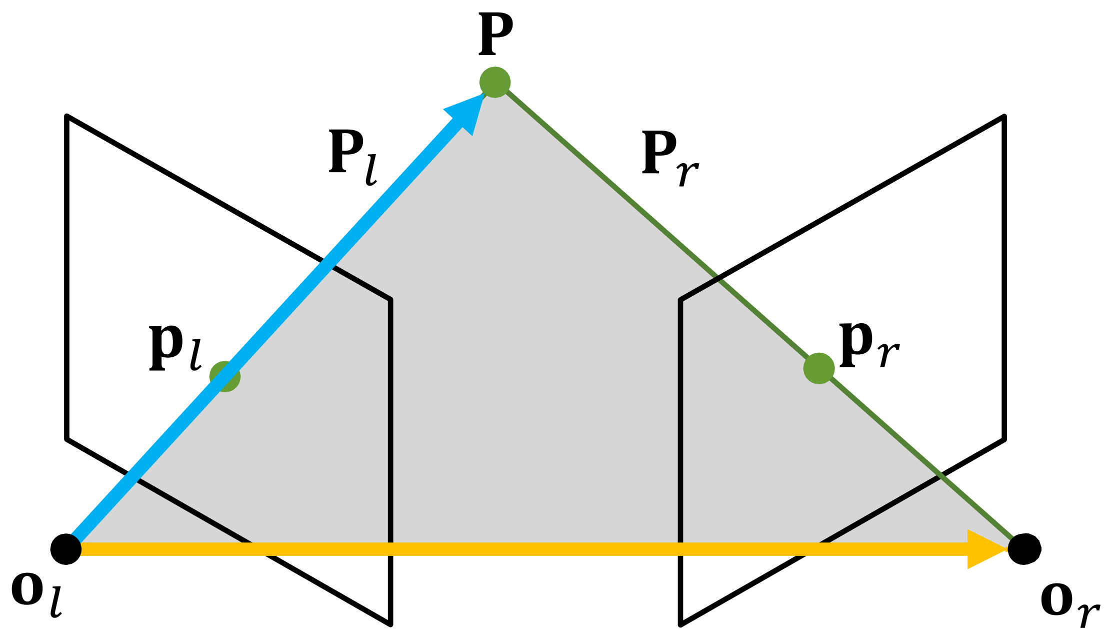
共面条件:

$$\tilde{\mathbf{p}}_r^{\mathrm{T}} \mathbf{M}_{\mathrm{int},r}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\mathrm{int},l}^{-1} \tilde{\mathbf{p}}_l = 0$$

令 $\mathbf{F} = \mathbf{M}_{\mathrm{int},\boldsymbol{r}}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\mathrm{int},\boldsymbol{l}}^{-\mathbf{1}}$

基础矩阵

有多少个自由度?



共面条件:

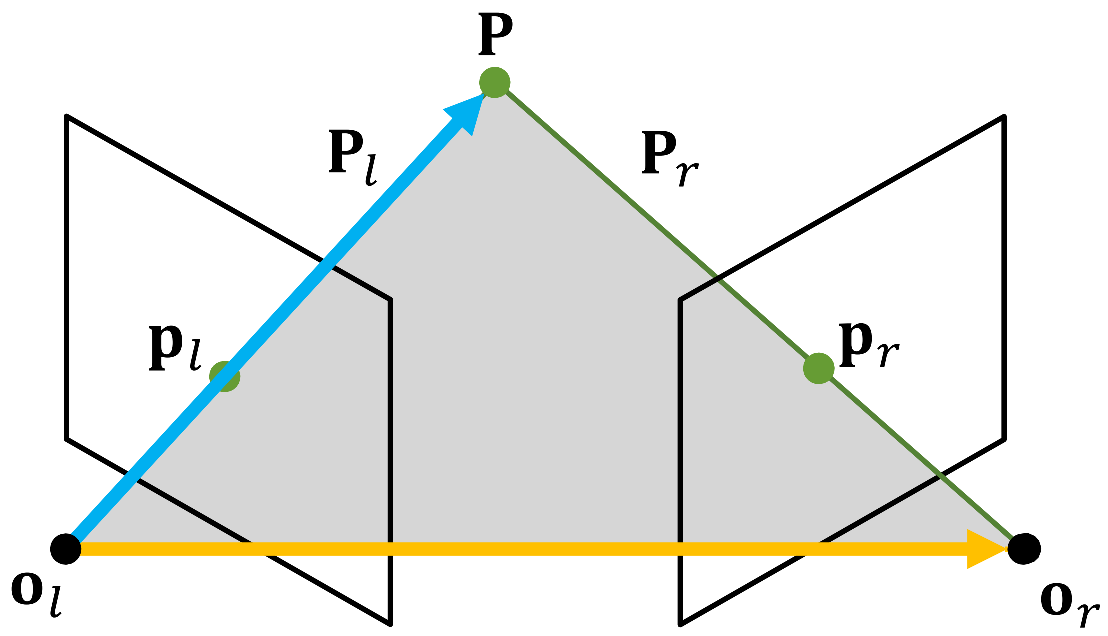
$$\tilde{\mathbf{p}}_r^{\mathrm{T}}\mathbf{M}_{\mathrm{int},r}^{-\mathrm{T}}\mathbf{E}\mathbf{M}_{\mathrm{int},l}^{-1}\tilde{\mathbf{p}}_l = 0$$

令 $\mathbf{F} = \mathbf{M}_{\mathrm{int},\boldsymbol{r}}^{-\mathrm{T}}\mathbf{E}\mathbf{M}_{\mathrm{int},\boldsymbol{l}}^{-1}$

基础矩阵

有多少个自由度?

7个自由度



共面条件:

$$\tilde{\mathbf{p}}_r^{\mathrm{T}}\mathbf{M}_{\mathrm{int},r}^{-\mathrm{T}}\mathbf{E}\mathbf{M}_{\mathrm{int},l}^{-1}\tilde{\mathbf{p}}_l = 0$$

令 $\mathbf{F} = \mathbf{M}_{\mathrm{int},\boldsymbol{r}}^{-\mathrm{T}}\mathbf{E}\mathbf{M}_{\mathrm{int},\boldsymbol{l}}^{-1}$

基础矩阵约束： $\tilde{\mathbf{p}}_r^{\mathrm{T}}\mathbf{F}\tilde{\mathbf{p}}_l = \mathbf{0}$

基础矩阵约束：$\tilde{\mathbf{p}}_r^{\mathrm{T}}\mathbf{F}\tilde{\mathbf{p}}_l = \mathbf{0}$

备注：

基础矩阵约束：$\tilde{\mathbf{p}}_r^{\mathrm{T}}\mathbf{F}\tilde{\mathbf{p}}_l = \mathbf{0}$

备注：

基础矩阵捕获两个视点间的投影几何

基础矩阵约束：$\tilde{\mathbf{p}}_r^{\mathrm{T}}\mathbf{F}\tilde{\mathbf{p}}_l = \mathbf{0}$

备注：

基础矩阵捕获两个视点间的投影几何

与场景结构无关

基础矩阵约束：$\tilde{\mathbf{p}}_r^{\mathrm{T}}\mathbf{F}\tilde{\mathbf{p}}_l = \mathbf{0}$

备注：

基础矩阵捕获两个视点间的投影几何

与场景结构无关

可以通过匹配对应点计算，而不需要事先知道相机的内参和外参

基础矩阵约束：$\tilde{\mathbf{p}}_r^{\mathrm{T}} \mathbf{F} \tilde{\mathbf{p}}_l = \mathbf{0}$

备注：

基础矩阵捕获两个视点间的投影几何

与场景结构无关

可以通过匹配对应点计算，而不需要事先知道相机的内参和外参

在齐次坐标系中是确定的，但不约束尺度因子

基础矩阵约束：$\tilde{\mathbf{p}}_r^{\mathrm{T}}\mathbf{F}\tilde{\mathbf{p}}_l = \mathbf{0}$

备注：

基础矩阵捕获两个视点间的投影几何

与场景结构无关

可以通过匹配对应点计算，而不需要事先知道相机的内参和外参

在齐次坐标系中是确定的，但不约束尺度因子

秩为2

基础矩阵约束：$\tilde{\mathbf{p}}_r^{\mathrm{T}}\mathbf{F}\tilde{\mathbf{p}}_l = \mathbf{0}$

备注：

基础矩阵捕获两个视点间的投影几何

与场景结构无关

可以通过匹配对应点计算，而不需要事先知道相机的内参和外参

在齐次坐标系中是确定的，但不约束尺度因子

秩为2

极点是左、右零空间：$\mathbf{F}\mathbf{e}_l = \mathbf{0}$和$\mathbf{e}_r^{\mathrm{T}}\mathbf{F} = \mathbf{0}$

基础矩阵约束：$\tilde{\mathbf{p}}_r^{\mathrm{T}}\mathbf{F}\tilde{\mathbf{p}}_l = \mathbf{0}$

备注：

基础矩阵捕获两个视点间的投影几何

与场景结构无关

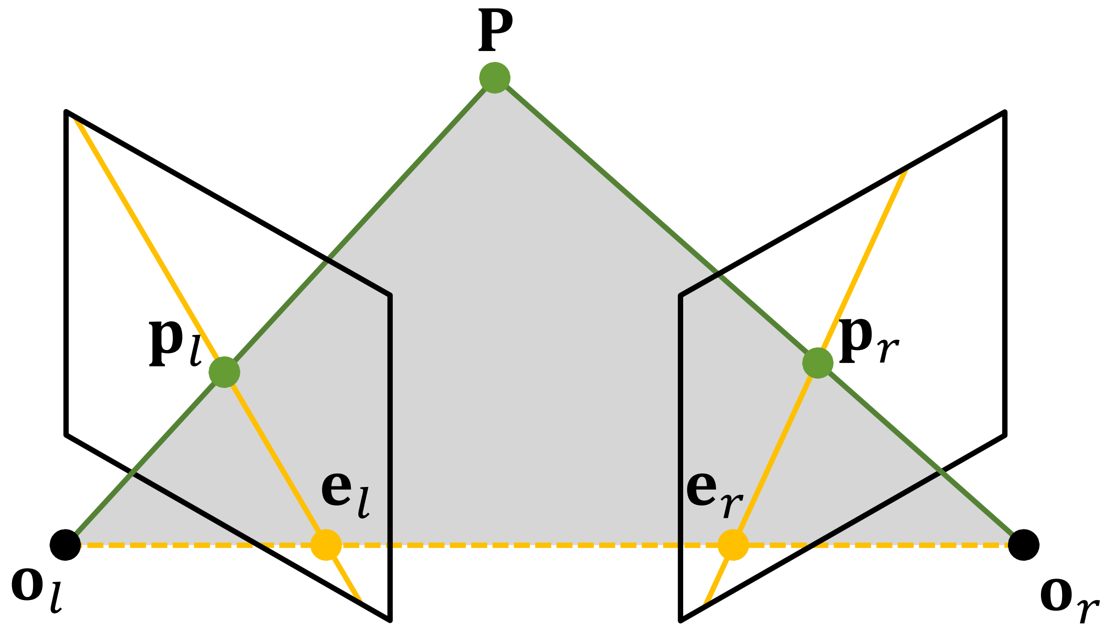
可以通过匹配对应点计算，而不需要事先知道相机的内参和外参

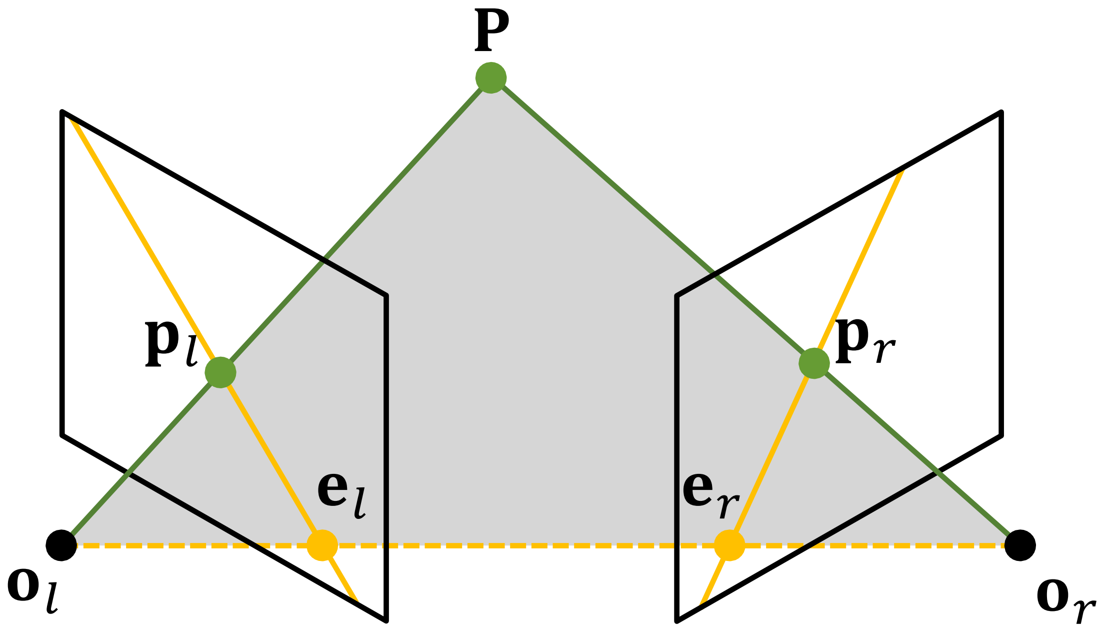
在齐次坐标系中是确定的，但不约束尺度因子

秩为2

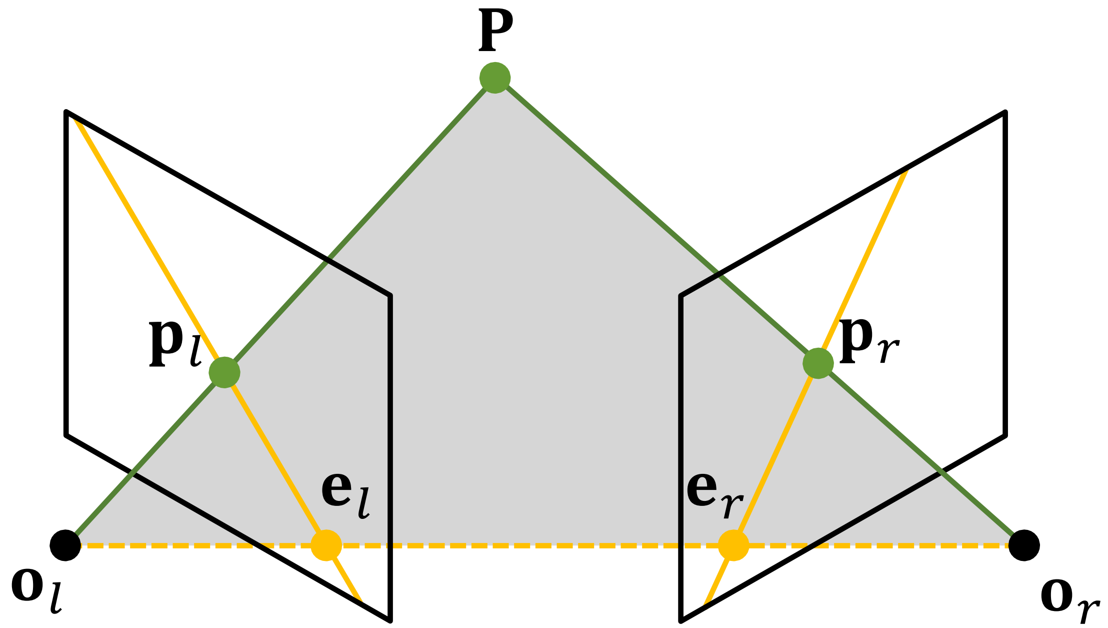
怎么得到的？

极点是左、右零空间：$\mathbf{F}\mathbf{e}_l = \mathbf{0}$和$\mathbf{e}_r^{\mathrm{T}}\mathbf{F} = \mathbf{0}$


$\mathbf{P}$
$\mathbf{p}_l$
$\mathbf{p}_r$
$\mathbf{e}_l$
$\mathbf{e}_r$
$\mathbf{o}_l$
$\mathbf{o}_r$

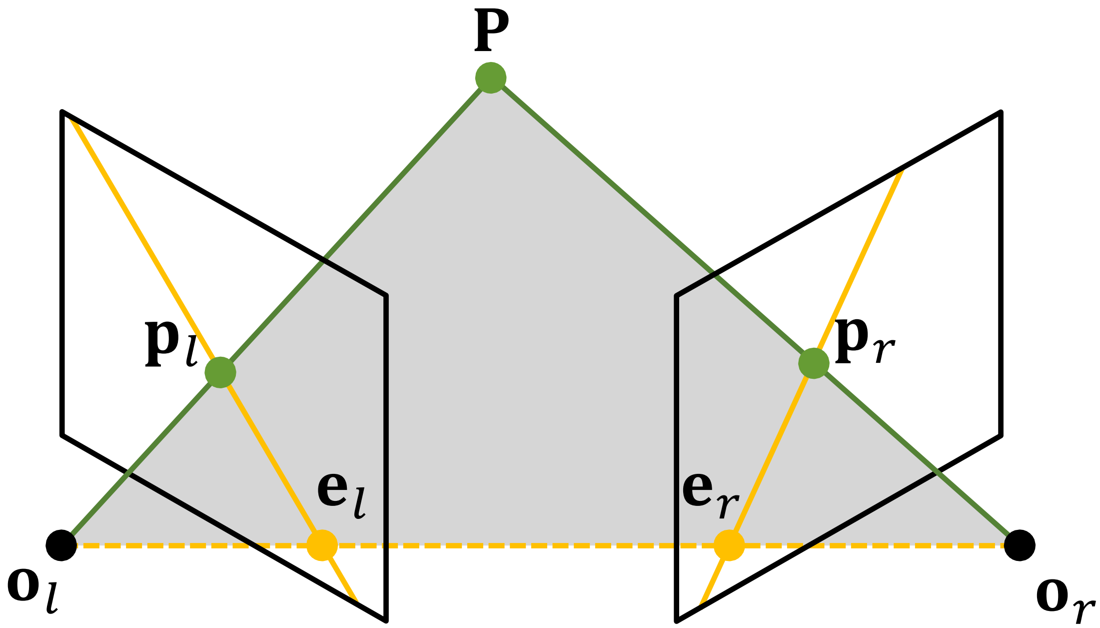


$$\mathbf{F} = \mathbf{M}_{\text{int},r}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\text{int},l}^{-\mathbf{1}}$$



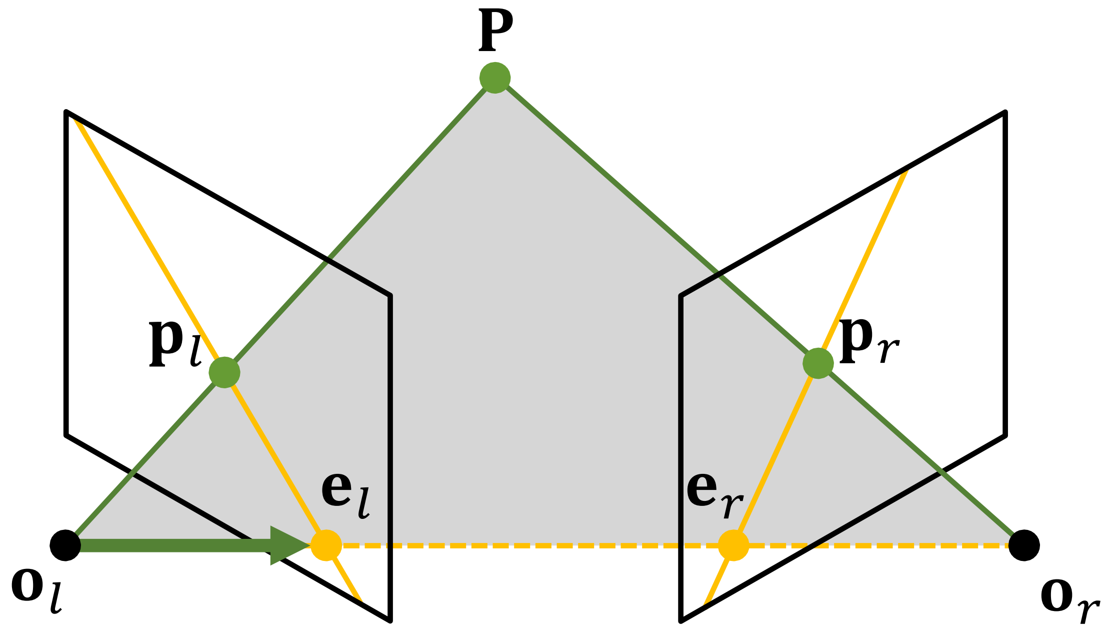
$$\mathbf{F} = \mathbf{M}_{\text{int},r}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\text{int},l}^{-\mathbf{1}}$$

$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\text{int},r}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\text{int},l}^{-\mathbf{1}} \mathbf{e}_l$$



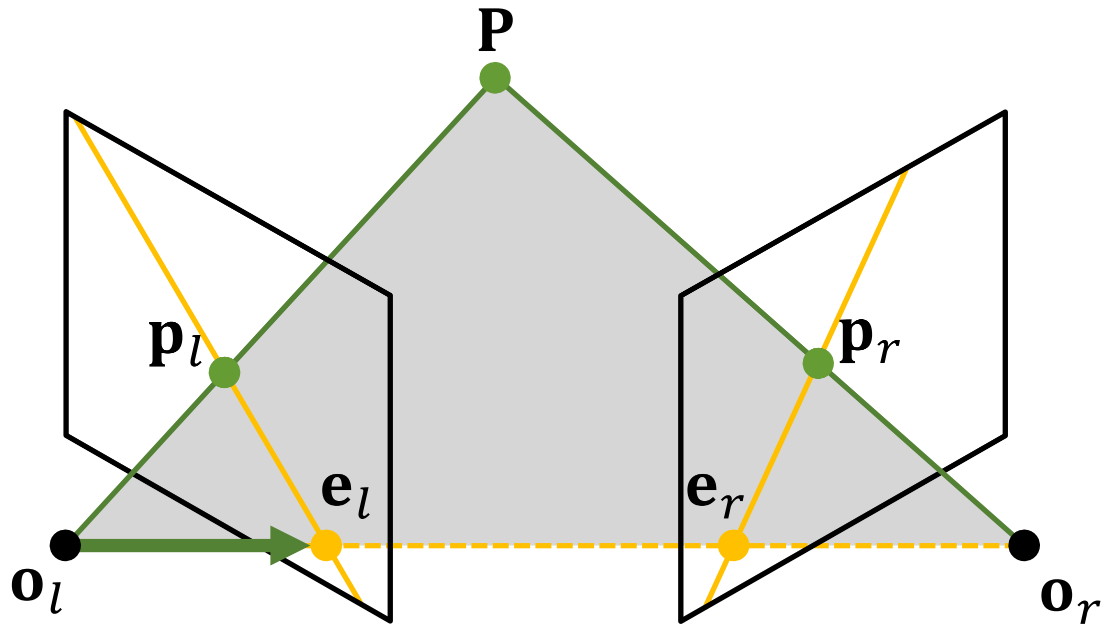
$$\mathbf{F} = \mathbf{M}_{\text{int},r}^{-\text{T}} \mathbf{E} \mathbf{M}_{\text{int},l}^{-\mathbf{1}}$$

$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\text{int},r}^{-\text{T}} \mathbf{E} \mathbf{M}_{\text{int},l}^{-\mathbf{1}} \mathbf{e}_l$$



$$\mathbf{F} = \mathbf{M}_{\text{int},r}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\text{int},l}^{-\mathbf{1}}$$

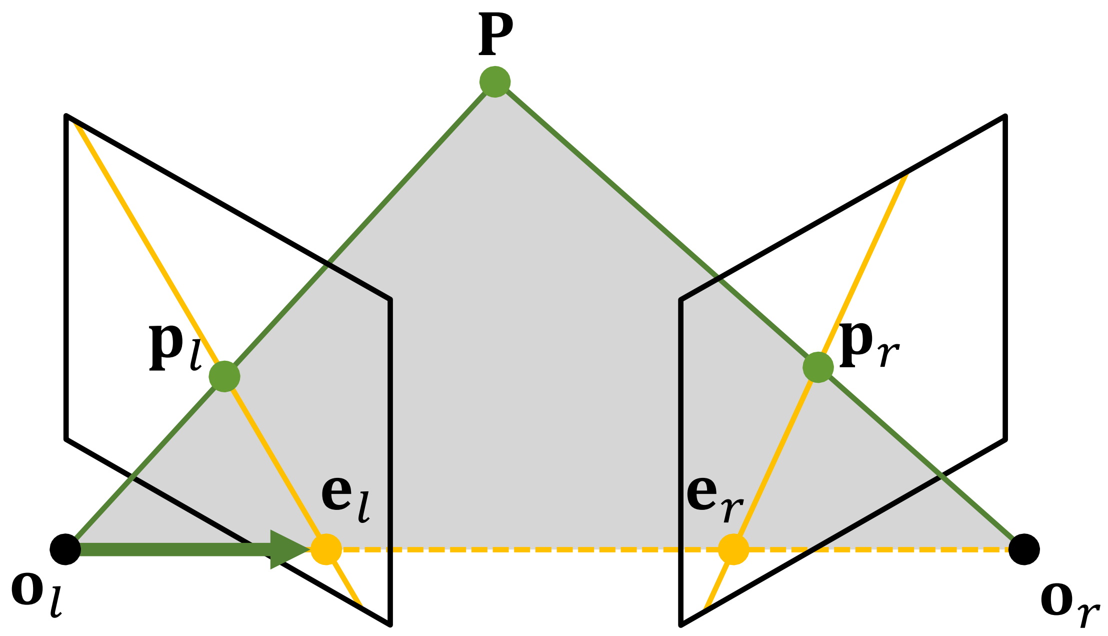
$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\text{int},r}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\text{int},l}^{-\mathbf{1}} \mathbf{e}_l$$



$$\mathbf{F} = \mathbf{M}_{\text{int},r}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\text{int},l}^{-\mathbf{1}}$$

$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\text{int},r}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\text{int},l}^{-\mathbf{1}} \mathbf{e}_l$$

代入 $\mathbf{E} = \mathbf{R}[\mathbf{T}_\times]$

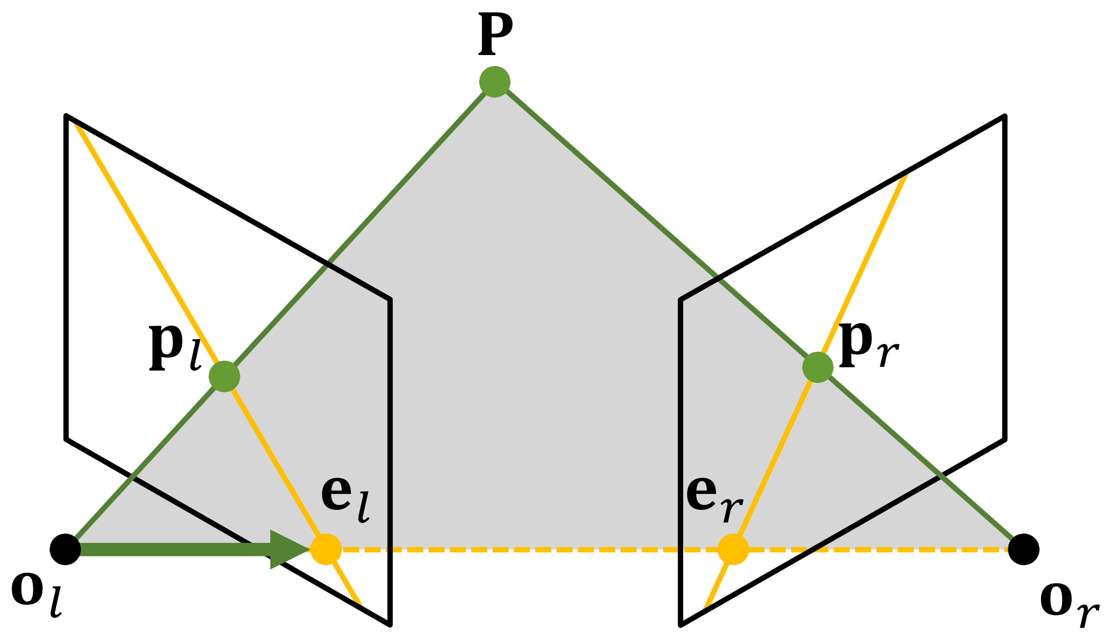


$$\mathbf{F} = \mathbf{M}_{\text{int},r}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\text{int},l}^{-\mathbf{1}}$$

$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\text{int},r}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\text{int},l}^{-\mathbf{1}} \mathbf{e}_l$$

代入 $\mathbf{E} = \mathbf{R}[\mathbf{T}_\times]$

$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\text{int},r}^{-\mathrm{T}} \mathbf{R}[\mathbf{T}_\times] \mathbf{M}_{\text{int},l}^{-\mathbf{1}} \mathbf{e}_l$$



$$\mathbf{F} = \mathbf{M}_{\text{int},r}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\text{int},l}^{-\mathbf{1}}$$

$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\text{int},r}^{-\mathrm{T}} \mathbf{E} \mathbf{M}_{\text{int},l}^{-\mathbf{1}} \mathbf{e}_l$$

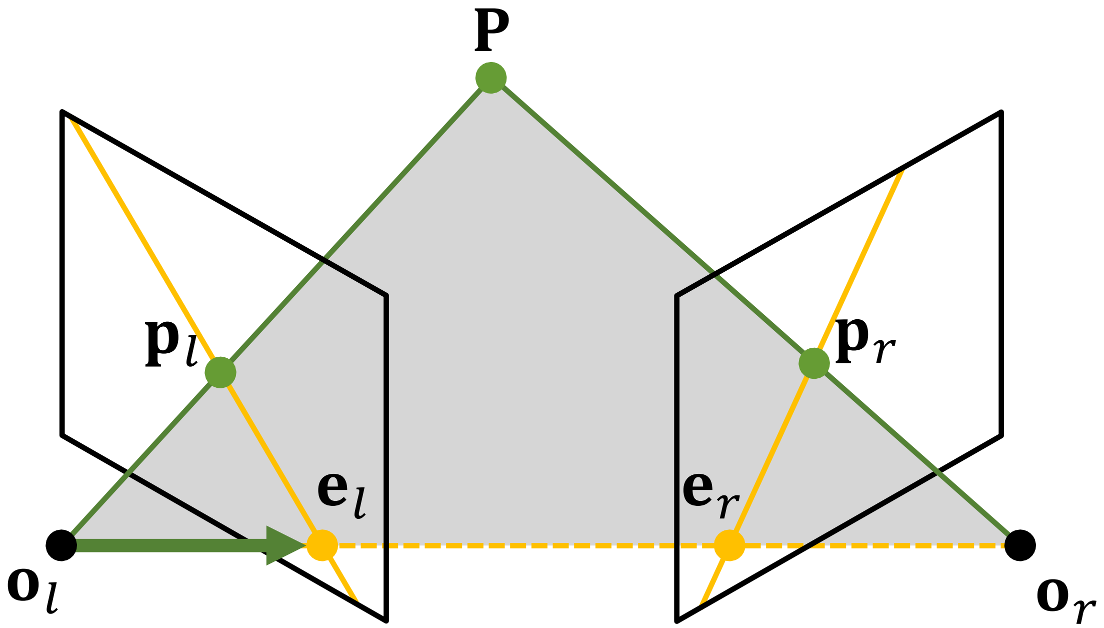
代入 $\mathbf{E} = \mathbf{R}[\mathbf{T}_\times]$

$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\text{int},r}^{-\mathrm{T}} \mathbf{R}[\mathbf{T}_\times] \mathbf{M}_{\text{int},l}^{-\mathbf{1}} \mathbf{e}_l$$

给定3×1向量$\mathbf{a} = (a_1, a_2, a_3)^{\mathrm{T}}$和$\mathbf{b} = (b_1, b_2, b_3)^{\mathrm{T}}$

$$\mathbf{a} \times \mathbf{b} = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$

$$= [\mathbf{a}_\times]\mathbf{b}$$

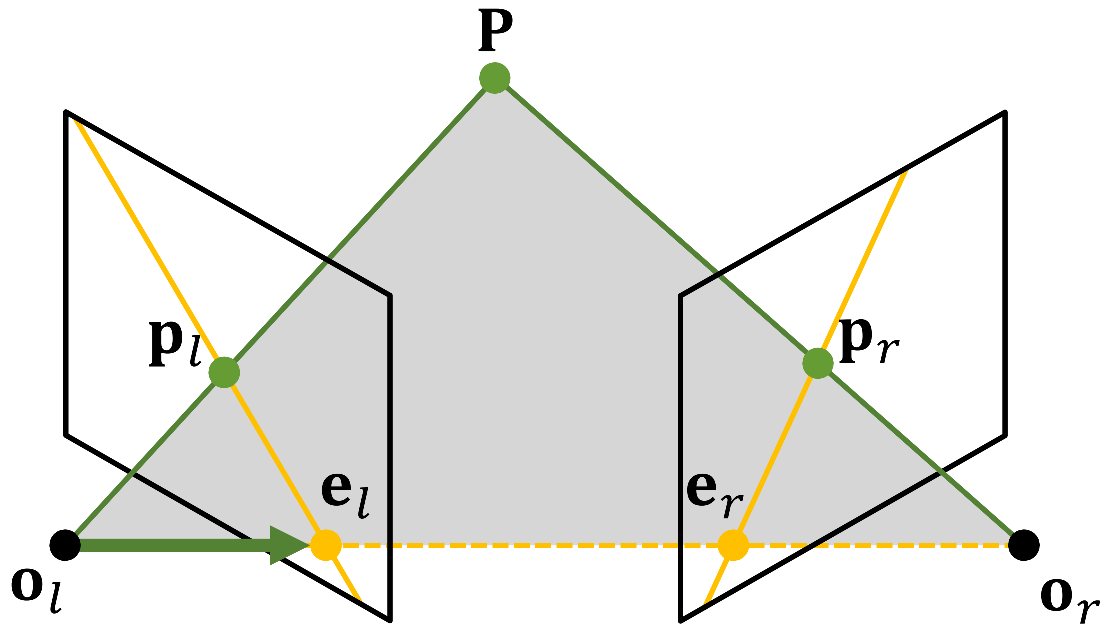


$$\mathbf{F} = \mathbf{M}_{\text{int},r}^{-\text{T}}\mathbf{E}\mathbf{M}_{\text{int},l}^{-\mathbf{1}}$$

$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\text{int},r}^{-\text{T}}\mathbf{E}\mathbf{M}_{\text{int},l}^{-\mathbf{1}}\mathbf{e}_l$$

代入 $\mathbf{E} = \mathbf{R}[\mathbf{T}_\times]$

$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\text{int},r}^{-\text{T}}\mathbf{R}[\mathbf{T}_\times]\mathbf{M}_{\text{int},l}^{-\mathbf{1}}\mathbf{e}_l$$



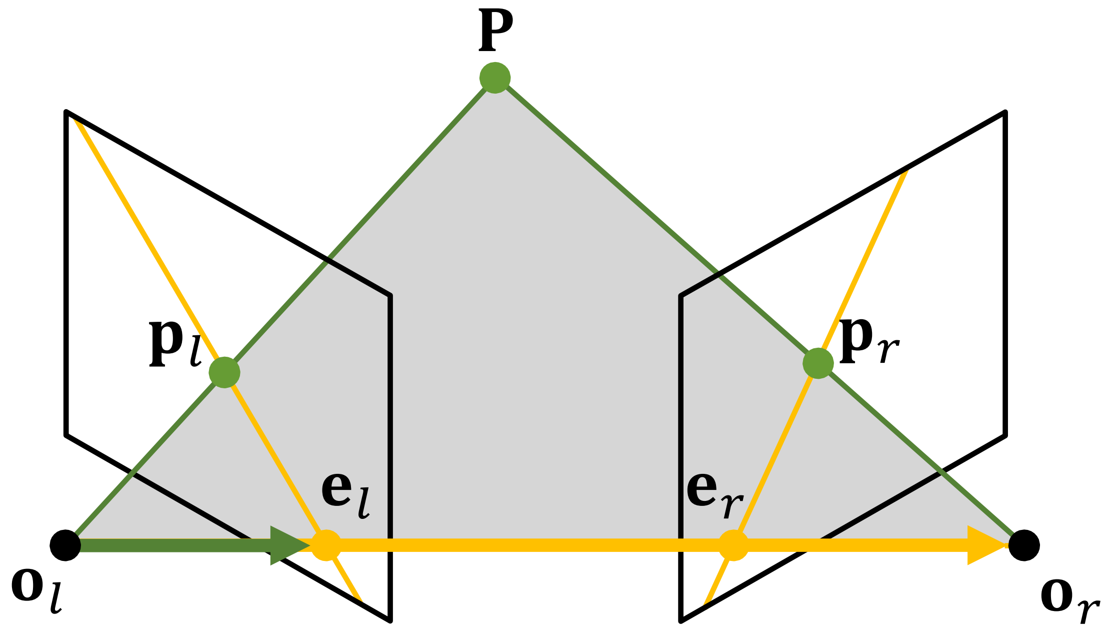
$$\mathbf{F} = \mathbf{M}_{\text{int},r}^{-\mathrm{T}}\mathbf{E}\mathbf{M}_{\text{int},l}^{-\mathbf{1}}$$

$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\text{int},r}^{-\mathrm{T}}\mathbf{E}\mathbf{M}_{\text{int},l}^{-\mathbf{1}}\mathbf{e}_l$$

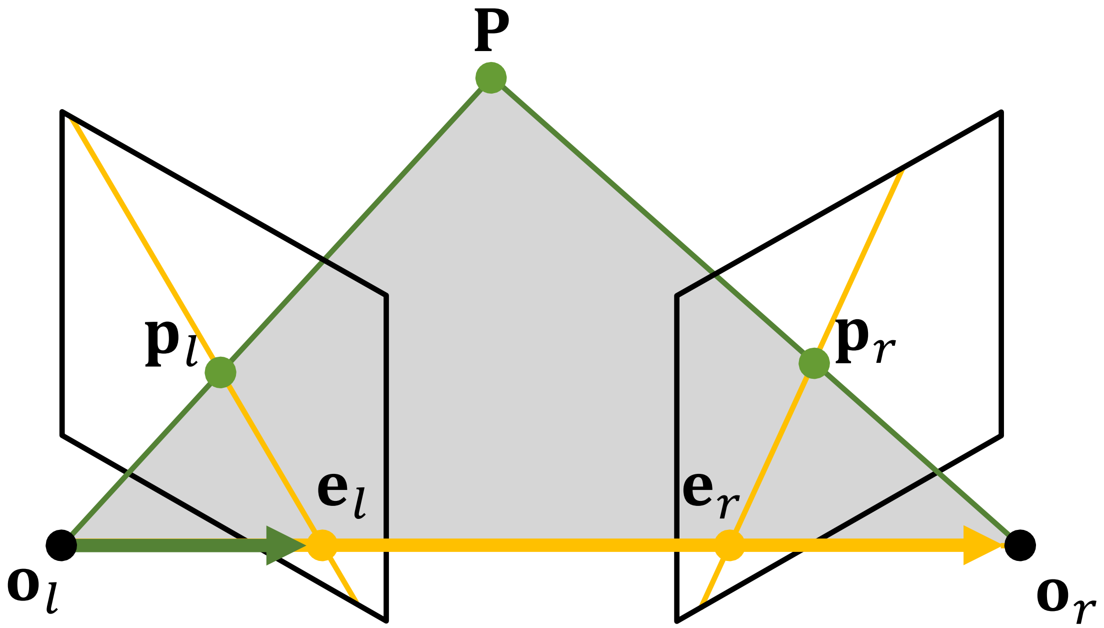
代入 $\mathbf{E} = \mathbf{R}[\mathbf{T}_\times]$

$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\text{int},r}^{-\mathrm{T}}\mathbf{R}[\mathbf{T}_\times]\mathbf{M}_{\text{int},l}^{-\mathbf{1}}\mathbf{e}_l$$

$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\text{int},r}^{-\mathrm{T}}\mathbf{R}\left(\mathbf{T} \times \mathbf{M}_{\text{int},l}^{-\mathbf{1}}\mathbf{e}_l\right)$$



$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\mathrm{int},r}^{-\mathrm{T}}\mathbf{R}\left(\mathbf{T} \times \mathbf{M}_{\mathrm{int},l}^{-\mathbf{1}}\mathbf{e}_l\right)$$



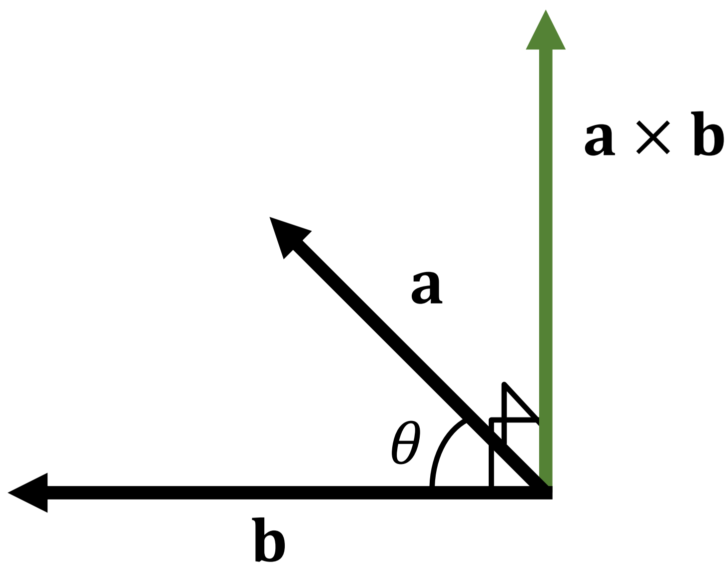
$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\mathrm{int},r}^{-\mathrm{T}}\mathbf{R}\left(\mathbf{T} \times \mathbf{M}_{\mathrm{int},l}^{-1}\mathbf{e}_l\right)$$

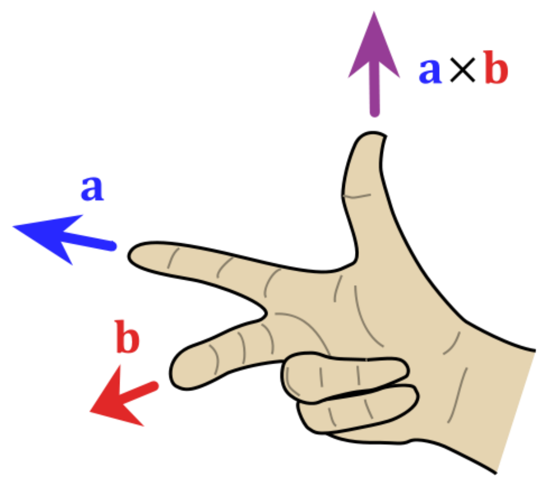
叉积是多少?

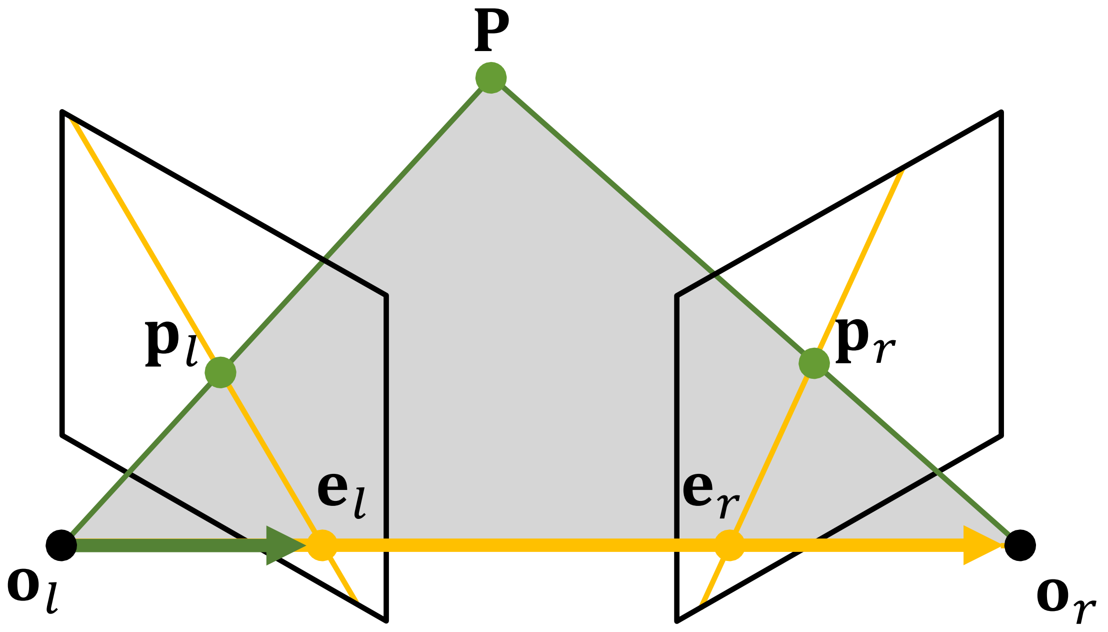
定义：三维空间中两个向量**a**和**b**的**叉积**$\mathbf{a} \times \mathbf{b}$是与**a**和**b**都垂直的向量，可以定义为：

$$\mathbf{a} \times \mathbf{b} = \|\mathbf{a}\| \|\mathbf{b}\| \sin(\theta)\, \mathbf{n}$$

其中θ表示**a**和**b**的夹角（$0 \leq \theta \leq \pi$）。$\|\mathbf{a}\|$和$\|\mathbf{b}\|$是向量**a**和**b**的模长，而**n**则是一个与**a**、**b**所构成的平面垂直的单位向量，方向由右手定则决定。

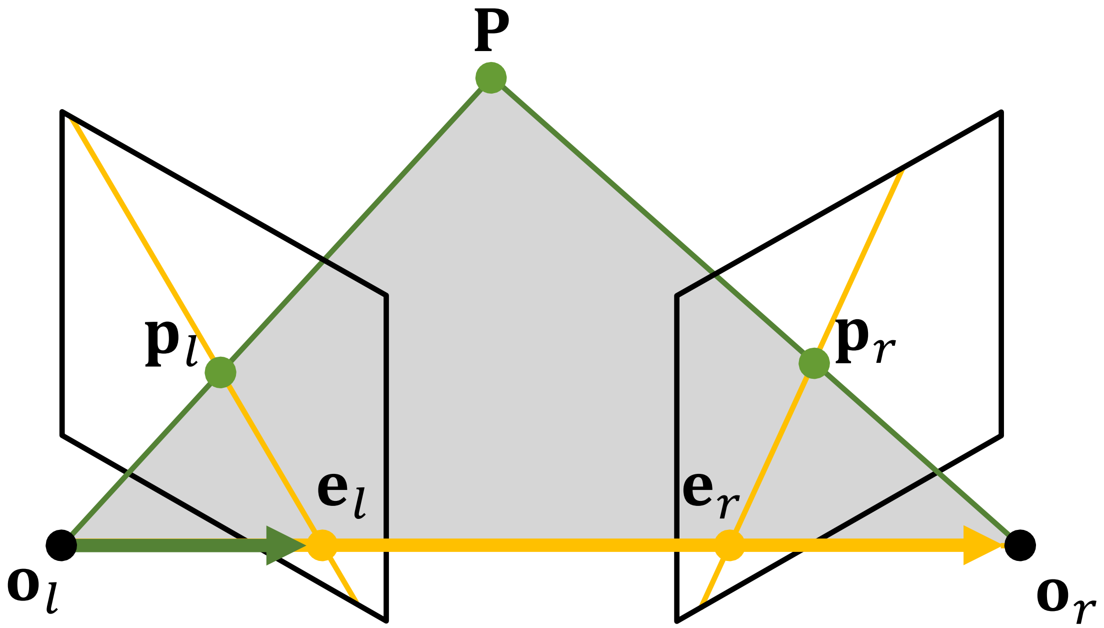






$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\mathrm{int},r}^{-\mathrm{T}}\mathbf{R}\left(\mathbf{T} \times \mathbf{M}_{\mathrm{int},l}^{-1}\mathbf{e}_l\right)$$

叉积是多少?



$$\mathbf{F}\mathbf{e}_l = \mathbf{M}_{\text{int},r}^{-\text{T}}\mathbf{R}\left(\mathbf{T} \times \mathbf{M}_{\text{int},l}^{-\mathbf{1}}\mathbf{e}_l\right)$$

$$\mathbf{F}\mathbf{e}_l = \mathbf{0}$$

基础矩阵约束：$\tilde{\mathbf{p}}_r^{\mathrm{T}}\mathbf{F}\tilde{\mathbf{p}}_l = \mathbf{0}$

备注：

基础矩阵捕获两个视点间的投影几何

与场景结构无关

可以通过匹配对应点计算，而不需要事先知道相机的内参和外参

在齐次坐标系中是确定的，但不约束尺度因子

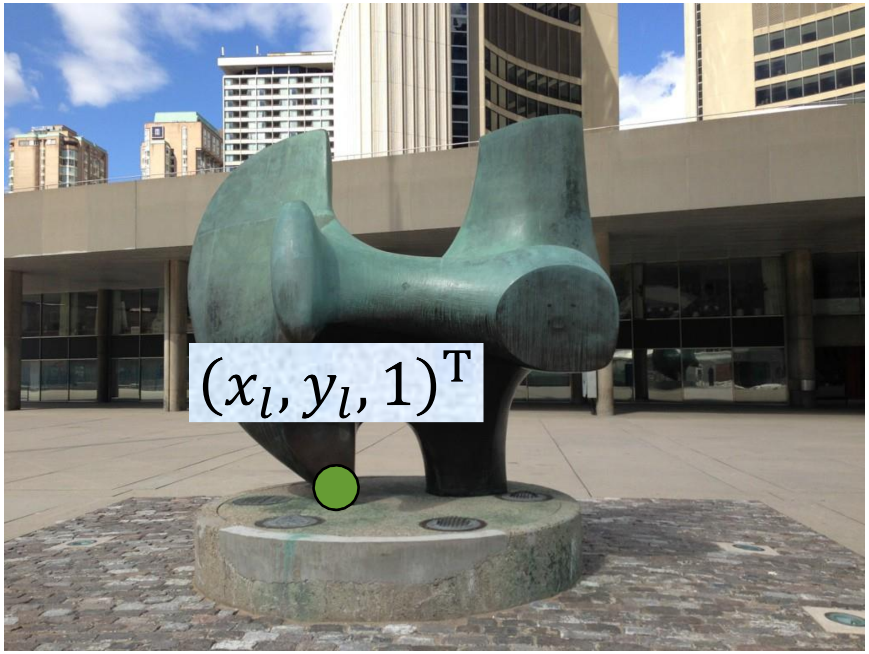
秩为2

极点是左、右零空间：$\mathbf{F}\mathbf{e}_l = \mathbf{0}$和$\mathbf{e}_r^{\mathrm{T}}\mathbf{F} = \mathbf{0}$

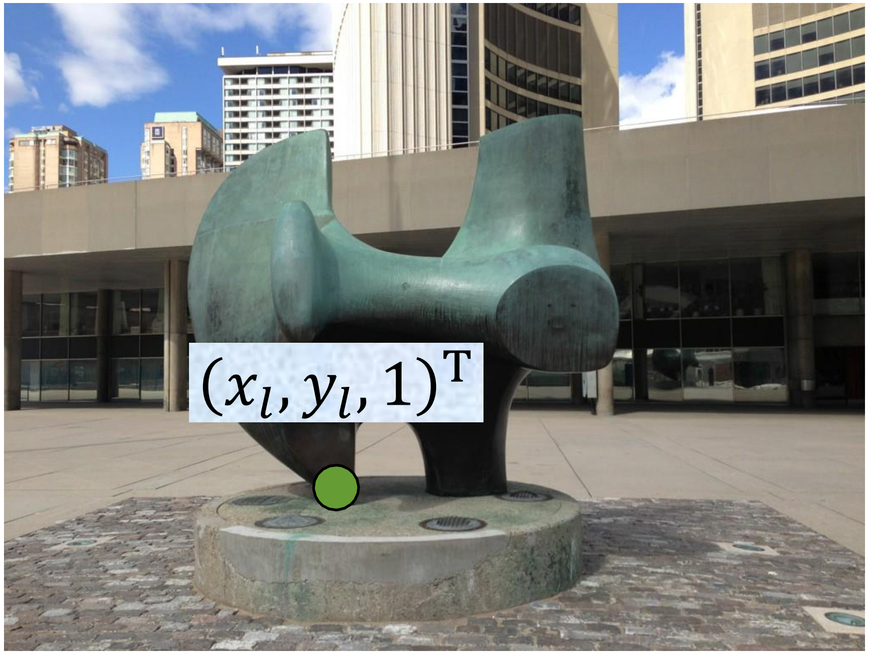
F有什么用呢?

$$\mathbf{p}_r^{\mathrm{T}} \mathbf{F} \mathbf{p}_l = 0$$

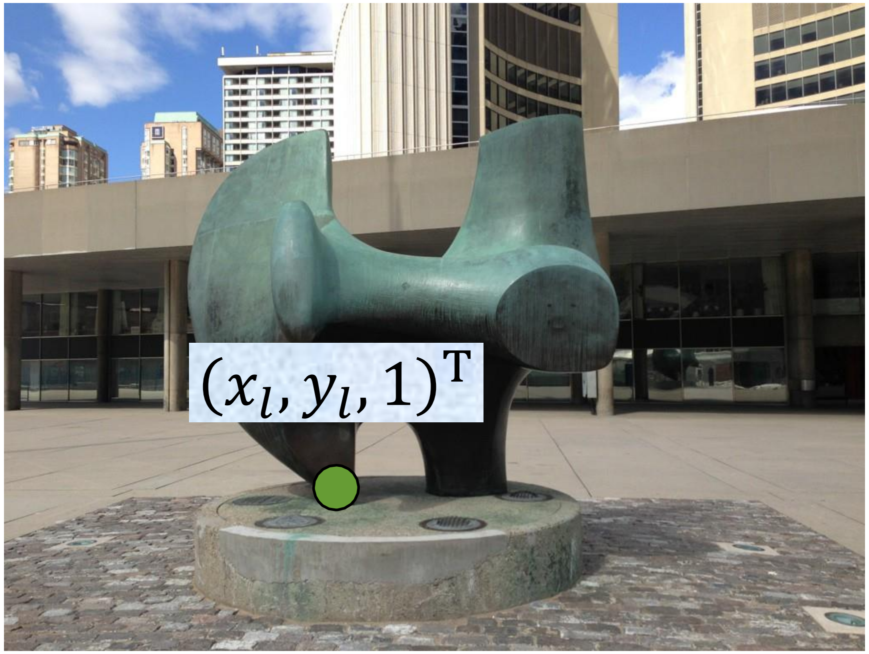


$$\mathbf{p}_r^T \mathbf{F} \mathbf{p}_l = 0$$



$$\mathbf{p}_r^{\mathrm{T}} \mathbf{F} \mathbf{p}_l = 0$$

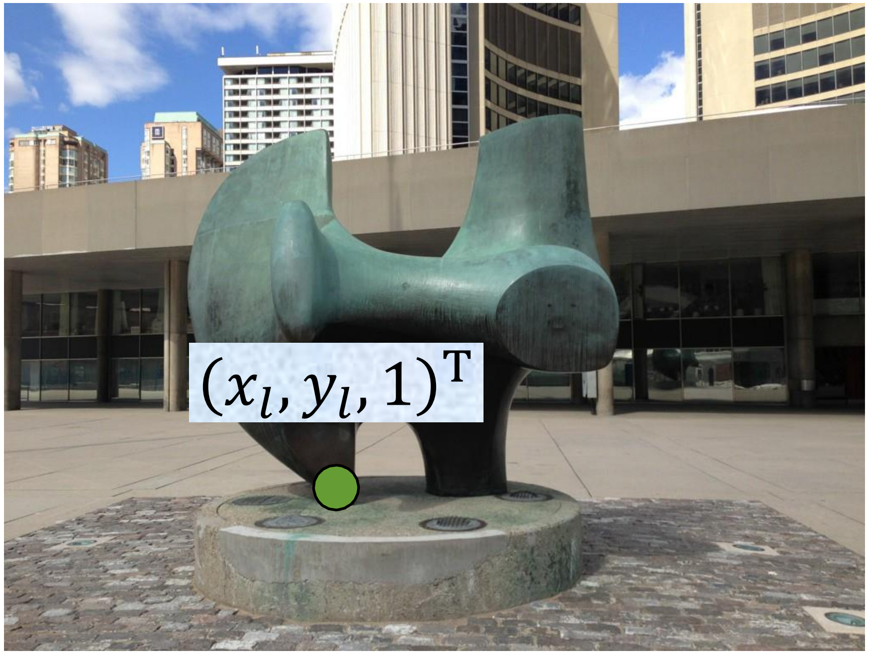
$$(x_r, y_r, 1)\mathbf{F}(x_l, y_l, 1)^{\mathrm{T}} = 0$$



$$\mathbf{p}_r^\mathrm{T}\mathbf{F}\mathbf{p}_l = 0$$

$$(x_r, y_r, 1)\mathbf{F}(x_l, y_l, 1)^\mathrm{T} = 0$$

令 $\mathbf{k} = \mathbf{F}(x_l, y_l, 1)^\mathrm{T}$

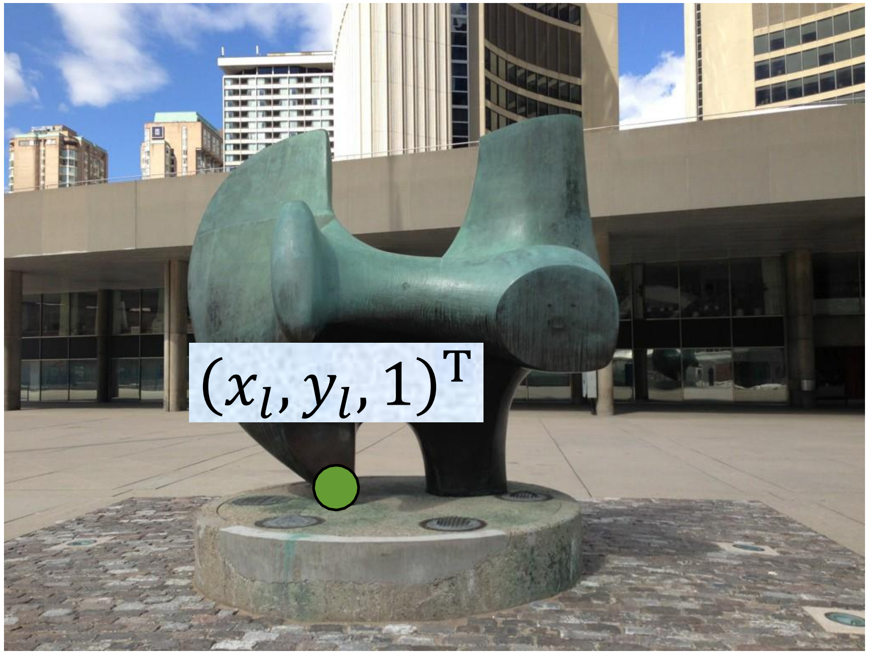


$$\mathbf{p}_r^{\mathrm{T}} \mathbf{F} \mathbf{p}_l = 0$$

$$(x_r, y_r, 1)\mathbf{F}(x_l, y_l, 1)^{\mathrm{T}} = 0$$

令 $\mathbf{k} = \mathbf{F}(x_l, y_l, 1)^{\mathrm{T}}$

$$(x_r, y_r, 1)\mathbf{k} = 0$$



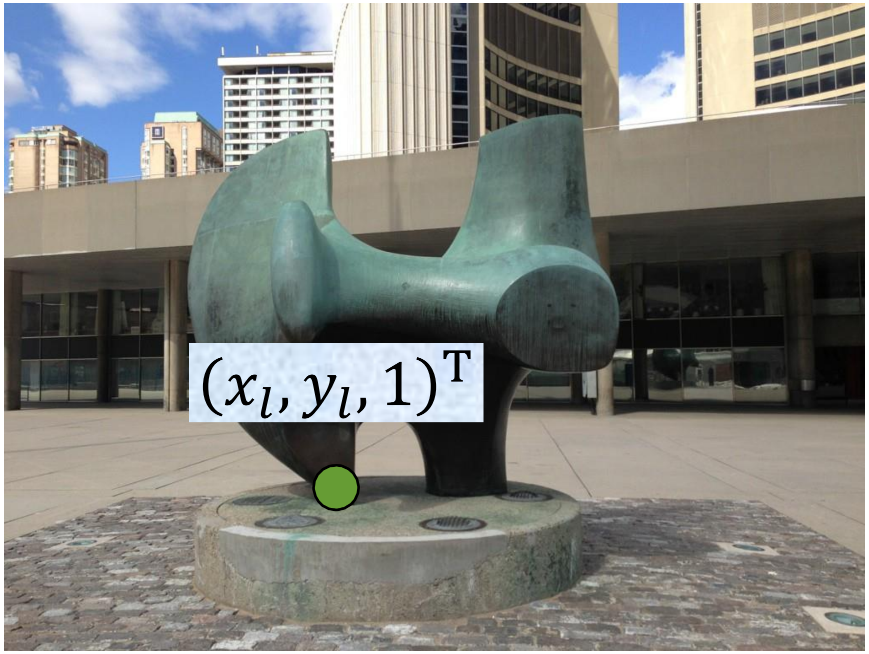
$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$(x_r, y_r, 1)\mathbf{F}(x_l, y_l, 1)^{\mathrm{T}} = 0$$

令 $\mathbf{k} = \mathbf{F}(x_l, y_l, 1)^{\mathrm{T}}$

$$(x_r, y_r, 1)\mathbf{k} = 0$$

是不是看起来很眼熟?



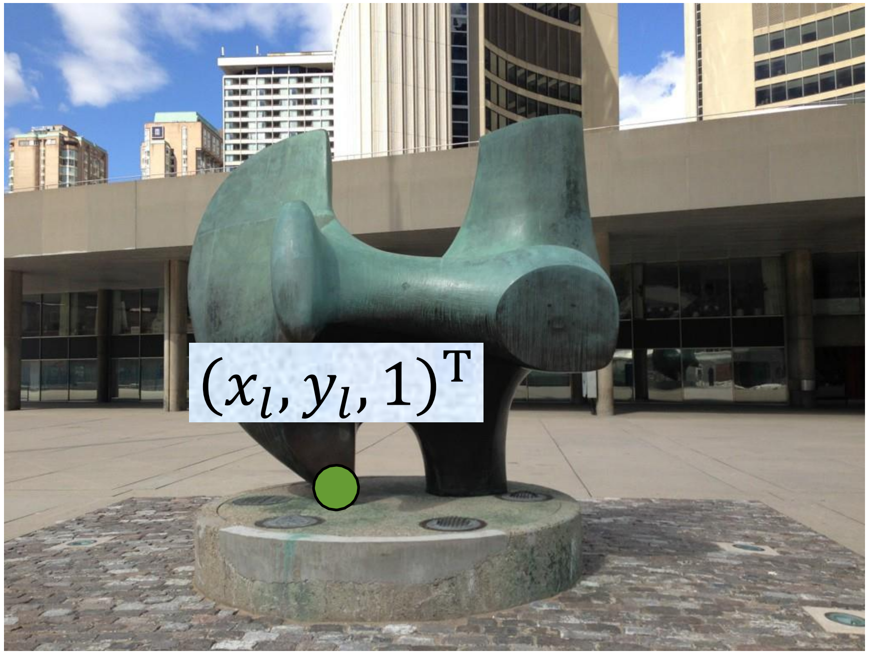
$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$(x_r, y_r, 1)\mathbf{F}(x_l, y_l, 1)^{\mathrm{T}} = 0$$

令 $\mathbf{k} = \mathbf{F}(x_l, y_l, 1)^{\mathrm{T}}$

$$(x_r, y_r, 1)\mathbf{k} = 0$$

直线方程：$Ax + By + C = 0$



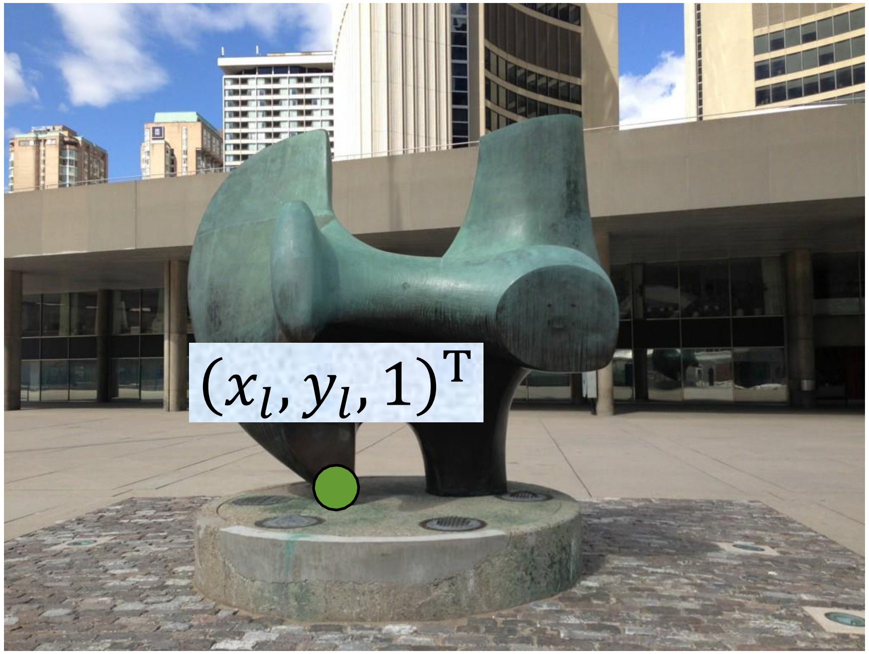
$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$(x_r, y_r, 1)\mathbf{F}(x_l, y_l, 1)^{\mathrm{T}} = 0$$

令 $\mathbf{k} = \mathbf{F}(x_l, y_l, 1)^{\mathrm{T}}$

$$(x_r, y_r, 1)\mathbf{k} = 0$$

极线： $\mathbf{l}' = \mathbf{F}\mathbf{x}_l$



$$\mathbf{p}_r^\mathrm{T}\mathbf{F}\mathbf{p}_l = 0$$

$$(x_r, y_r, 1)\mathbf{F}(x_l, y_l, 1)^\mathrm{T} = 0$$

令 $\mathbf{k} = \mathbf{F}(x_l, y_l, 1)^\mathrm{T}$

$$(x_r, y_r, 1)\mathbf{k} = 0$$

极线：$\mathbf{l}' = \mathbf{F}\mathbf{x}_l$

搜索空间被缩小到一条直线

如果该点是极点会发生什么?

如果该点是极点会发生什么?

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$


如果该点是极点会发生什么？

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$(x_r, y_r, 1)\mathbf{F}\mathbf{e}_l = 0$$

基础矩阵约束：$\tilde{\mathbf{p}}_r^{\mathrm{T}}\mathbf{F}\tilde{\mathbf{p}}_l = \mathbf{0}$

备注：

基础矩阵捕获两个视点间的投影几何

与场景结构无关

可以通过匹配对应点计算，而不需要事先知道相机的内参和外参

在齐次坐标系中是确定的，但不约束尺度因子

秩为2

极点是左、右零空间：$\mathbf{F}\mathbf{e}_l = \mathbf{0}$和$\mathbf{e}_r^{\mathrm{T}}\mathbf{F} = \mathbf{0}$

如果该点是极点会发生什么?

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$(x_r, y_r, 1)\mathbf{F}\mathbf{e}_l = 0$$

如果该点是极点会发生什么?

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$(x_r, y_r, 1)\mathbf{F}\mathbf{e}_l = 0$$

$$(x_r, y_r, 1)\mathbf{0} = 0$$

如果该点是极点会发生什么？

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$(x_r, y_r, 1)\mathbf{F}\mathbf{e}_l = 0$$

$$(x_r, y_r, 1)\mathbf{0} = 0$$

这个约束意味着什么？

如果该点是极点会发生什么?

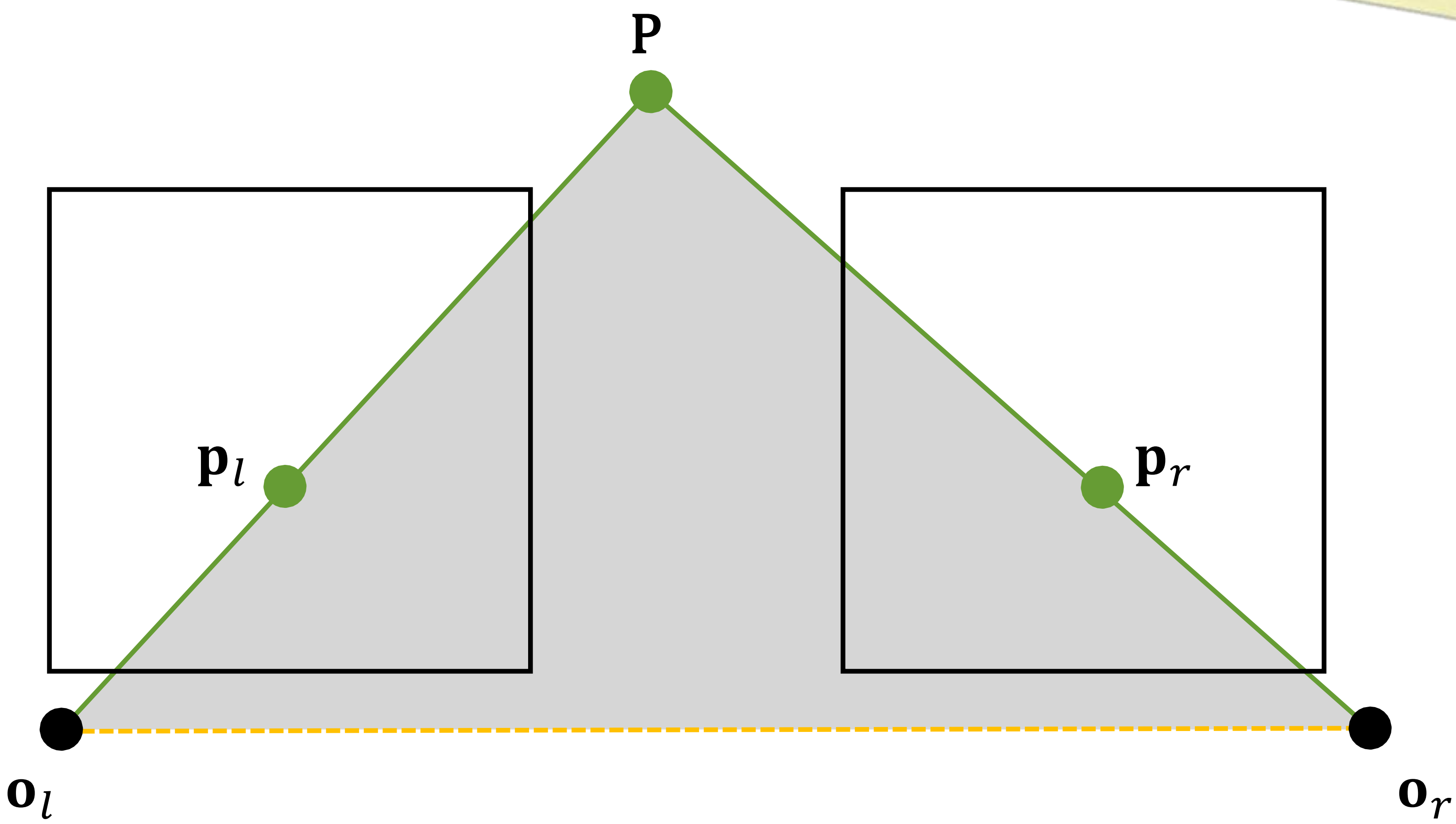
$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

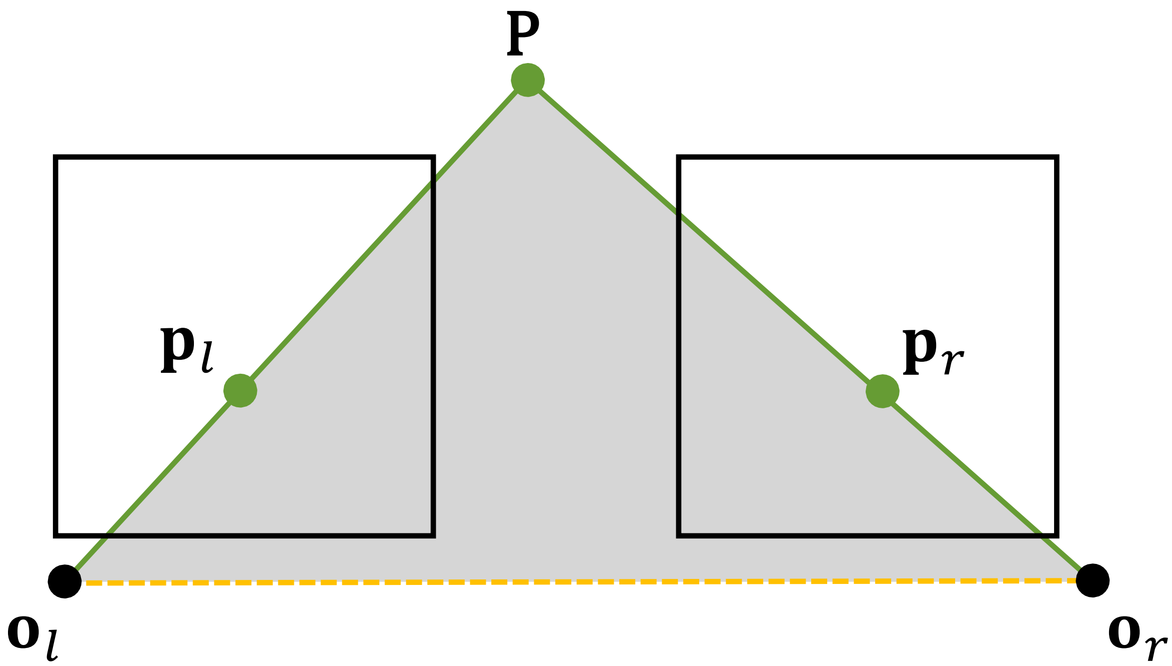
$$(x_r, y_r, 1)\mathbf{F}\mathbf{e}_l = 0$$

$$(x_r, y_r, 1)\mathbf{0} = 0$$

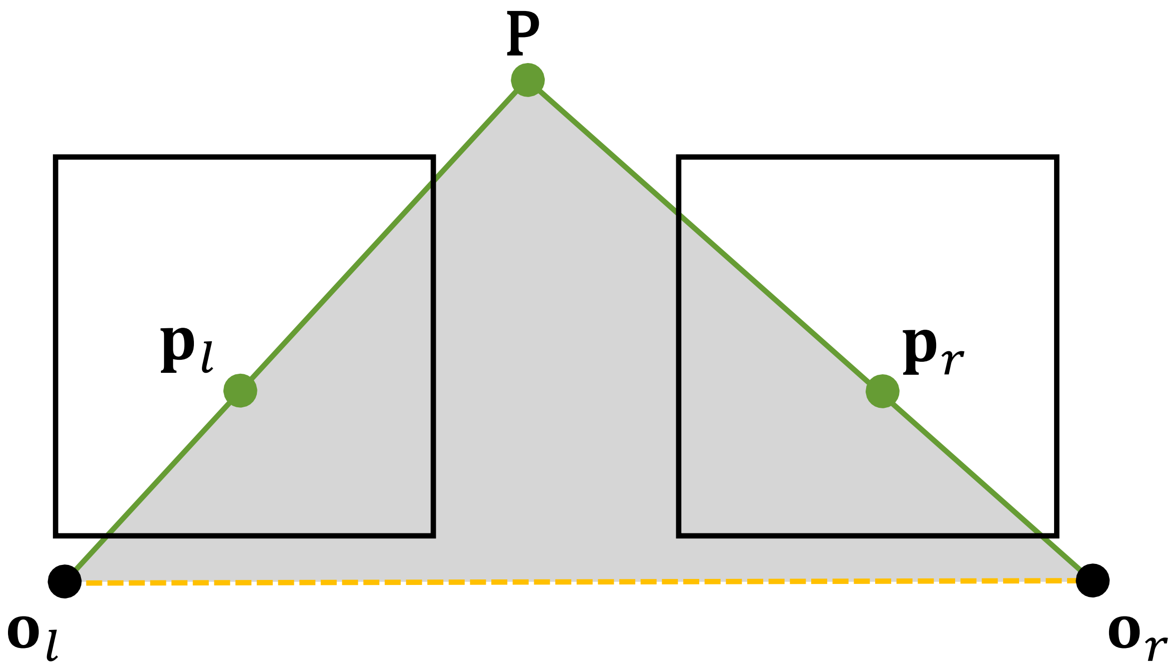
所有极线都穿过极点

特殊情况
本质矩阵
平行相机



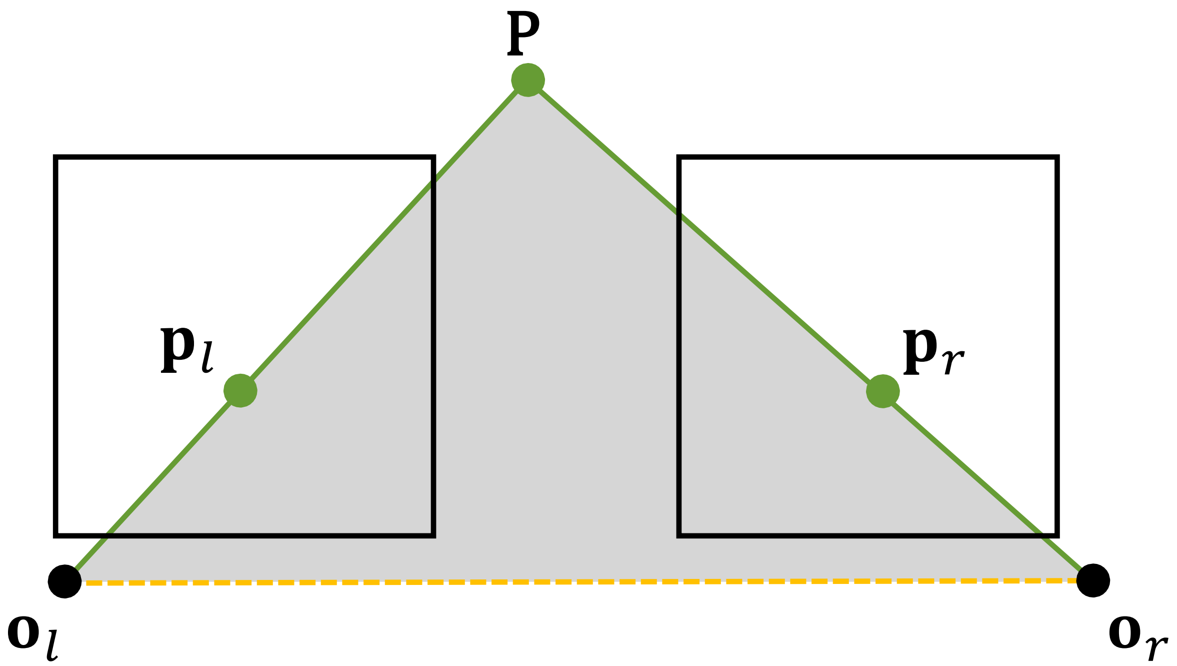


两个相机之间的相对旋转是多少?



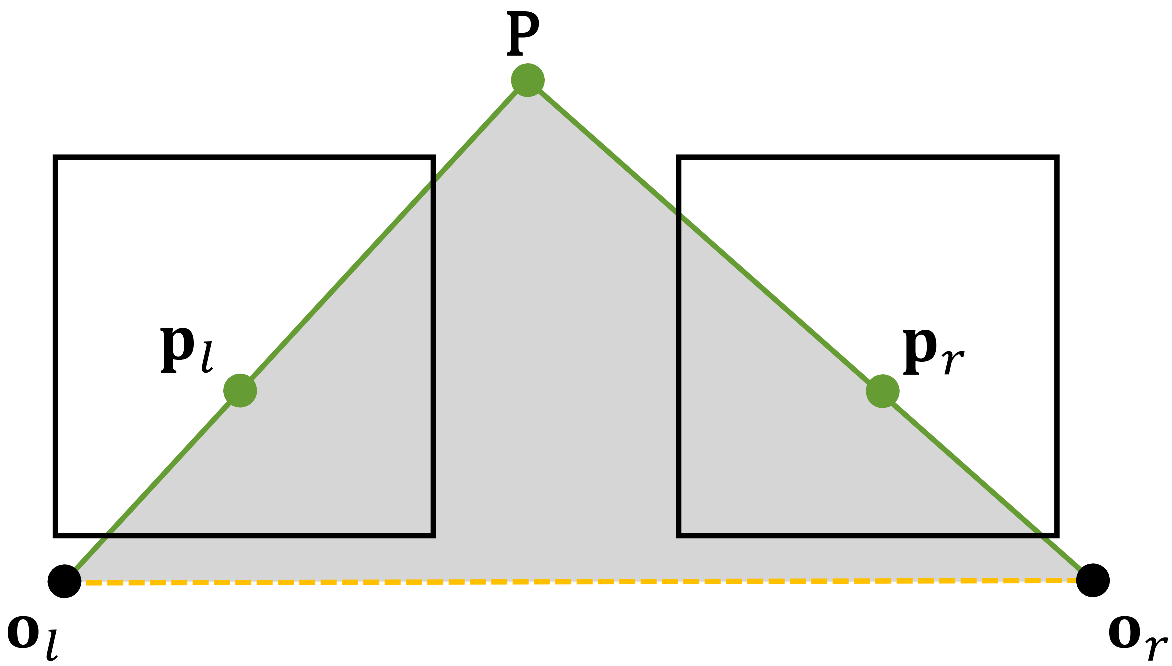
$$\mathbf{R} = \mathbf{I}_{3\times3}$$

两个相机之间的相对旋转是多少?



$\mathbf{R} = \mathbf{I}_{3\times 3}$

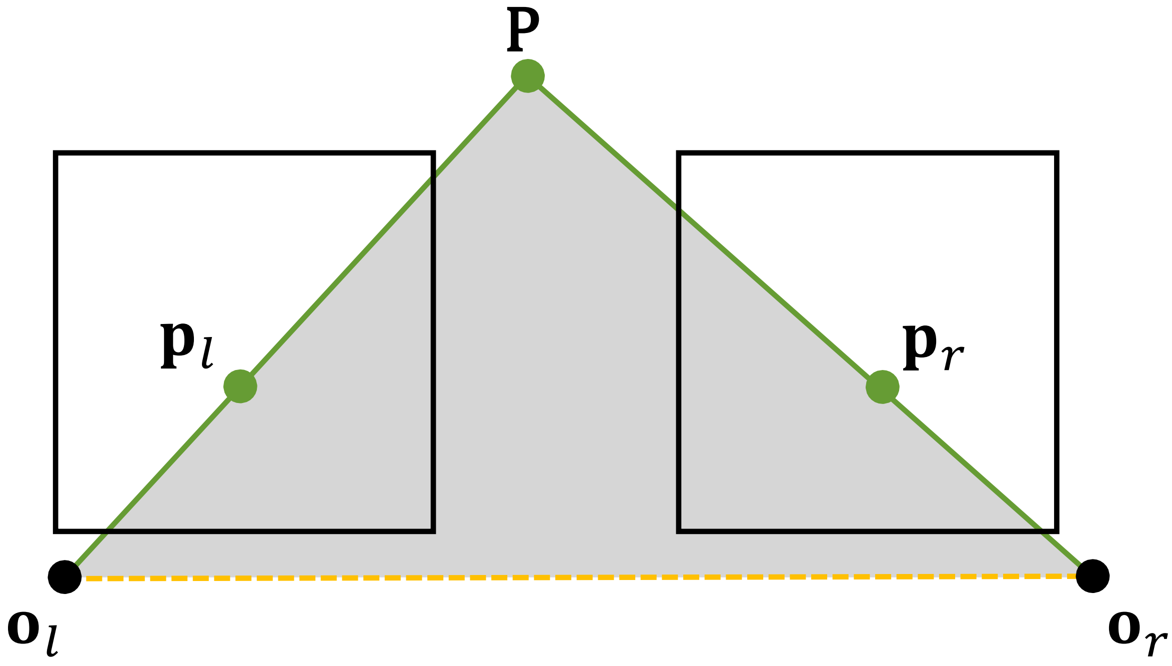
两个相机之间的相对平移是多少?



$\mathbf{R} = \mathbf{I}_{3\times 3}$

$\mathbf{T} = (B, 0,0)^{\mathrm{T}}$

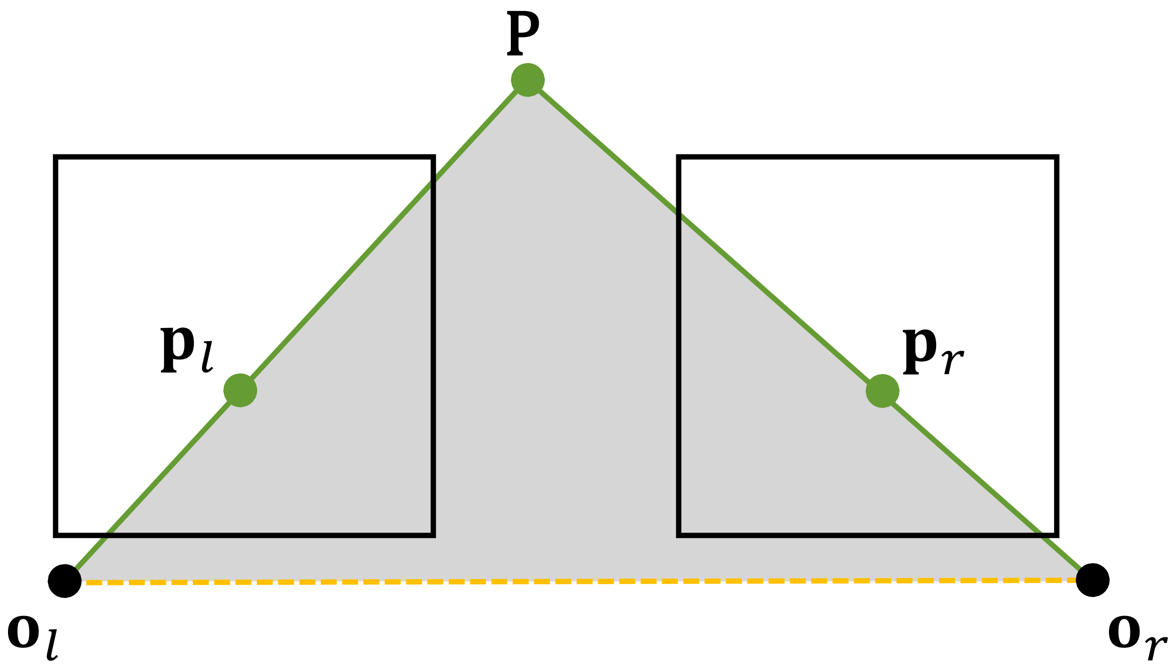
两个相机之间的相对平移是多少?



$$\mathbf{R} = \mathbf{I}_{3\times 3}$$

$$\mathbf{T} = (B, 0, 0)^{\mathrm{T}}$$

$$\mathbf{p}_r^{\mathrm{T}} \mathbf{E} \mathbf{p}_l = 0$$

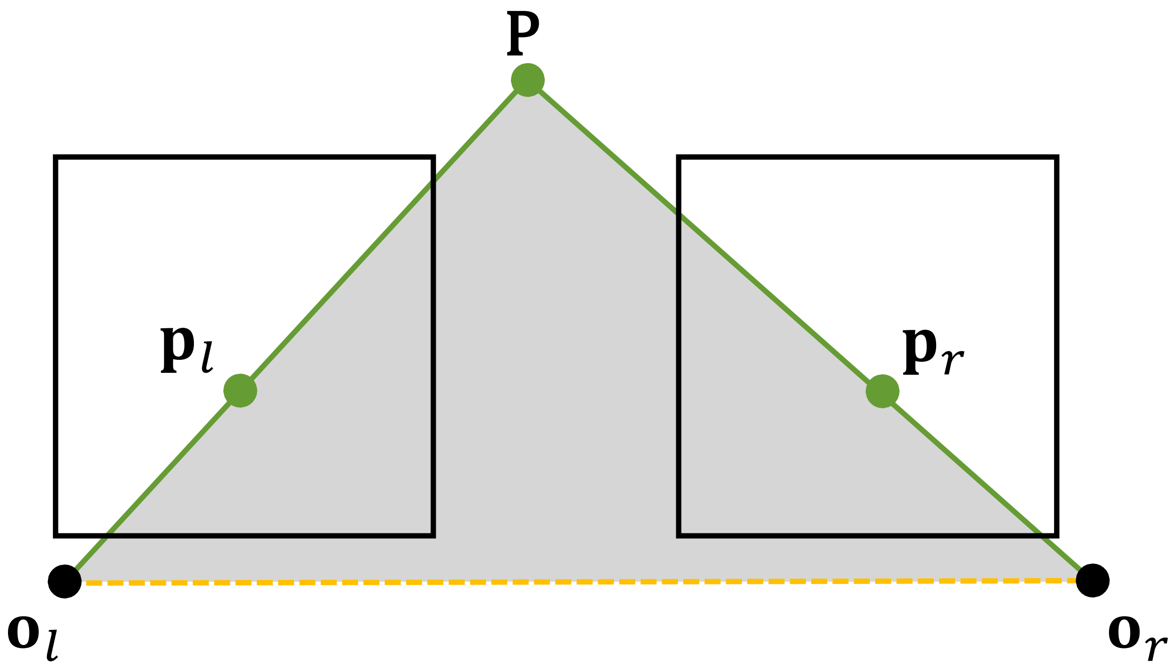


$$\mathbf{R} = \mathbf{I}_{3\times 3}$$

$$\mathbf{T} = (B, 0, 0)^{\mathrm{T}}$$

$$\mathbf{p}_r^{\mathrm{T}} \mathbf{E} \mathbf{p}_l = 0$$

$$\mathbf{p}_r^{\mathrm{T}} \mathbf{R} [\mathbf{T}_\times] \mathbf{p}_l = 0$$



$$\mathbf{R} = \mathbf{I}_{3\times 3}$$

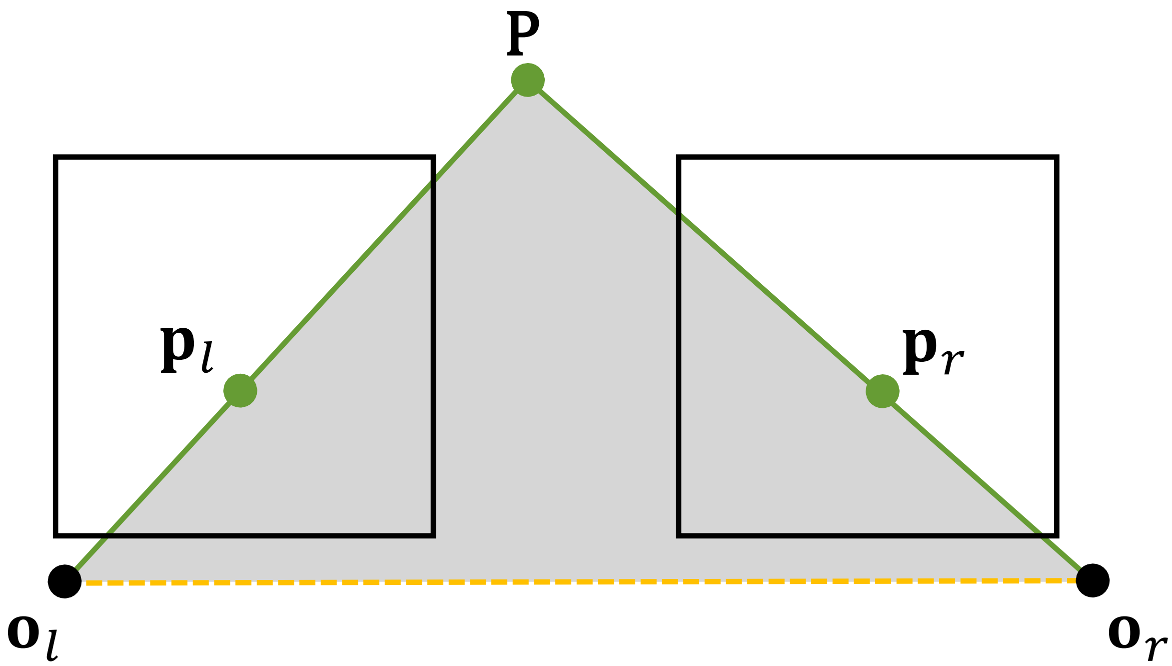
$$\mathbf{T} = (B, 0,0)^{\mathrm{T}}$$

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{E}\mathbf{p}_l = 0$$

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{R}[\mathbf{T}_\times]\mathbf{p}_l = 0$$

$$(x_r, y_r, 1)\mathbf{R}[\mathbf{T}_\times](x_l, y_l, 1)^{\mathrm{T}} = 0$$

化简



$$\mathbf{R} = \mathbf{I}_{3\times 3}$$

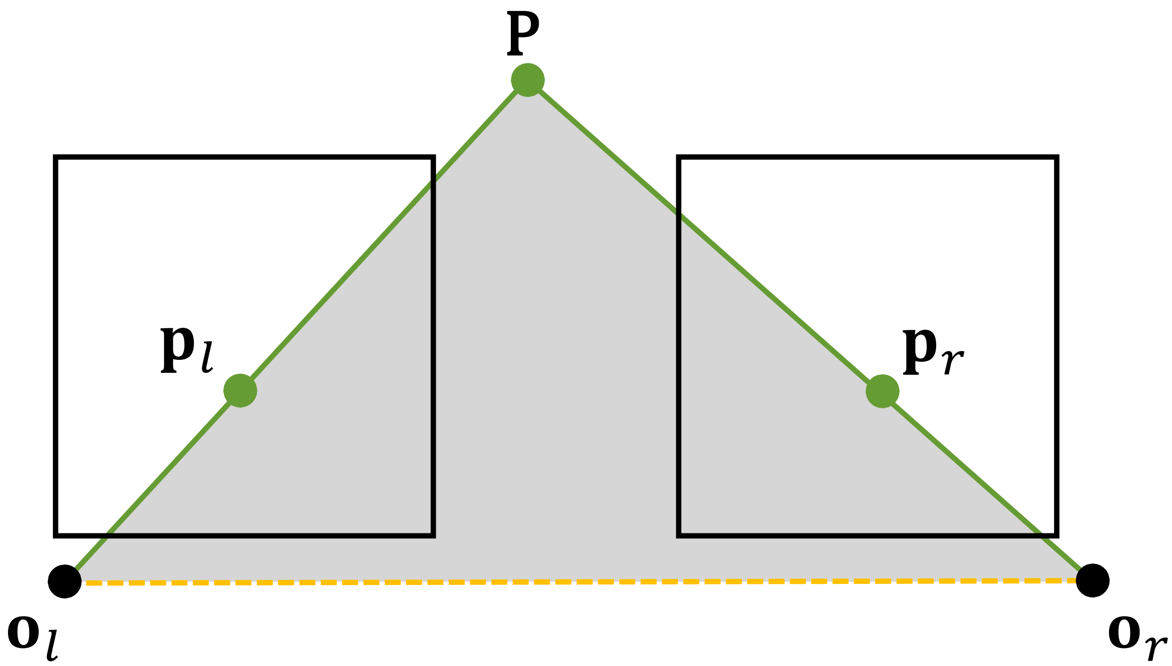
$$\mathbf{T} = (B, 0,0)^{\mathrm{T}}$$

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{E}\mathbf{p}_l = 0$$

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{R}[\mathbf{T}_\times]\mathbf{p}_l = 0$$

$$(x_r, y_r, 1)\mathbf{R}[\mathbf{T}_\times](x_l, y_l, 1)^{\mathrm{T}} = 0$$

化简



$$\mathbf{R} = \mathbf{I}_{3\times 3}$$

$$\mathbf{T} = (B, 0, 0)^{\mathrm{T}}$$

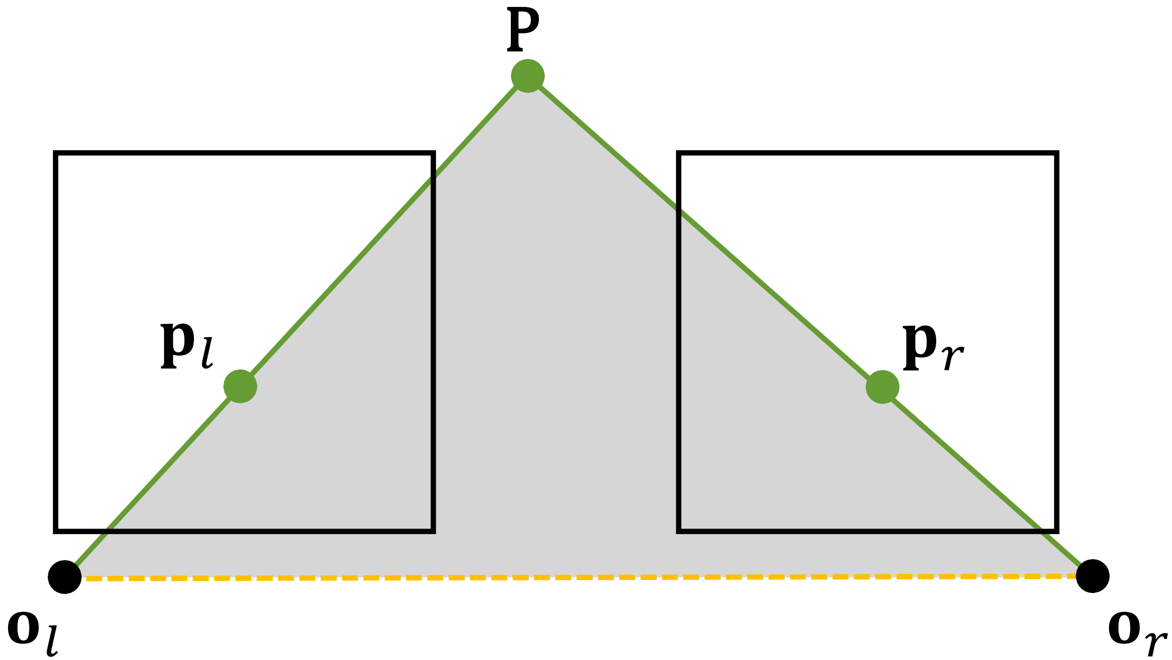
$$\mathbf{p}_r^{\mathrm{T}}\mathbf{E}\mathbf{p}_l = 0$$

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{R}[\mathbf{T}_\times]\mathbf{p}_l = 0$$

$$(x_r, y_r, 1)\cancel{\mathbf{R}}[\mathbf{T}_\times](x_l, y_l, 1)^{\mathrm{T}} = 0$$

化简

$$(x_r, y_r, 1)[\mathbf{T}_\times](x_l, y_l, 1)^{\mathrm{T}} = 0$$

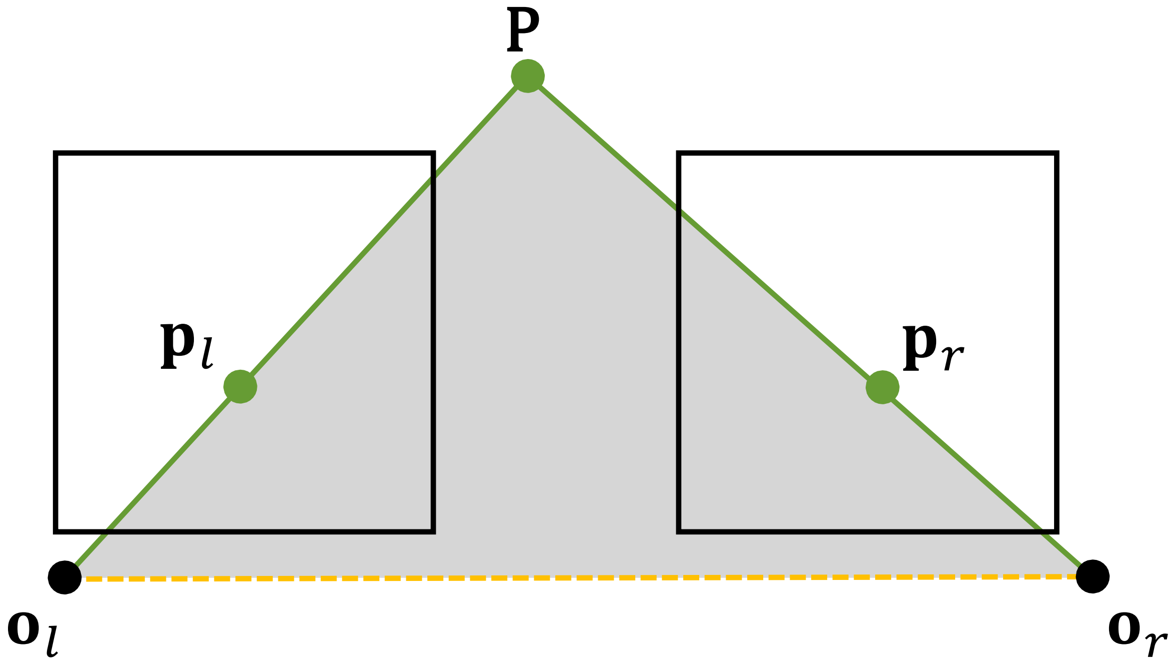


$$\mathbf{R} = \mathbf{I}_{3\times 3}$$

$$\mathbf{T} = (B, 0,0)^{\mathrm{T}}$$

$$\mathbf{p}_r^{\mathrm{T}} \mathbf{E} \mathbf{p}_l = 0$$

$$(x_r, y_r, 1)[\mathbf{T}_\times](x_l, y_l, 1)^{\mathrm{T}} = 0$$



$$\mathbf{R} = \mathbf{I}_{3\times 3}$$

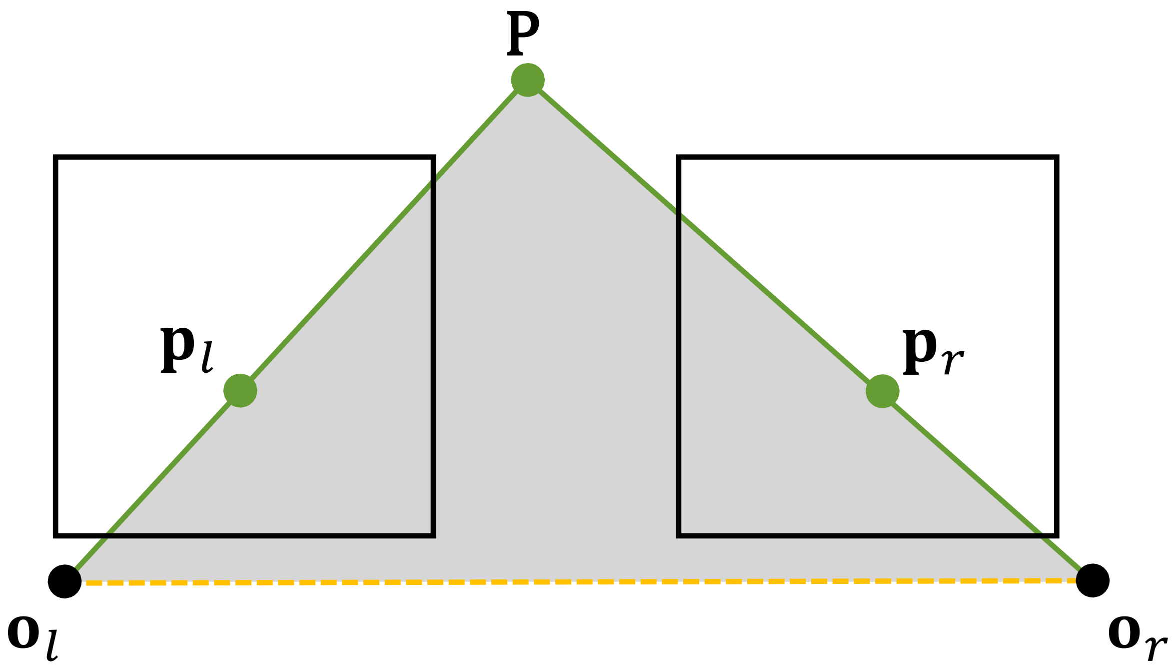
$$\mathbf{T} = (B, 0, 0)^{\mathrm{T}}$$

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{E}\mathbf{p}_l = 0$$

$$(x_r, y_r, 1)[\mathbf{T}_\times](x_l, y_l, 1)^{\mathrm{T}} = 0$$

展开并化简

$$[\mathbf{T}_\times] = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -B \\ 0 & B & 0 \end{pmatrix}$$



$$\mathbf{R} = \mathbf{I}_{3\times3}$$

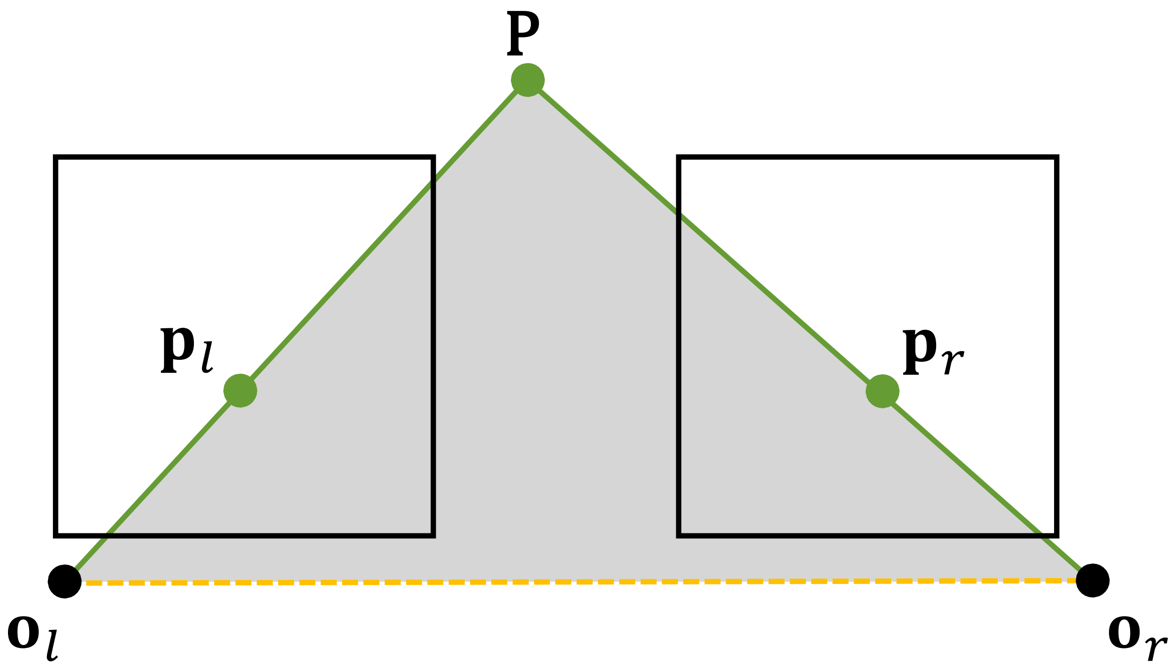
$$\mathbf{T} = (B, 0, 0)^{\mathrm{T}}$$

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{E}\mathbf{p}_l = 0$$

$$(x_r, y_r, 1)[\mathbf{T}_\times](x_l, y_l, 1)^{\mathrm{T}} = 0$$

展开并化简

$$y_r = y_l$$



$$\mathbf{R} = \mathbf{I}_{3\times 3}$$

$$\mathbf{T} = (B, 0,0)^{\mathrm{T}}$$

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{E}\mathbf{p}_l = 0$$

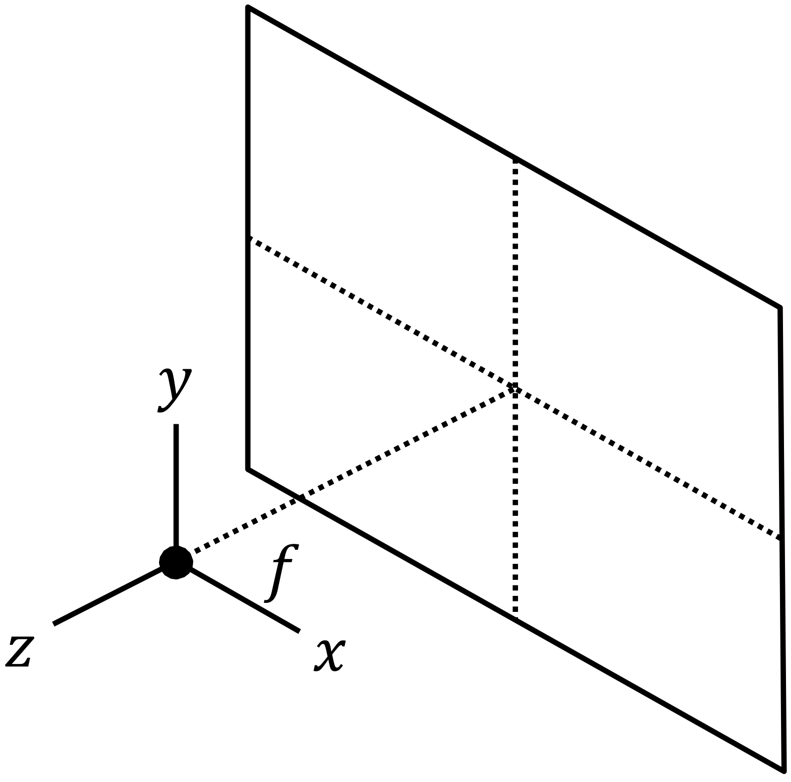
$$(x_r, y_r, 1)[\mathbf{T}_\times](x_l, y_l, 1)^{\mathrm{T}} = 0$$

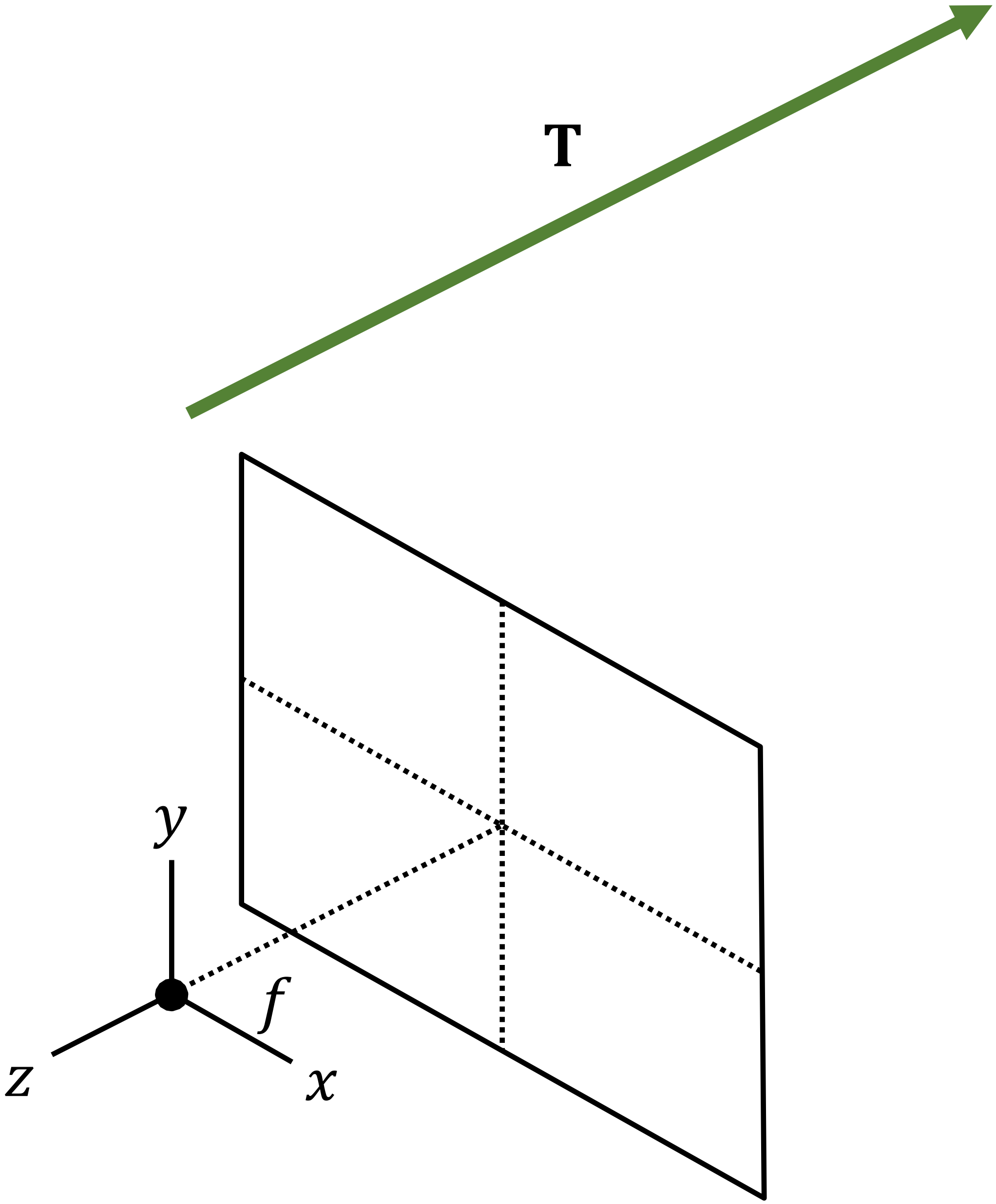
展开并化简

$$y_r = y_l$$

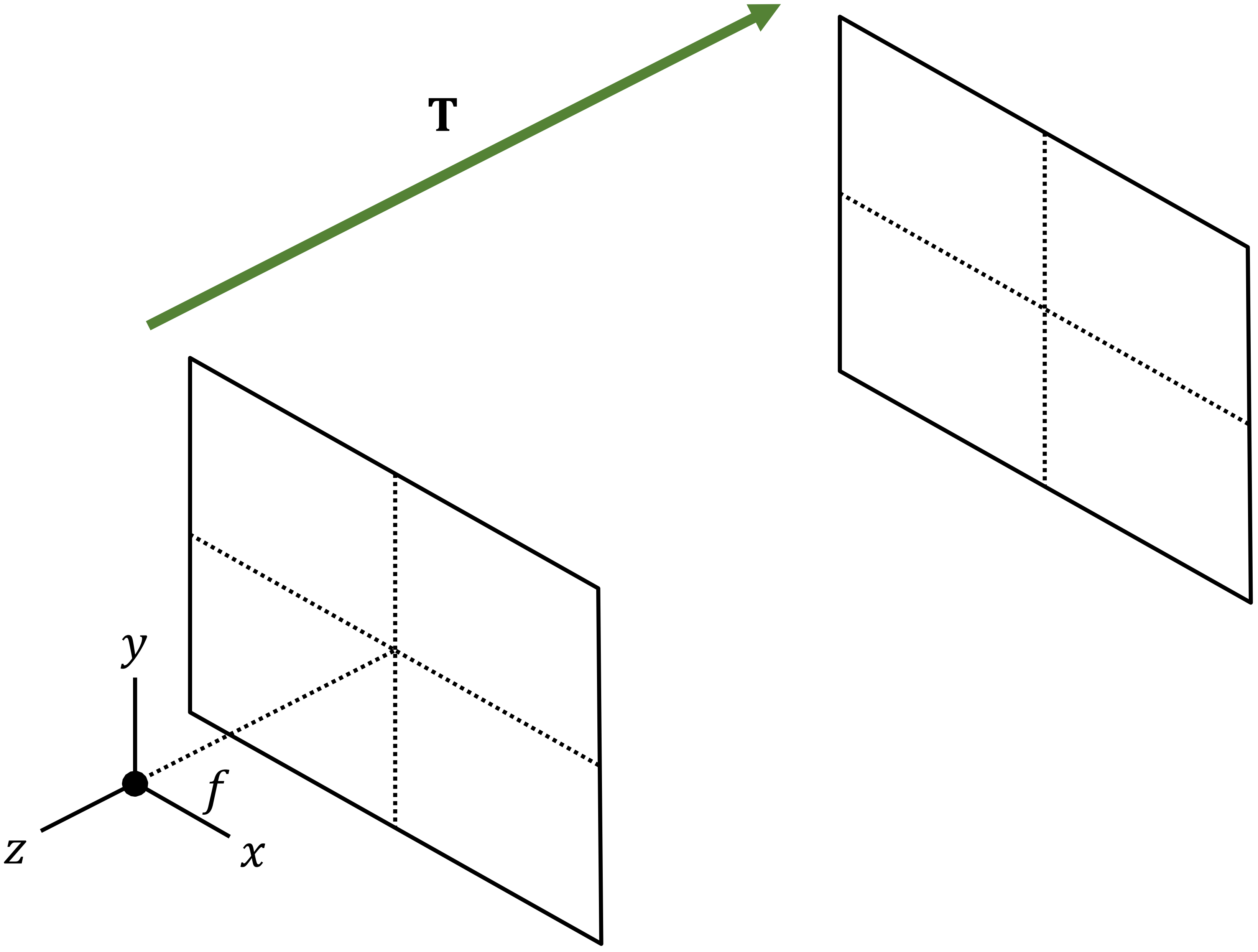
对应点位于相同一条水平直线上

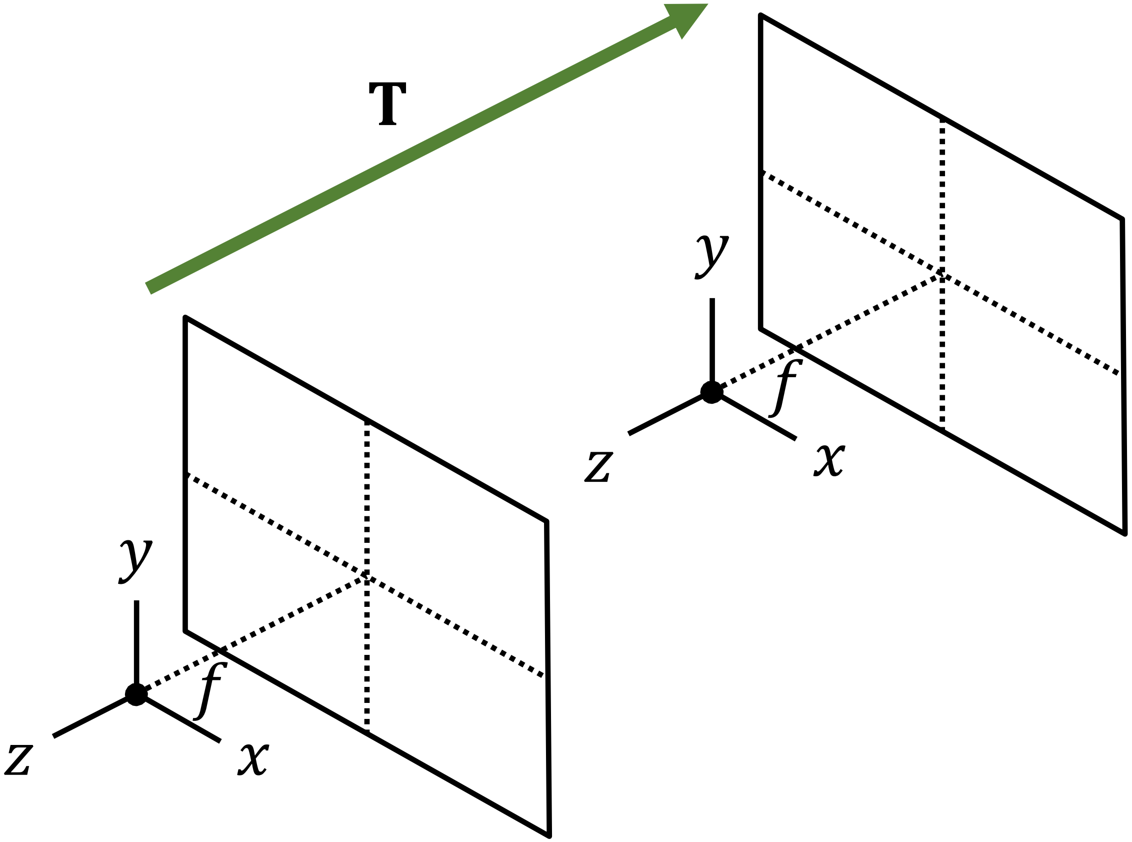
特殊情况
本质矩阵
前向运动



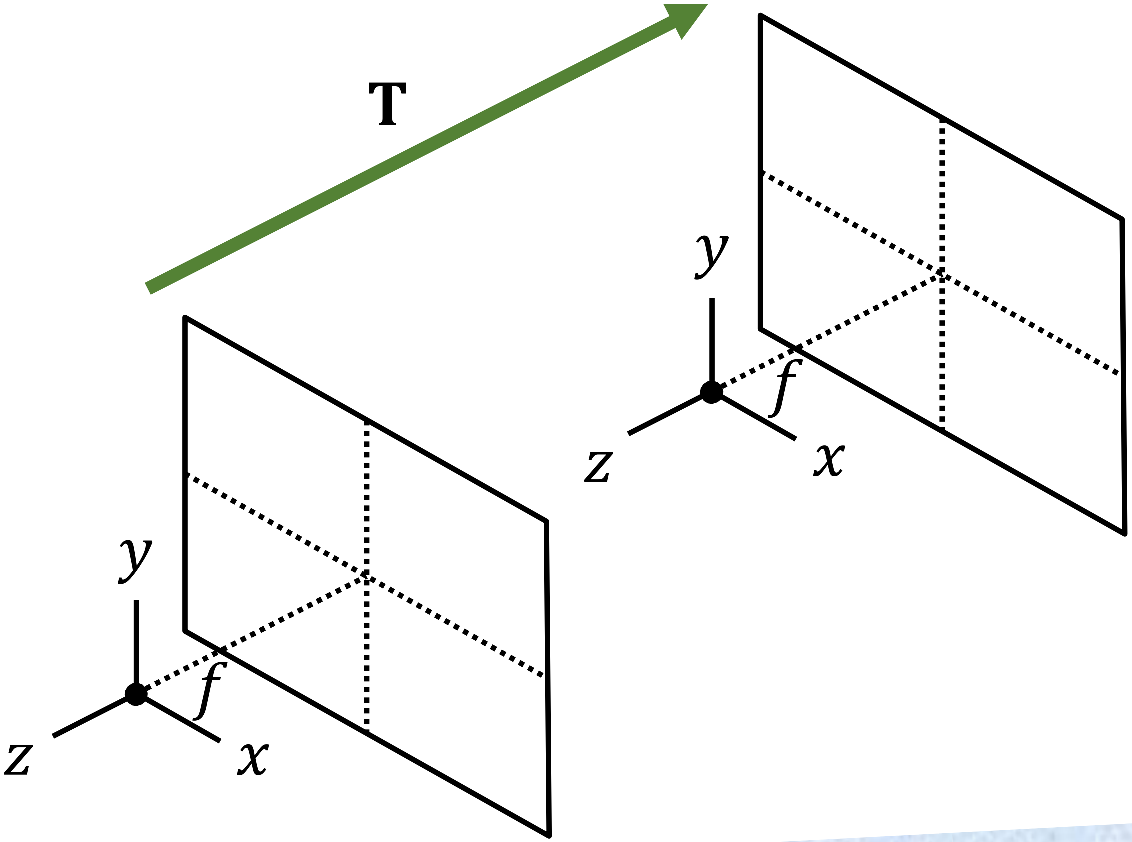


特殊情况
本质矩阵
前向运动

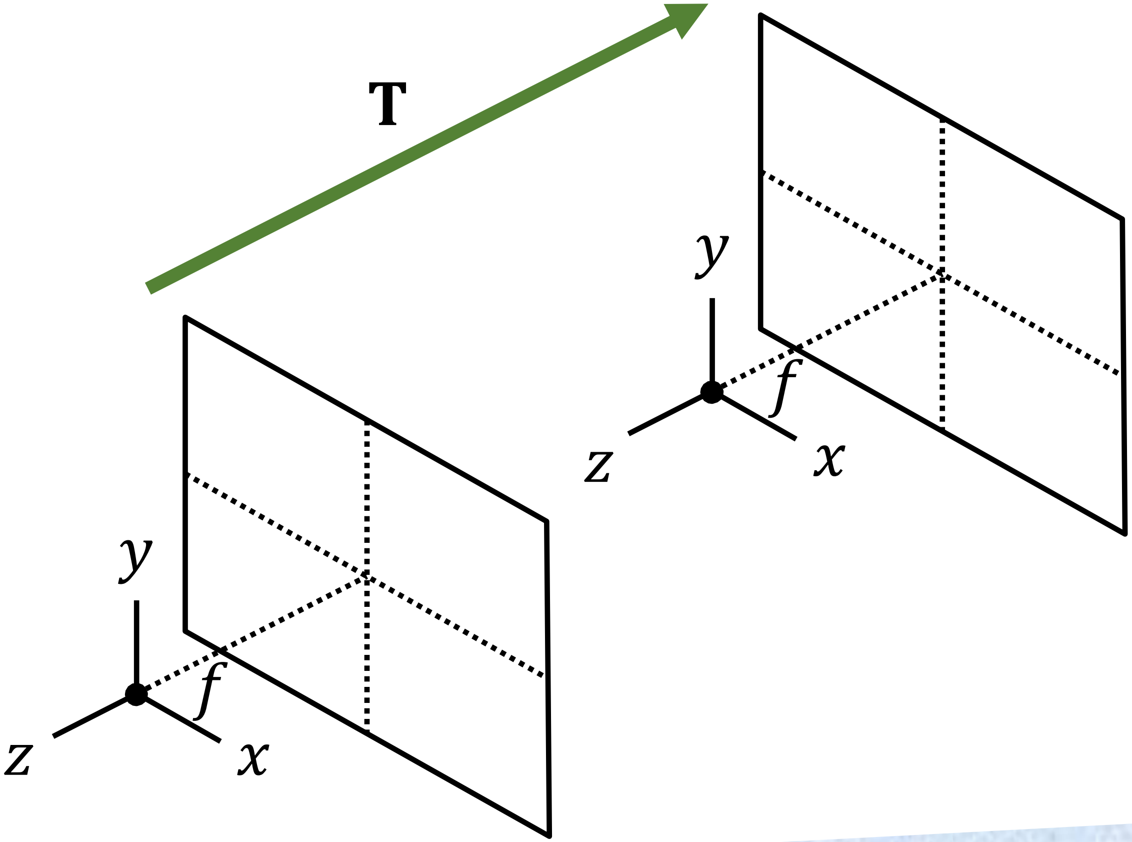

T
y
f
z
x



T
y
f
z
x
y
f
z
x

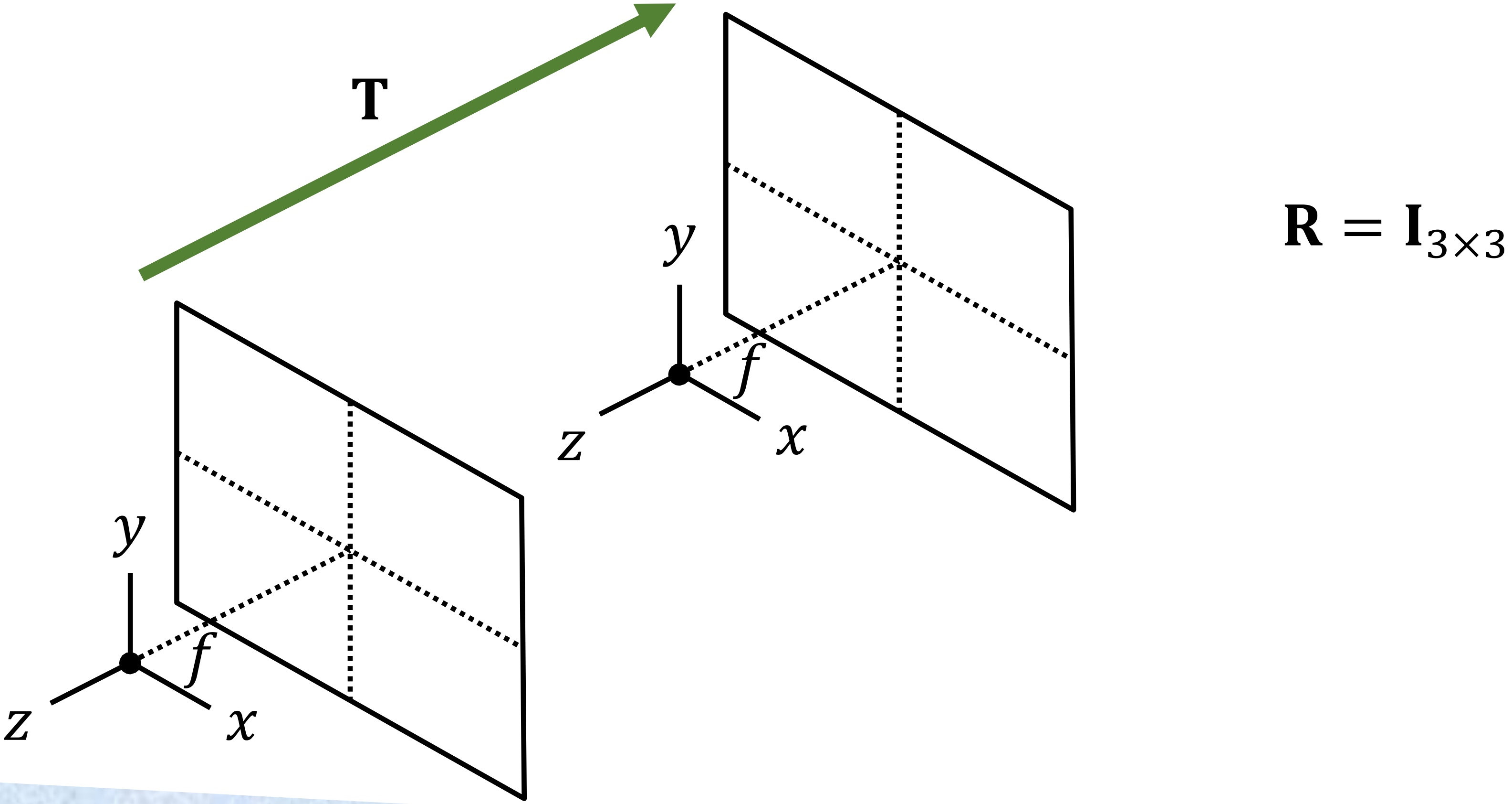


两个相机之间的相对旋转是多少?

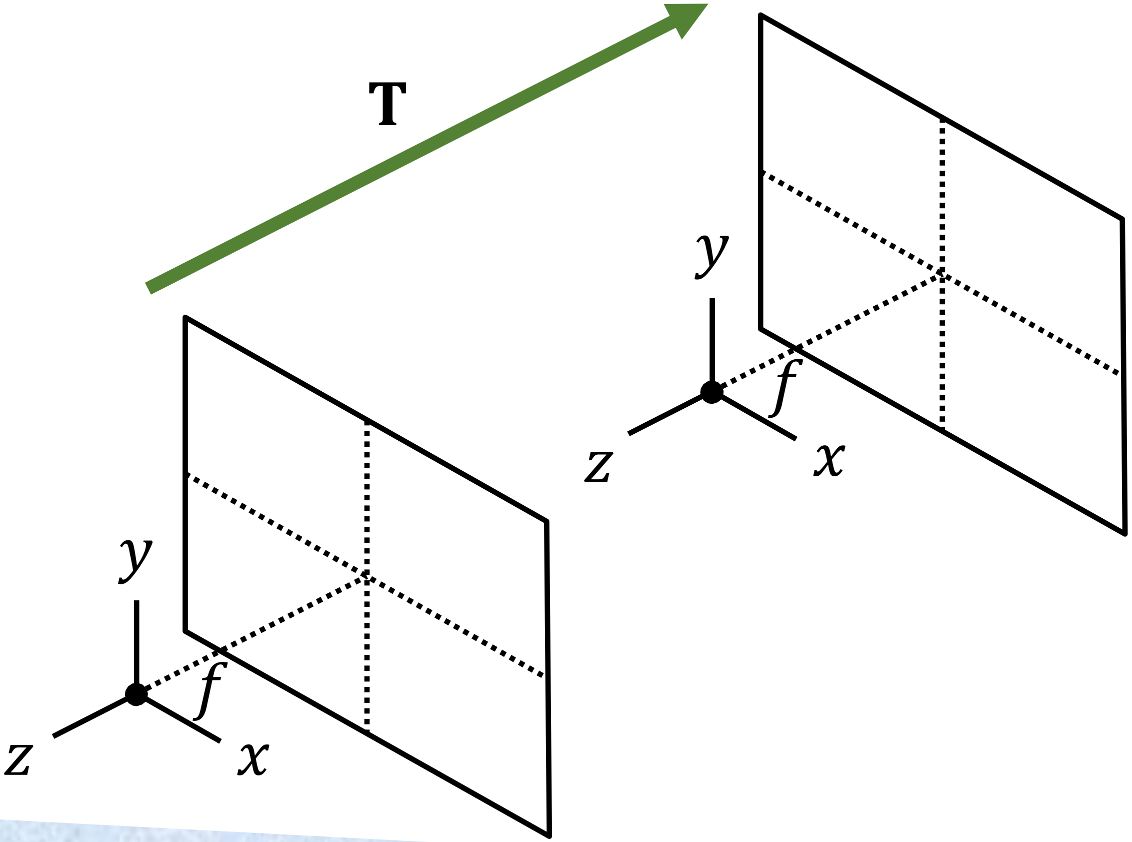


$\mathbf{R} = \mathbf{I}_{3\times3}$

两个相机之间的相对旋转是多少?



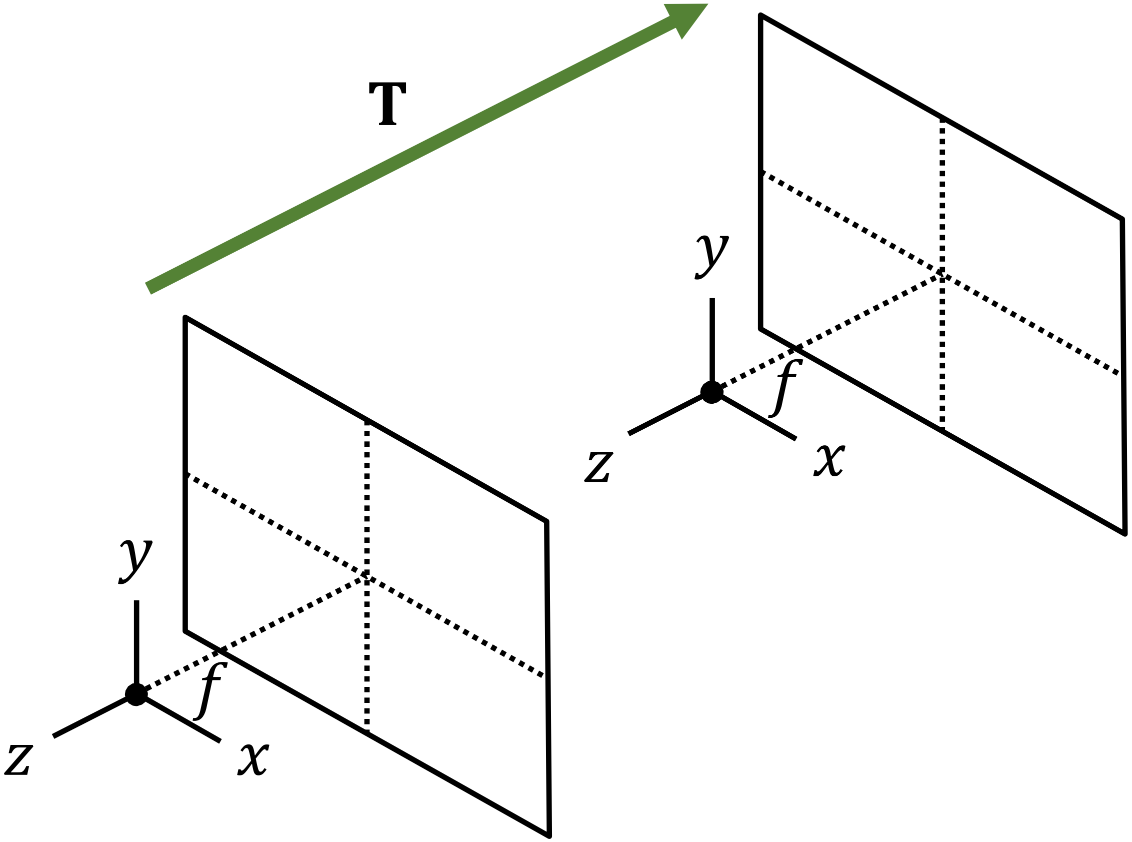
两个相机之间的相对平移是多少?



$$\mathbf{R} = \mathbf{I}_{3\times3}$$

$$\mathbf{T} = (0,0,t_z)^{\mathrm{T}}$$

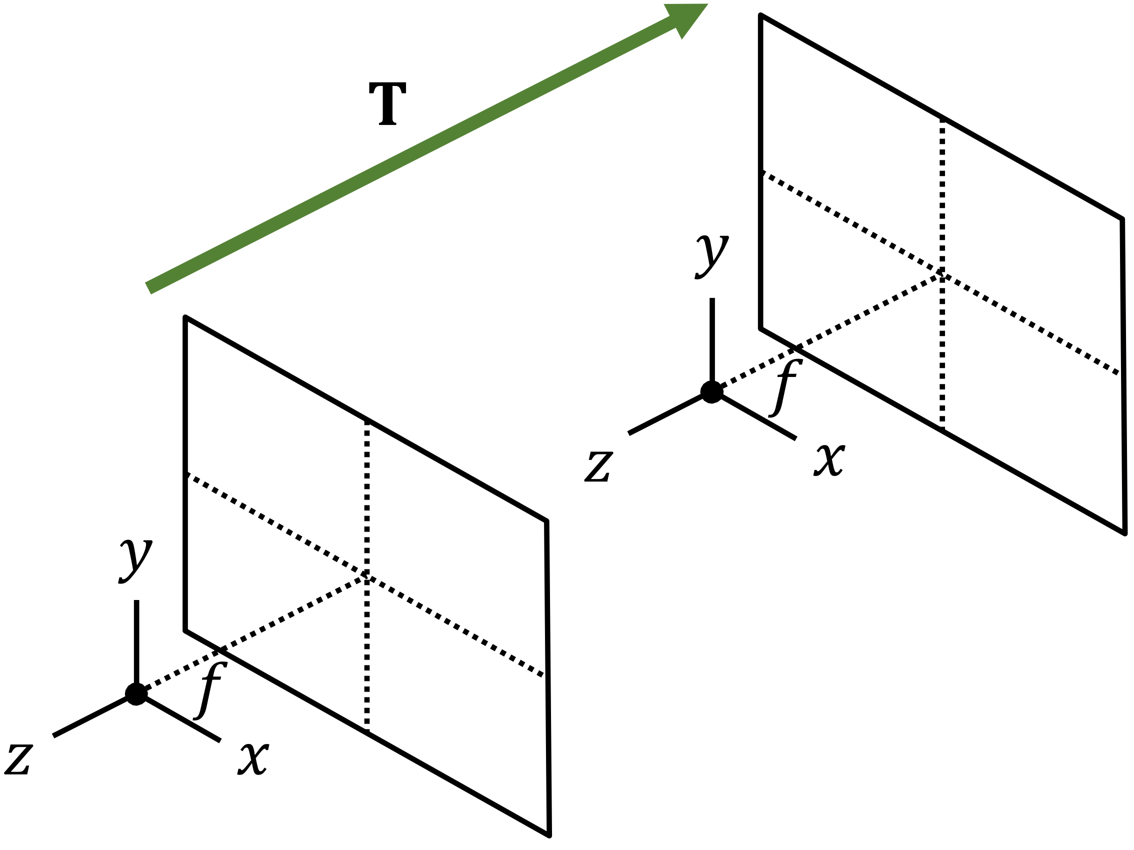
两个相机之间的相对平移是多少?



$$\mathbf{R} = \mathbf{I}_{3\times 3}$$

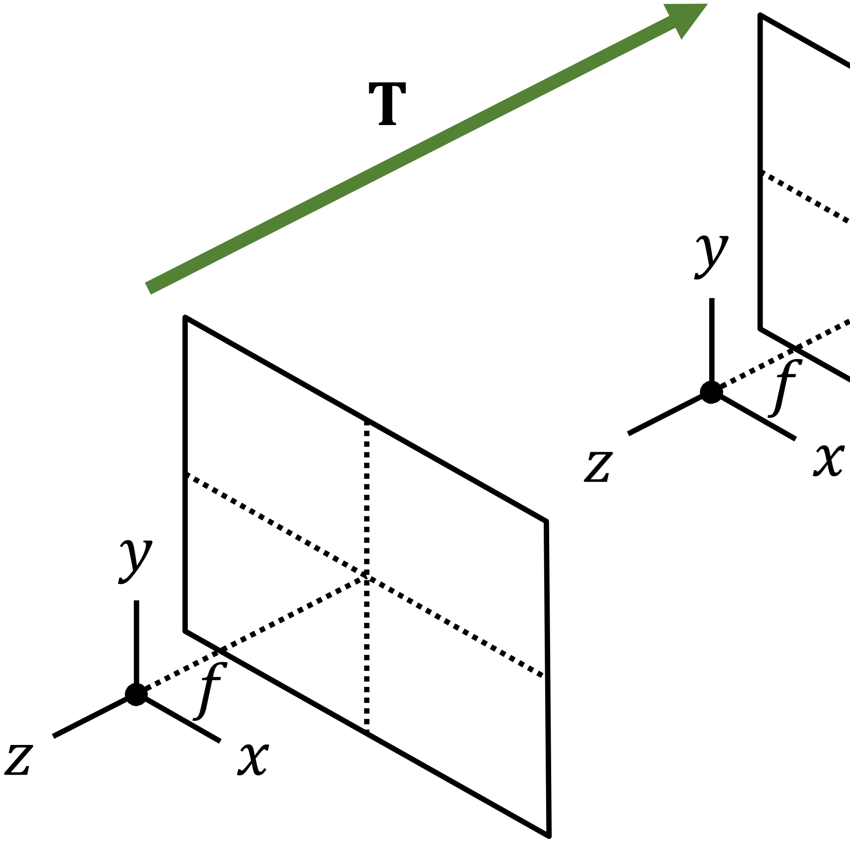
$$\mathbf{T} = (0,0,t_z)^{\mathrm{T}}$$

$$\mathbf{E} = \mathbf{R}[\mathbf{T}_\times]$$



$$\mathbf{R} = \mathbf{I}_{3\times 3}$$
$$\mathbf{T} = (0, 0, t_z)^{\mathrm{T}}$$

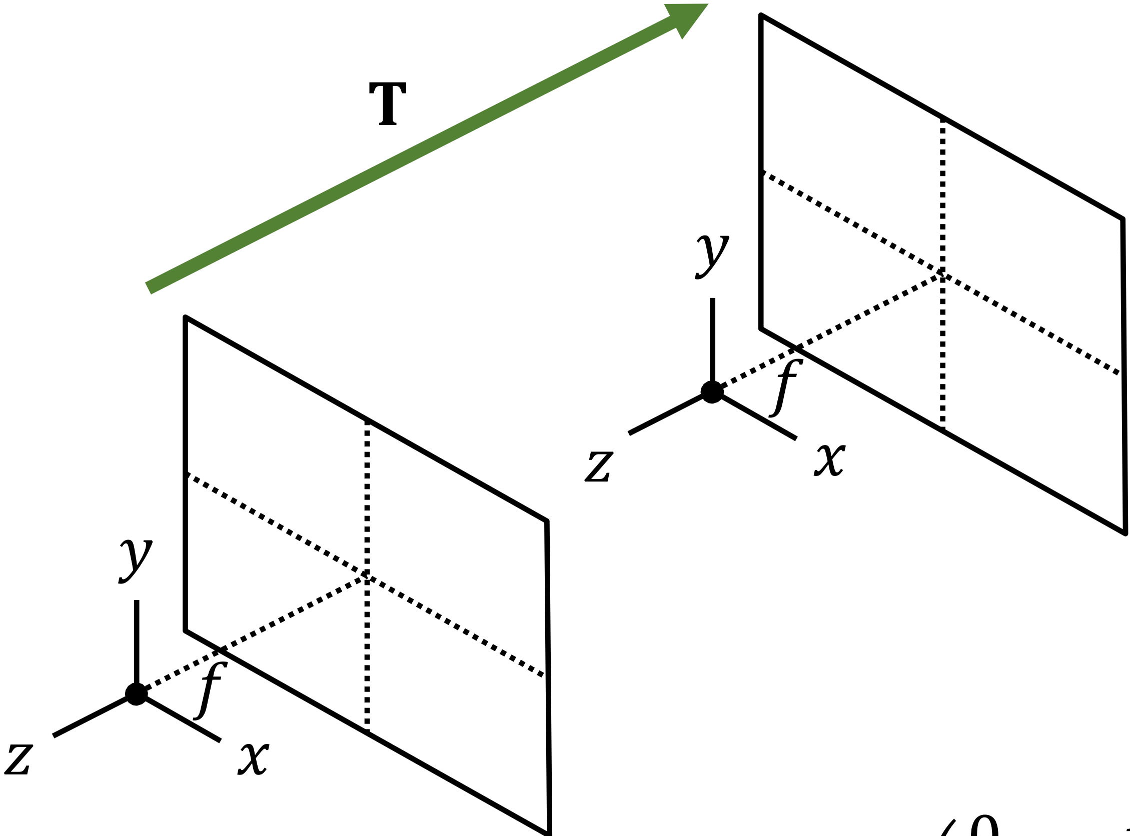
$$\mathbf{E} = \mathbf{R}[\mathbf{T}_\times] = \mathbf{I}_{3\times 3}\begin{pmatrix} 0 & -t_z & 0 \\ t_z & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$



$$\mathbf{R} = \mathbf{I}_{3\times 3}$$

$$\mathbf{T} = (0, 0, t_z)^{\mathrm{T}}$$

$$\mathbf{E} = \begin{pmatrix} 0 & -t_z & 0 \\ t_z & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

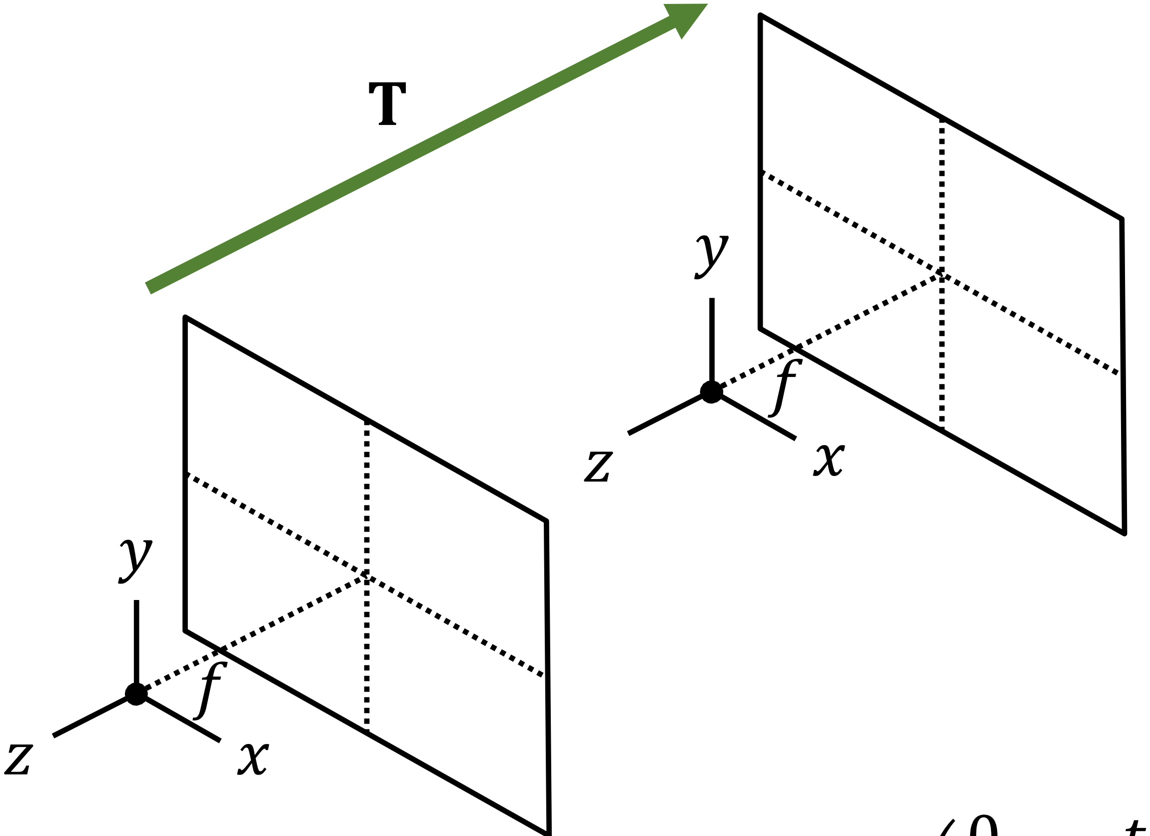


$$\mathbf{R} = \mathbf{I}_{3\times 3}$$

$$\mathbf{T} = (0, 0, t_z)^{\mathrm{T}}$$

$$\mathbf{E} = \begin{pmatrix} 0 & -t_z & 0 \\ t_z & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

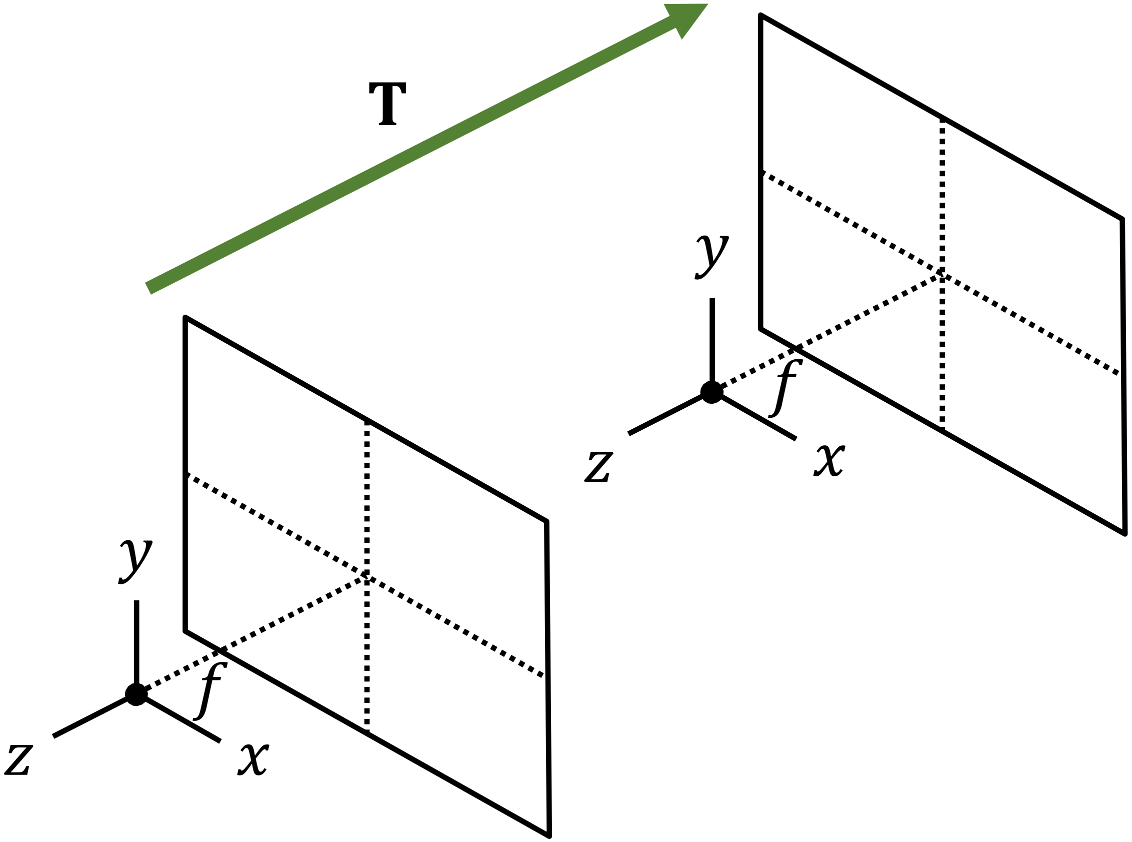
本质矩阵是确定的，但不约束尺度因子



$$\mathbf{R} = \mathbf{I}_{3\times 3}$$

$$\mathbf{T} = (0,0,t_z)^{\mathrm{T}}$$

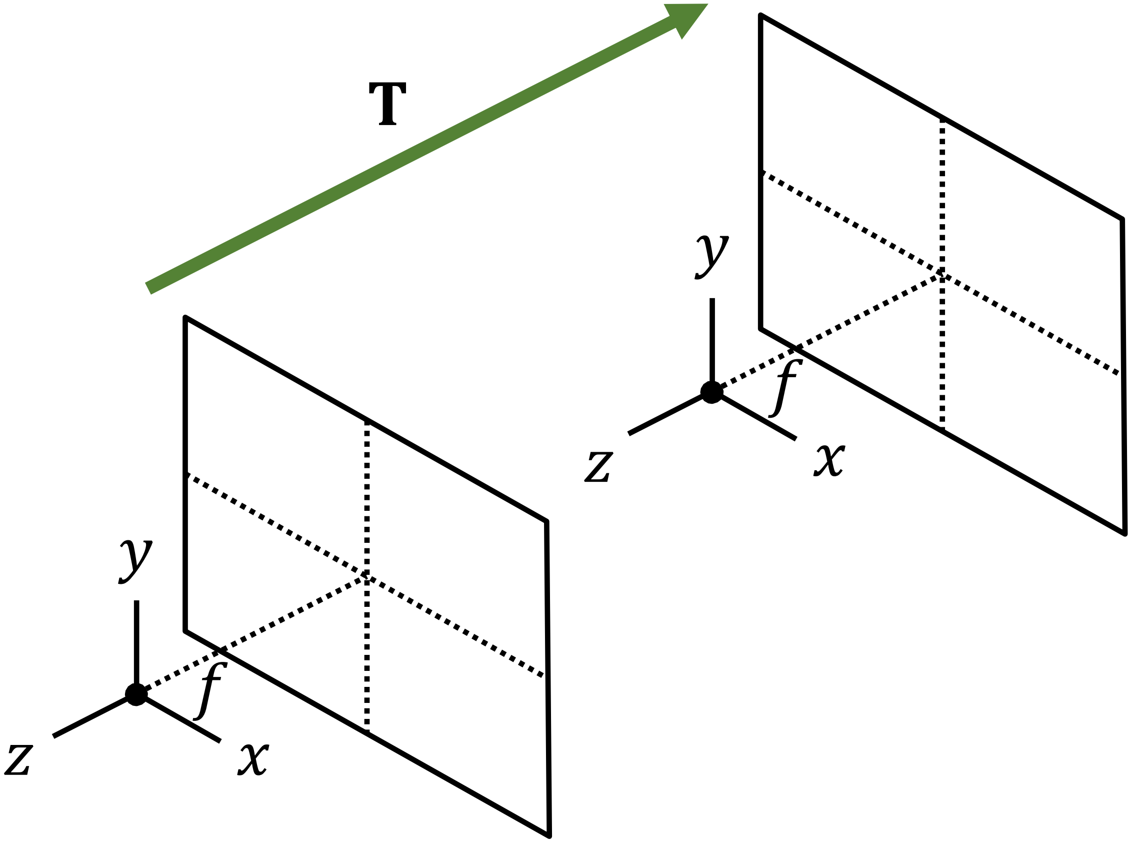
$$\mathbf{E} = \begin{pmatrix} 0 & -t_z & 0 \\ t_z & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$



$$\mathbf{R} = \mathbf{I}_{3\times3}$$

$$\mathbf{T} = (0,0,t_z)^{\mathrm{T}}$$

$$\mathbf{E} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

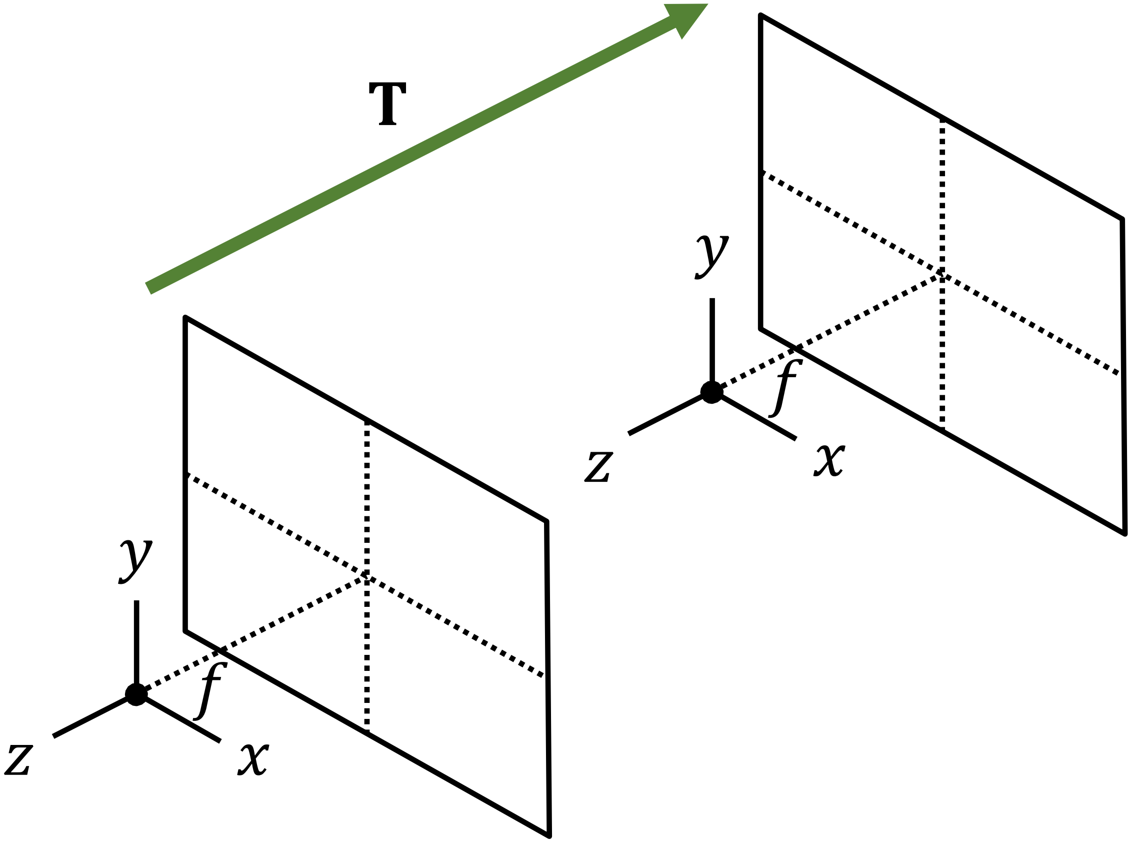


$$\mathbf{R} = \mathbf{I}_{3\times3}$$

$$\mathbf{T} = (0,0,t_z)^{\mathrm{T}}$$

$$\mathbf{E} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\mathbf{l}' = \mathbf{E}\mathbf{x}$$

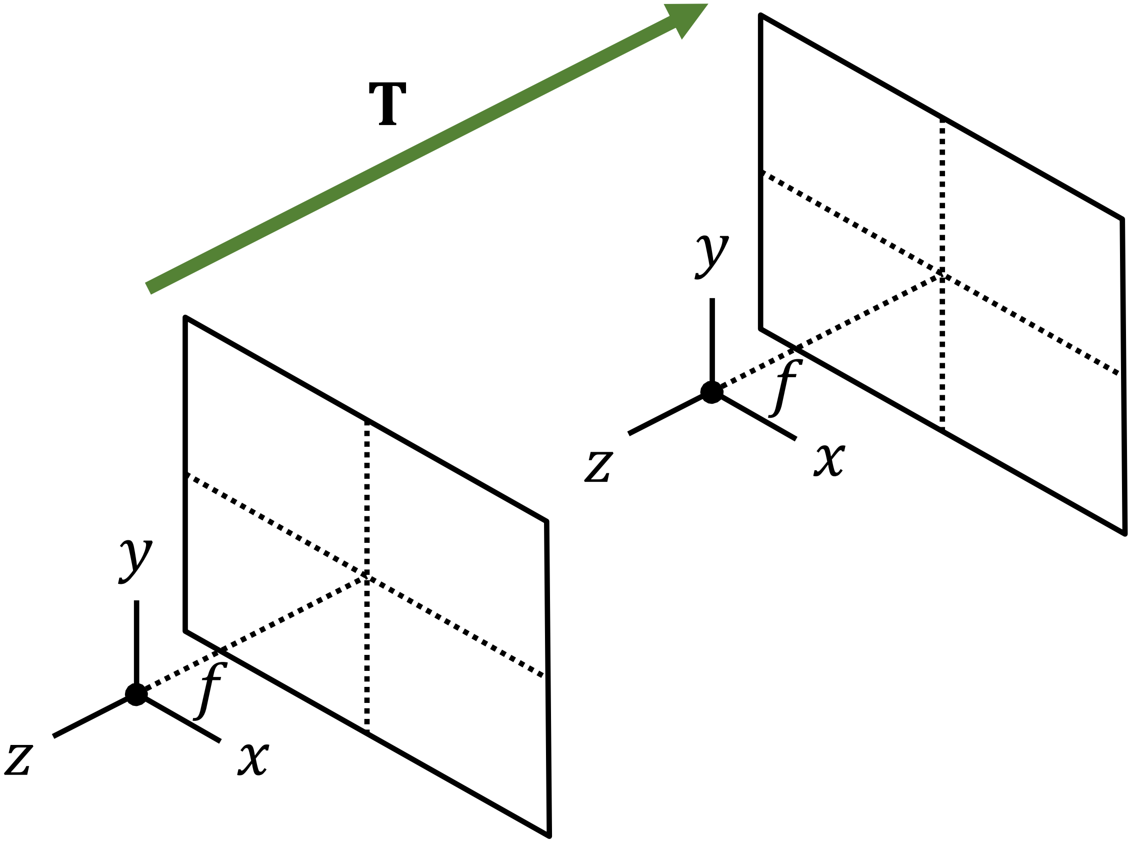


$$\mathbf{R} = \mathbf{I}_{3\times 3}$$

$$\mathbf{T} = (0, 0, t_z)^{\mathrm{T}}$$

$$\mathbf{E} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\mathbf{l}' = \mathbf{E}\mathbf{x} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

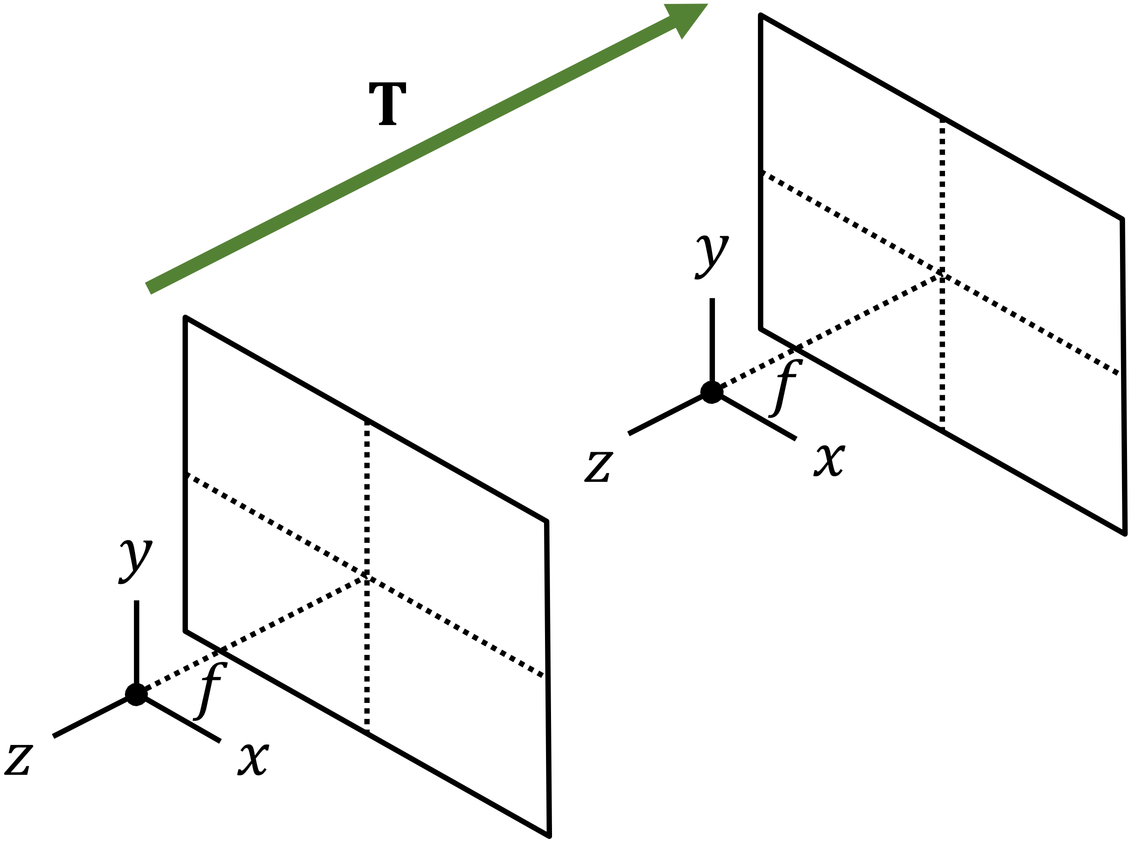


$$\mathbf{R} = \mathbf{I}_{3\times 3}$$

$$\mathbf{T} = (0, 0, t_z)^{\mathrm{T}}$$

$$\mathbf{E} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\mathbf{l}' = \mathbf{E}\mathbf{x} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} -y \\ x \\ 0 \end{pmatrix}$$



$$\mathbf{R} = \mathbf{I}_{3\times 3}$$

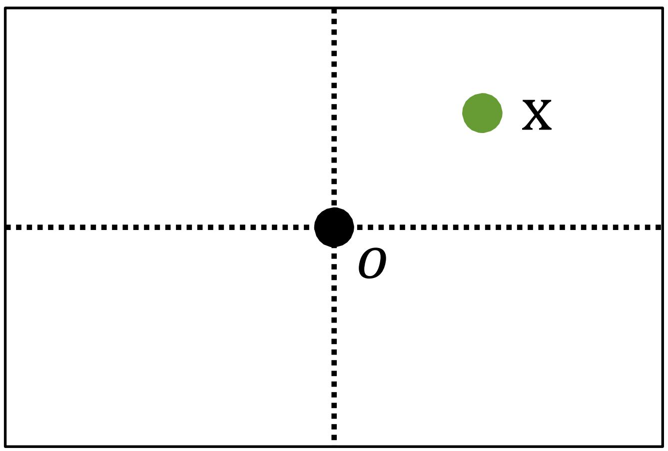
$$\mathbf{T} = (0,0,t_z)^{\mathrm{T}}$$

$$\mathbf{E} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\mathbf{l}' = \mathbf{Ex} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} -y \\ x \\ 0 \end{pmatrix}$$

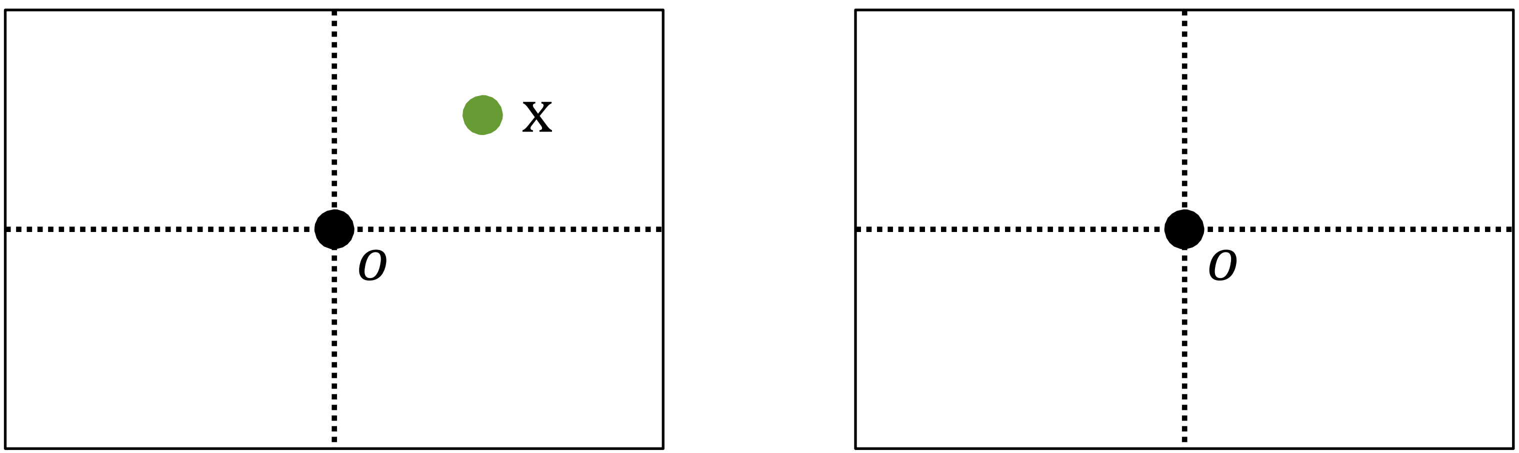
图像原点是两张图像的极点

$$\mathbf{l}' = \mathbf{E}\mathbf{x} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} -y \\ x \\ 0 \end{pmatrix}$$



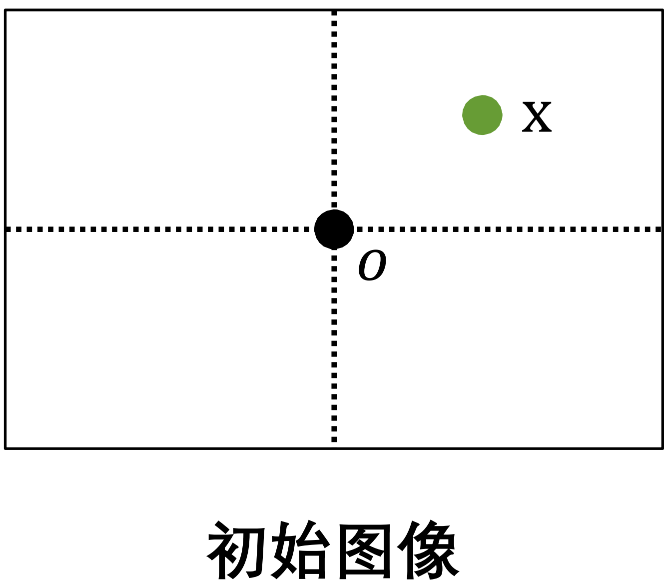
初始图像

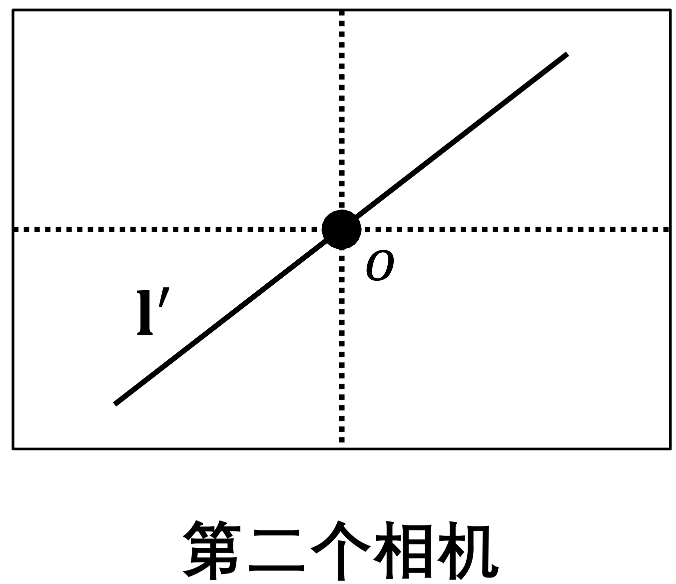
$$\mathbf{l}' = \mathbf{E}\mathbf{x} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} -y \\ x \\ 0 \end{pmatrix}$$



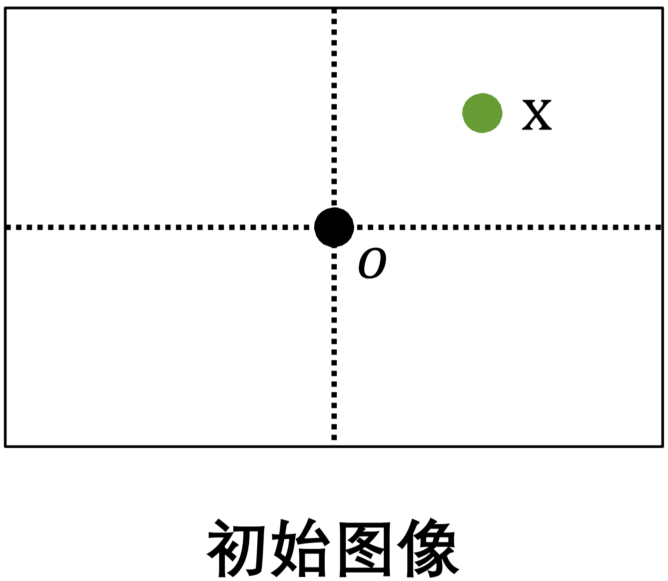
初始图像　　　　第二个相机

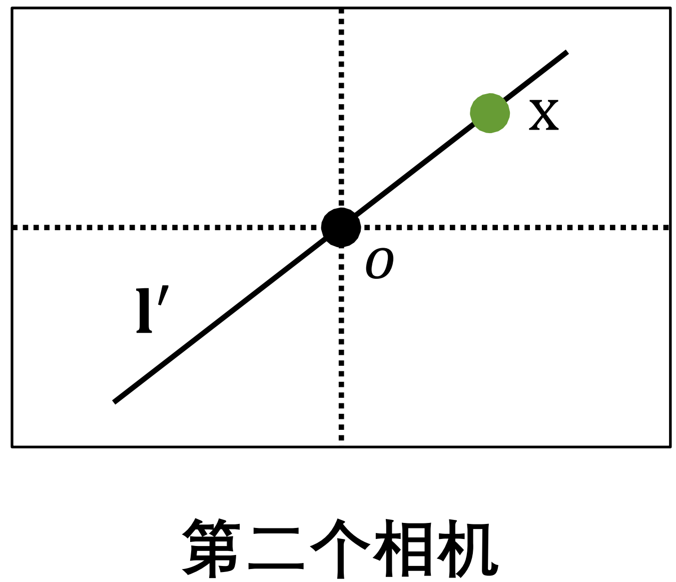
$$\mathbf{l}' = \mathbf{E}\mathbf{x} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} -y \\ x \\ 0 \end{pmatrix}$$



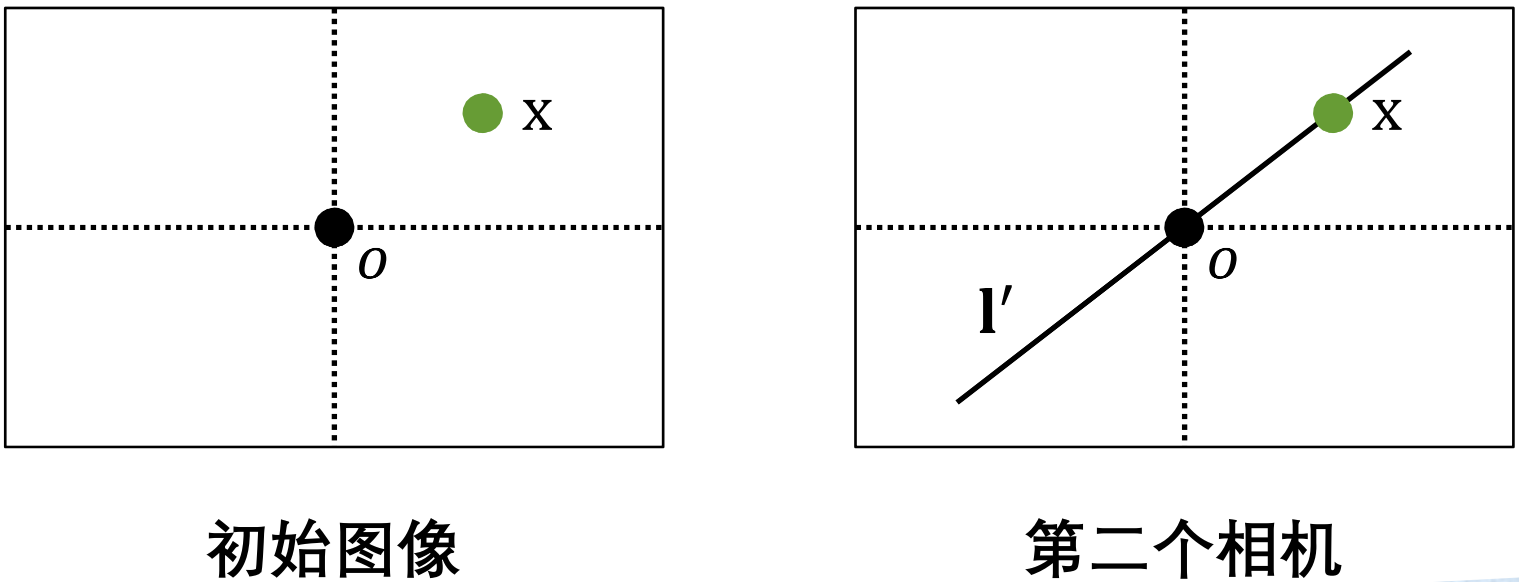


$$\mathbf{l}' = \mathbf{E}\mathbf{x} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} -y \\ x \\ 0 \end{pmatrix}$$





$$\mathbf{l}' = \mathbf{E}\mathbf{x} = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} -y \\ x \\ 0 \end{pmatrix}$$



点沿着从极点辐射的线移动


超空间跳跃

H vs. F

单应矩阵 基础矩阵

$$\mathbf{x}' = \mathbf{H}\mathbf{x}$$

单应矩阵将一个点映射到一个点

$$\mathbf{l'} = \mathbf{F}\mathbf{x}$$

基础矩阵将一个点映射到一条直线

估计基础矩阵

A computer algorithm for reconstructing a scene from two projections

H. C. Longuet-Higgins

Laboratory of Experimental Psychology, University of Sussex, Brighton BN1 9QG, UK

A simple algorithm for computing the three-dimensional structure of a scene from a correlated pair of perspective projections is described here, when the spatial relationship between the two projections is unknown. This problem is relevant not only to photographic surveying[1] but also to binocular vision[2], where the non-visual information available to the observer about the orientation and focal length of each eye is much less accurate than the optical information supplied by the retinal images themselves. The prob nocular perception of

Nature, 1981

A computer algorithm for reconstructing a scene from two projections

H. C. Longuet-Higgins

Laboratory of Experimental Psychology, University of Sussex,
Brighton BN1 9QG, UK

8点算法

A simple algorithm for computing the three-dimensional structure of a scene from a correlated pair of perspective projections is described here, when the spatial relationship between the two projections is unknown. This problem is relevant not only to photographic surveying[1] but also to binocular vision[2], where the non-visual information available to the observer about the orientation and focal length of each eye is much less accurate than the optical information supplied by the retinal images

Nature, 1981

8点算法

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

8点算法

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$(x_r, y_r, 1)\begin{pmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{pmatrix}\begin{pmatrix} x_l \\ y_l \\ 1 \end{pmatrix} = 0$$

8点算法

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$(x_r x_l, x_r y_l, x_r, y_r x_l, y_r y_l, y_r, x_l, y_l, 1)\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = 0$$

8点算法

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$(x_r x_l, x_r y_l, x_r, y_r x_l, y_r y_l, y_r, x_l, y_l, 1)\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = 0$$

两个视点之间的一对对应点

8点算法

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix} \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = \mathbf{0}$$

n对对应点

8点算法

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix} \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = \mathbf{0}$$

8点算法

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix} \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = \mathbf{0}$$

矩阵秩为8

8点算法

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix} \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = \mathbf{0}$$

矩阵秩为8

8对对应点可得一组非零解

8点算法

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix} \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = \mathbf{0}$$

矩阵秩为8

8对对应点可得一组非零解

解是零空间

8点算法

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix} \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = \mathbf{0}$$

矩阵秩为8

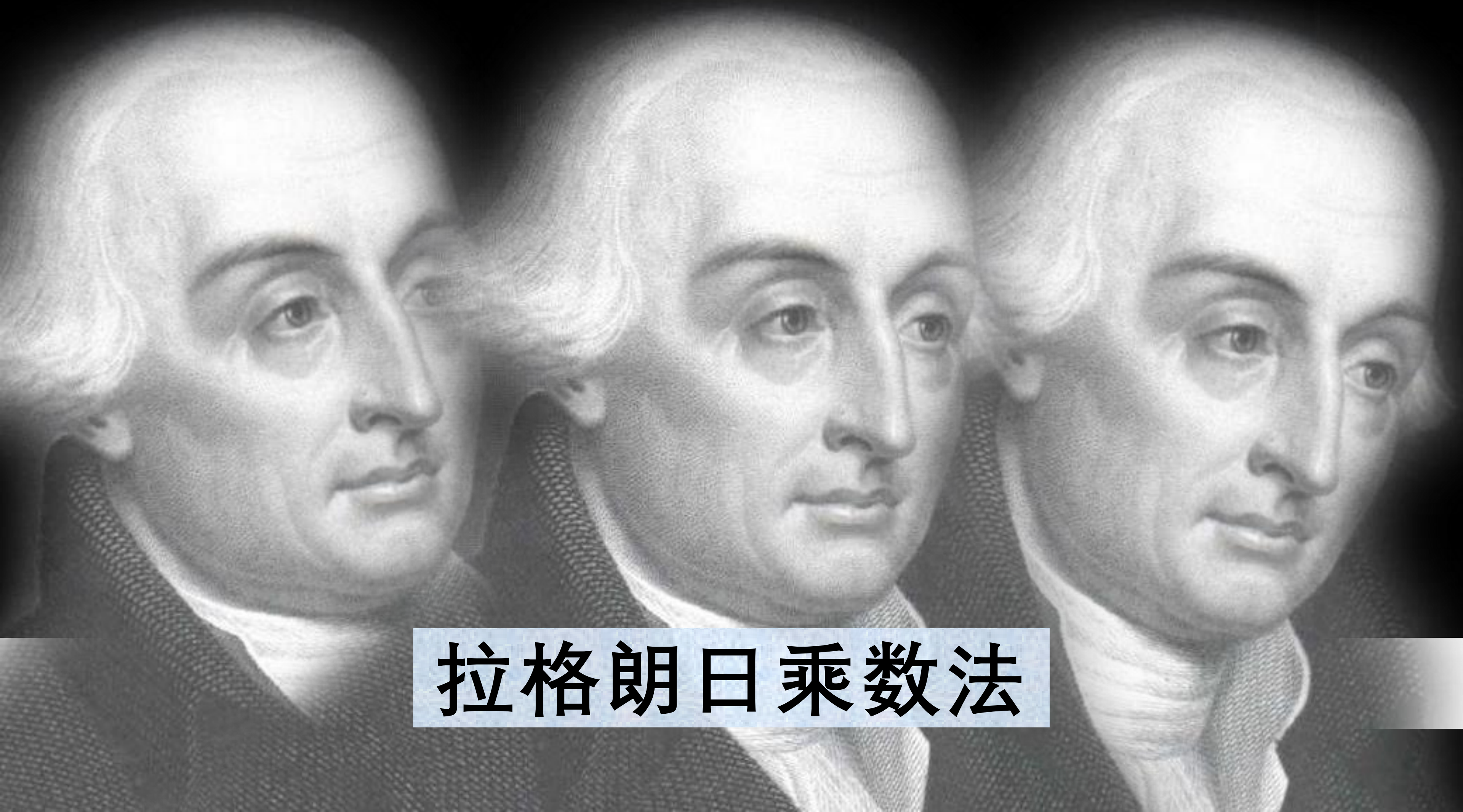
使用八个以上点并用齐次最小二乘法求解

2种
常见解法

1

约瑟夫·路易斯·拉格朗日

拉格朗日乘数法



2

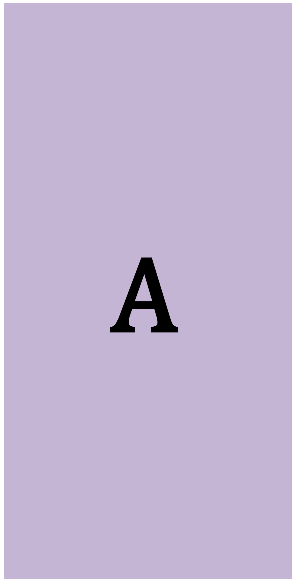
SVD

奇异值分解

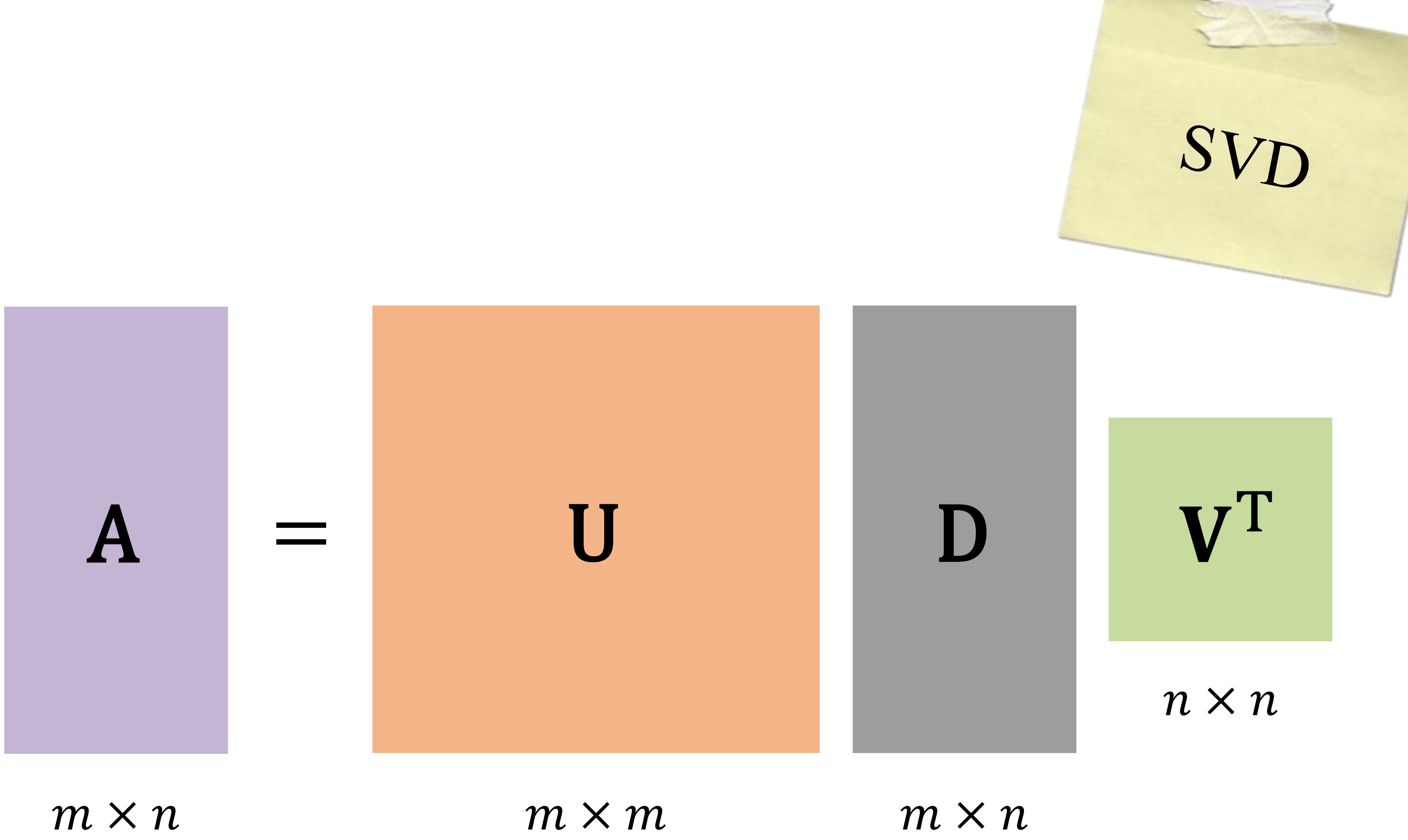
（Singular Value Decomposition）

回顾： SVD

SVD



$m \times n$


SVD
A
=
U
D
V^T
m × n
m × m
m × n
n × n

定义： 对于任意给定的矩阵$\mathbf{A} \in \mathbb{R}^{m \times n}$，它的**奇异值分解（Singular Value Decomposition, SVD）**定义为

$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^{\mathrm{T}}$$

定义： 对于任意给定的矩阵$\mathbf{A} \in \mathbb{R}^{m \times n}$，它的**奇异值分解（Singular Value Decomposition, SVD）**定义为

$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^{\mathrm{T}}$$

使得

定义： 对于任意给定的矩阵$\mathbf{A} \in \mathbb{R}^{m \times n}$，它的**奇异值分解（Singular Value Decomposition, SVD）**定义为

$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^{\mathrm{T}}$$

使得

$\mathbf{U}$是具有正交列向量的$m \times m$的正交矩阵

定义： 对于任意给定的矩阵$\mathbf{A} \in \mathbb{R}^{m \times n}$，它的**奇异值分解**（**Singular Value Decomposition, SVD**）定义为

$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^{\mathrm{T}}$$

使得

$\mathbf{U}$是具有正交列向量的 $m \times m$的正交矩阵

$\mathbf{V}^{\mathrm{T}}$是$n \times n$的正交矩阵

定义： 对于任意给定的矩阵$\mathbf{A} \in \mathbb{R}^{m\times n}$，它的**奇异值分解（Singular Value Decomposition, SVD）**定义为

$$\mathbf{A} = \mathbf{UDV}^{\mathrm{T}}$$

使得

$\mathbf{U}$是具有正交列向量的 $m \times m$的正交矩阵

$\mathbf{V}^{\mathrm{T}}$是$n \times n$的正交矩阵

$\mathbf{D}$是$m \times n$的具有非负元素的对角矩阵，称为$\mathbf{A}$的**奇异值**

定义： 对于任意给定的矩阵$\mathbf{A} \in \mathbb{R}^{m \times n}$，它的**奇异值分解（Singular Value Decomposition, SVD）**定义为

$$\mathbf{A} = \mathbf{UDV}^{\mathrm{T}}$$

使得

$\mathbf{U}$是具有正交列向量的$m \times m$的正交矩阵

$\mathbf{V}^{\mathrm{T}}$是$n \times n$的正交矩阵

$\mathbf{D}$是$m \times n$的具有非负元素的对角矩阵，称为$\mathbf{A}$的奇异值

假设：$\mathbf{D}$的主对角线元素值按降序排序，

$$\sigma_1 \geq \sigma_2 \geq \cdots \sigma_n \geq 0$$

SVD求解优化问题

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

SVD求解优化问题

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

SVD求解优化问题

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

齐次最小二乘问题

SVD求解优化问题

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

为什么需要单位向量这个约束条件?

SVD求解
优化问题

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

为什么需要单位向量这个约束条件?

避免平凡解

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

$$\|\mathbf{Ax}\|^2$$

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

$$\|\mathbf{Ax}\|^2 = \left\|\mathbf{UDV}^{\mathrm{T}}\mathbf{x}\right\|^2$$

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

$$\|\mathbf{Ax}\|^2 = \left\|\mathbf{UDV}^{\mathrm{T}}\mathbf{x}\right\|^2$$

$$= \left(\mathbf{UDV}^{\mathrm{T}}\mathbf{x}\right)^{\mathrm{T}}\left(\mathbf{UDV}^{\mathrm{T}}\mathbf{x}\right)$$

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

$$\begin{aligned}\|\mathbf{Ax}\|^2 &= \left\|\mathbf{UDV}^\mathrm{T}\mathbf{x}\right\|^2 \\ &= \left(\mathbf{UDV}^\mathrm{T}\mathbf{x}\right)^\mathrm{T}\left(\mathbf{UDV}^\mathrm{T}\mathbf{x}\right) \\ &= \left(\mathbf{x}^\mathrm{T}\mathbf{VD}^\mathrm{T}\mathbf{U}^\mathrm{T}\right)\left(\mathbf{UDV}^\mathrm{T}\mathbf{x}\right)\end{aligned}$$

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

$$\|\mathbf{Ax}\|^2 = \left\|\mathbf{UDV}^{\mathrm{T}}\mathbf{x}\right\|^2$$

$$= \left(\mathbf{UDV}^{\mathrm{T}}\mathbf{x}\right)^{\mathrm{T}}\left(\mathbf{UDV}^{\mathrm{T}}\mathbf{x}\right)$$

$$= \left(\mathbf{x}^{\mathrm{T}}\mathbf{VD}^{\mathrm{T}}\mathbf{U}^{\mathrm{T}}\right)\left(\mathbf{UDV}^{\mathrm{T}}\mathbf{x}\right)$$

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

$$\|\mathbf{Ax}\|^2 = \left\|\mathbf{UDV}^\mathrm{T}\mathbf{x}\right\|^2$$
$$= \left(\mathbf{UDV}^\mathrm{T}\mathbf{x}\right)^\mathrm{T}\left(\mathbf{UDV}^\mathrm{T}\mathbf{x}\right)$$
$$= \left(\mathbf{x}^\mathrm{T}\mathbf{VD}^\mathrm{T}\mathbf{U}^\mathrm{T}\right)\left(\mathbf{UDV}^\mathrm{T}\mathbf{x}\right)$$

单位矩阵

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

$$\begin{aligned}
\|\mathbf{Ax}\|^2 &= \left\|\mathbf{UDV}^{\mathrm{T}}\mathbf{x}\right\|^2 \\
&= \left(\mathbf{UDV}^{\mathrm{T}}\mathbf{x}\right)^{\mathrm{T}}\left(\mathbf{UDV}^{\mathrm{T}}\mathbf{x}\right) \\
&= \left(\mathbf{x}^{\mathrm{T}}\mathbf{VD}^{\mathrm{T}}\mathbf{U}^{\mathrm{T}}\right)\left(\mathbf{UDV}^{\mathrm{T}}\mathbf{x}\right) \\
&= \mathbf{x}^{\mathrm{T}}\mathbf{VD}^{\mathrm{T}}\mathbf{DV}^{\mathrm{T}}\mathbf{x}
\end{aligned}$$

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

$$\|\mathbf{Ax}\|^2 = \mathbf{x}^{\mathrm{T}}\mathbf{V}\mathbf{D}^{\mathrm{T}}\mathbf{D}\mathbf{V}^{\mathrm{T}}\mathbf{x}$$

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

$$\|\mathbf{Ax}\|^2 = \mathbf{x}^T\mathbf{V}\mathbf{D}^T\mathbf{D}\mathbf{V}^T\mathbf{x}$$

令 $\mathbf{y} = \mathbf{V}^T\mathbf{x}$

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

$$\|\mathbf{Ax}\|^2 = \mathbf{x}^{\mathrm{T}}\mathbf{V}\mathbf{D}^{\mathrm{T}}\mathbf{D}\mathbf{V}^{\mathrm{T}}\mathbf{x}$$

令 $\mathbf{y} = \mathbf{V}^{\mathrm{T}}\mathbf{x}$

现在我们最小化

$$\mathbf{y}^{\mathrm{T}}\mathbf{D}^{\mathrm{T}}\mathbf{D}\mathbf{y}$$

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

$$\|\mathbf{Ax}\|^2 = \mathbf{x}^T\mathbf{V}\mathbf{D}^T\mathbf{D}\mathbf{V}^T\mathbf{x}$$

令 $\mathbf{y} = \mathbf{V}^T\mathbf{x}$

现在我们最小化

$$\mathbf{y}^T\mathbf{D}^T\mathbf{D}\mathbf{y}$$

并服从如下约束

$$\|\mathbf{y}\| = 1$$

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

$$\|\mathbf{Ax}\|^2 = \mathbf{x}^T\mathbf{V}\mathbf{D}^T\mathbf{D}\mathbf{V}^T\mathbf{x}$$

令 $\mathbf{y} = \mathbf{V}^T\mathbf{x}$

现在我们最小化

$$\mathbf{y}^T\mathbf{D}^T\mathbf{D}\mathbf{y}$$

并服从如下约束

$$\|\mathbf{y}\| = 1$$

$$\|\mathbf{y}\| = \|\mathbf{V}^T\mathbf{x}\| = \|\mathbf{x}\| = 1$$

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

现在我们最小化

$$\mathbf{y}^{\mathrm{T}}\mathbf{D}^{\mathrm{T}}\mathbf{D}\mathbf{y}$$

并服从如下约束

$$\|\mathbf{y}\| = 1$$

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

现在我们最小化

$$\mathbf{y}^{\mathrm{T}}\mathbf{D}^{\mathrm{T}}\mathbf{D}\mathbf{y}$$

并服从如下约束

$$\|\mathbf{y}\| = 1$$

$$\mathbf{D} = \begin{pmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n \end{pmatrix}$$

令

$$\mathbf{y} = (y_1, y_2, \dots, y_n)^{\mathrm{T}}$$

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

现在我们最小化

$$\mathbf{y}^{\mathrm{T}}\mathbf{D}^{\mathrm{T}}\mathbf{Dy}$$

并服从如下约束

$$\|\mathbf{y}\| = 1$$

$$\sigma_1^2 y_1^2 + \sigma_2^2 y_2^2 + \cdots + \sigma_n^2 y_n^2$$

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

现在我们最小化

$$\mathbf{y}^{\mathrm{T}}\mathbf{D}^{\mathrm{T}}\mathbf{D}\mathbf{y}$$

并服从如下约束

$$\|\mathbf{y}\| = 1$$

$$\sigma_1^2 y_1^2 + \sigma_2^2 y_2^2 + \cdots + \sigma_n^2 y_n^2$$

当y取什么值时获得最小值?

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

$$\|\mathbf{x}\| = 1$$

现在我们最小化

$$\mathbf{y}^{\mathrm{T}}\mathbf{D}^{\mathrm{T}}\mathbf{D}\mathbf{y}$$

并服从如下约束

$$\|\mathbf{y}\| = 1$$

$$\sigma_1^2 y_1^2 + \sigma_2^2 y_2^2 + \cdots + \sigma_n^2 y_n^2$$

当y取什么值时获得最小值?

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

找到一个向量x来最小化

$$\|\mathbf{Ax}\|^2$$

并服从如下约束

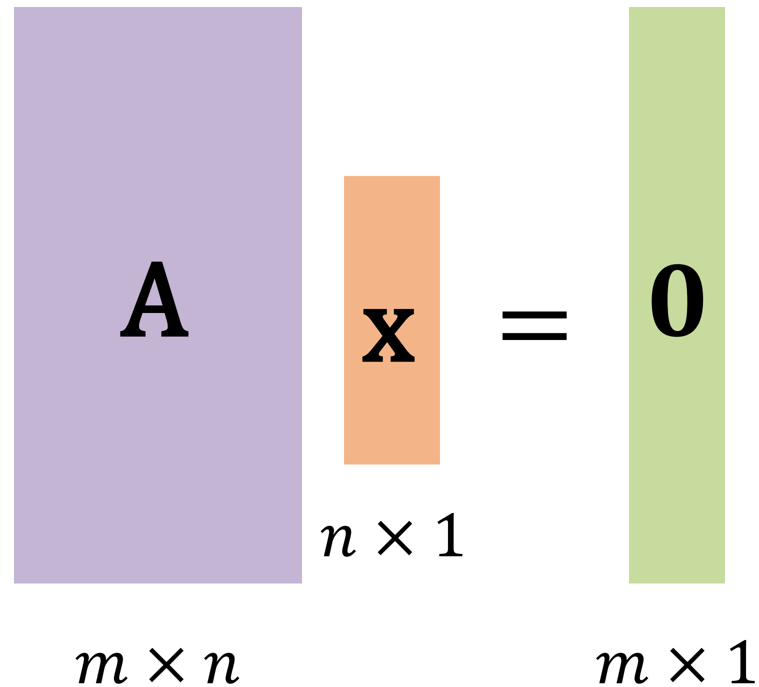
$$\|\mathbf{x}\| = 1$$

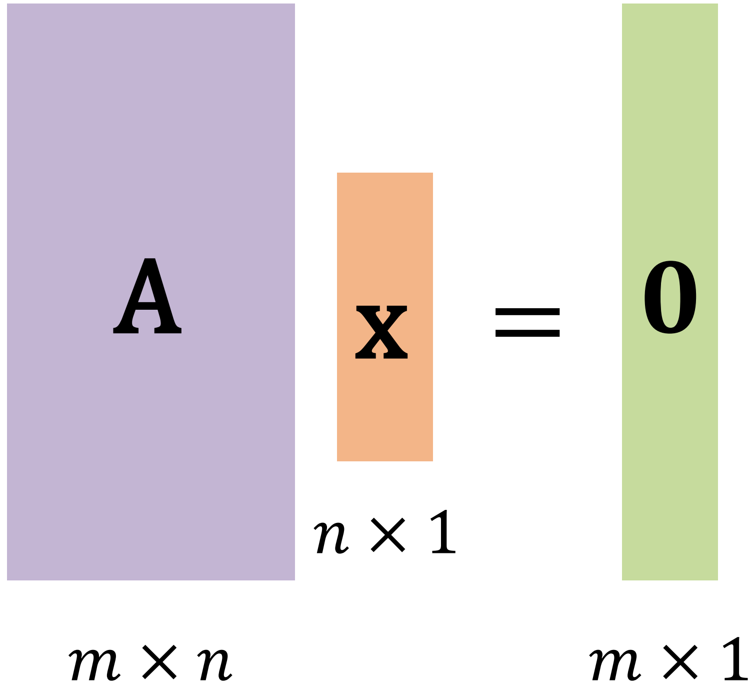
当y取什么值时获得最小值?

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

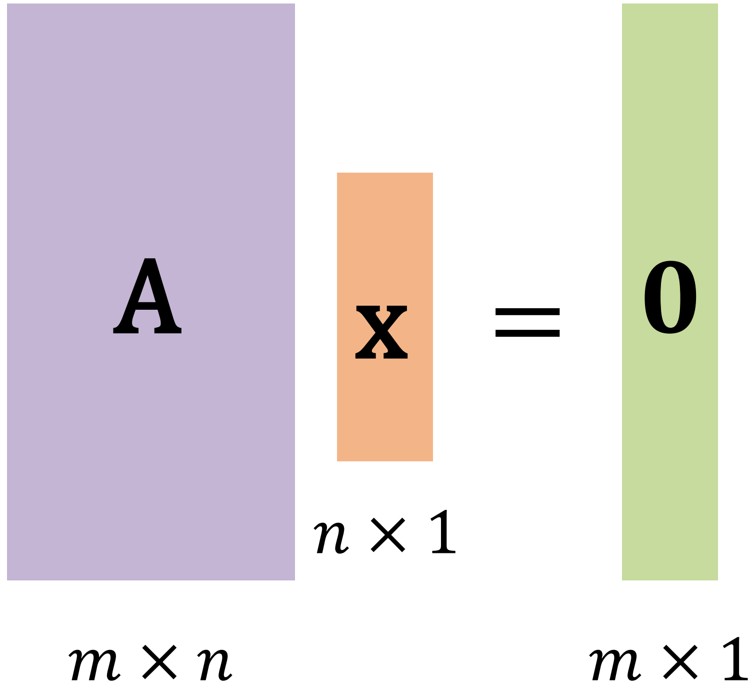
由于$y = \mathbf{V}^{\mathrm{T}}\mathbf{x}$, $\mathbf{x}$是$\mathbf{V}$的最后一列

太长不看？


A
x
=
0
n × 1
m × n
m × 1



$$\mathbf{x}^* = \arg\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x}\|^2 \text{ subject to } \|\mathbf{x}\| = 1$$



$$\mathbf{x}^* = \arg\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x}\|^2 \quad \text{subject to } \|\mathbf{x}\| = 1$$

计算A的SVD，解就是V的最后一列

回顾：SVD

SVD求解基础矩阵

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix} \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = \mathbf{0}$$

SVD求解基础矩阵

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$\underbrace{\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix}}_{\mathbf{A}} \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = \mathbf{0}$$

SVD求解基础矩阵

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

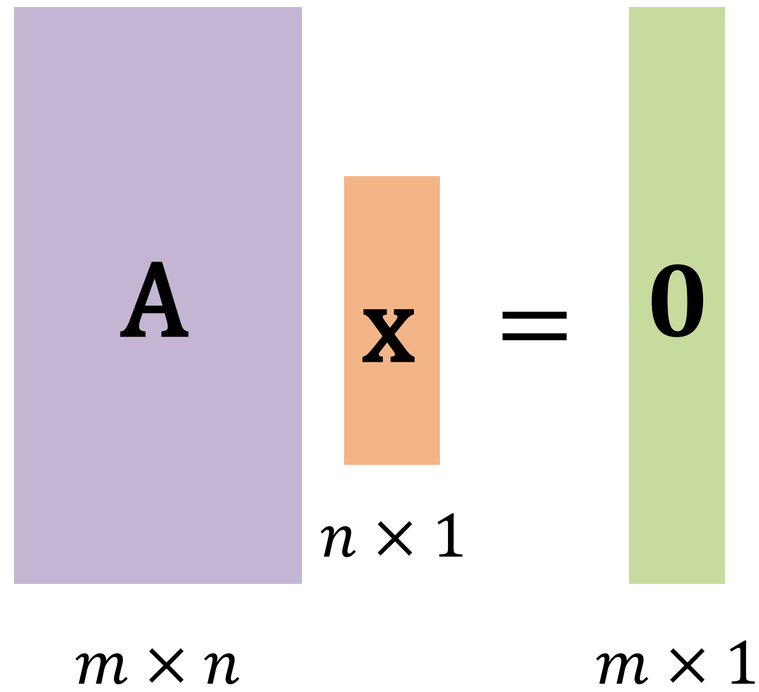
$$\underbrace{\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix}}_{\mathbf{f}} = \mathbf{0}$$

SVD求解基础矩阵

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$\underbrace{\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix}}_{\mathbf{f}} = \mathbf{0}$$

如何求解基础矩阵f?



$$\mathbf{x}^* = \arg\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x}\|^2 \quad \text{subject to } \|\mathbf{x}\| = 1$$

计算A的SVD，解就是V的最后一列

SVD求解基础矩阵

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$\underbrace{\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix}}_{\mathbf{f}} = \mathbf{0}$$

如何求解基础矩阵f?

$$\underbrace{\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix}}_{\mathbf{f}} = \mathbf{0}$$

如何求解基础矩阵f?

$$\underbrace{\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix}}_{\mathbf{f}} = \mathbf{0}$$

如何求解基础矩阵f?

$$\arg\min_{\mathbf{f}} \|\mathbf{Af}\|^2$$

并服从如下约束

$$\|\mathbf{f}\| = 1$$

$$\underbrace{\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix}}_{\mathbf{f}} = \mathbf{0}$$

计算A的SVD，解就是V的最后一列

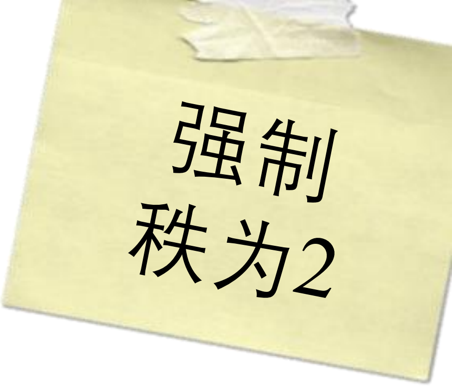
$$\underbrace{\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix}}_{\mathbf{f}} = \mathbf{0}$$

基础矩阵的秩必须为2

$$
\underbrace{\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix}}_{\mathbf{f}} = \mathbf{0}
$$

基础矩阵的秩必须为2

所估计的F可能为**满秩**



$$\tilde{\mathbf{F}}^* = \arg\min_{\tilde{\mathbf{F}}} \left\| \tilde{\mathbf{F}} - \hat{\mathbf{F}} \right\|^2 \text{ subject to } \det\left(\tilde{\mathbf{F}}\right) = 0$$

3

主要步骤

步骤1

计算SVD

$$\mathbf{F} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathrm{T}}$$

步骤2

调整奇异值

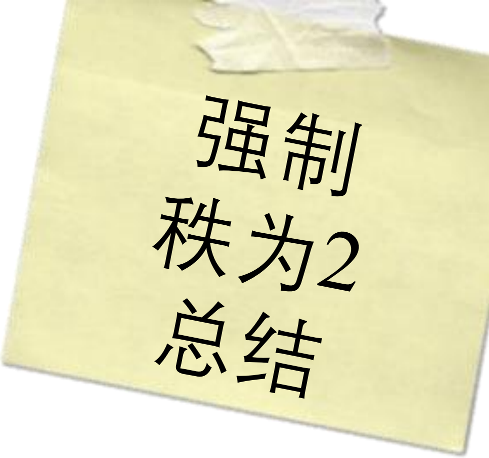
$$\mathbf{\Sigma} = \begin{pmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \end{pmatrix}$$

$$\mathbf{\Sigma}' = \begin{pmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

步骤3

重新组合基础矩阵

$$\mathbf{F}' = \mathbf{U}\boldsymbol{\Sigma}'\mathbf{V}^{\mathrm{T}}$$



1. 计算F的SVD

强制秩为2总结

1. 计算F的SVD

2. 将F最小的奇异值设为0

强制秩为2总结

1. 计算F的SVD

2. 将F最小的奇异值设为0

3. 重新计算F

Python时间

8点算法

```
# build constraint matrix
A = np.stack([x2[:, 0]*x1[:, 0], x2[:, 0]*x1[:, 1], \
    x2[:, 0], x2[:, 1]*x1[:, 0], x2[:, 1]*x1[:, 1], x2[:, 1], \
    x1[:, 0], x1[:, 1], np.ones(npts)], axis=1)

# compute SVD matrix factorization
U, D, V = np.linalg.svd(A)

# extract Fundamental Matrix from the column of V
# corresponding to the smallest singular value
F = V[:, -1]
F = F.reshape(3, 3).T

# Enforce rank 2 constraint
U, D, V = np.linalg.svd(F)
F = U @ np.diag([D[0], D[1], 0]) @ V.T
```

```python
# build constraint matrix
A = np.stack([x2[:, 0]*x1[:, 0], x2[:, 0]*x1[:, 1], \
    x2[:, 0], x2[:, 1]*x1[:, 0], x2[:, 1]*x1[:, 1], x2[:, 1], \
    x1[:, 0], x1[:, 1], np.ones(npts)], axis=1)

# compute SVD matrix factorization
U, D, V = np.linalg.svd(A)

# extract Fundamental Matrix from the column of V
# corresponding to the smallest singular value
F = V[:, -1]
F = F.reshape(3, 3).T

# Enforce rank 2 constraint
U, D, V = np.linalg.svd(F)
F = U @ np.diag([D[0], D[1], 0]) @ V.T
```

$$\mathbf{p}_r^{\mathrm{T}}\mathbf{F}\mathbf{p}_l = 0$$

$$\underbrace{\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix}}_{\mathbf{f}} = \mathbf{0}$$

```python
# build constraint matrix
A = np.stack([x2[:, 0]*x1[:, 0], x2[:, 0]*x1[:, 1], \
    x2[:, 0], x2[:, 1]*x1[:, 0], x2[:, 1]*x1[:, 1], x2[:, 1], \
    x1[:, 0], x1[:, 1], np.ones(npts)], axis=1)

# compute SVD matrix factorization
U, D, V = np.linalg.svd(A)

# extract Fundamental Matrix from the column of V
# corresponding to the smallest singular value
F = V[:, -1]
F = F.reshape(3, 3).T

# Enforce rank 2 constraint
U, D, V = np.linalg.svd(F)
F = U @ np.diag([D[0], D[1], 0]) @ V.T
```

```python
# build constraint matrix
A = np.stack([x2[:, 0]*x1[:, 0], x2[:, 0]*x1[:, 1], \
    x2[:, 0], x2[:, 1]*x1[:, 0], x2[:, 1]*x1[:, 1], x2[:, 1], \
    x1[:, 0], x1[:, 1], np.ones(npts)], axis=1)

# compute SVD matrix factorization
U, D, V = np.linalg.svd(A)

# extract Fundamental Matrix from the column of V
# corresponding to the smallest singular value
F = V[:, -1]
F = F.reshape(3, 3).T

# Enforce rank 2 constraint
U, D, V = np.linalg.svd(F)
F = U @ np.diag([D[0], D[1], 0]) @ V.T
```

```python
# build constraint matrix
A = np.stack([x2[:, 0]*x1[:, 0], x2[:, 0]*x1[:, 1], \
    x2[:, 0], x2[:, 1]*x1[:, 0], x2[:, 1]*x1[:, 1], x2[:, 1], \
    x1[:, 0], x1[:, 1], np.ones(npts)], axis=1)

# compute SVD matrix factorization
U, D, V = np.linalg.svd(A)

# extract Fundamental Matrix from the column of V
# corresponding to the smallest singular value
F = V[:, -1]
F = F.reshape(3, 3).T

# Enforce rank 2 constraint
U, D, V = np.linalg.svd(F)
F = U @ np.diag([D[0], D[1], 0]) @ V.T
```

$$\mathbf{p}_r^{\mathrm{T}} \mathbf{F} \mathbf{p}_l = 0$$

$$\underbrace{\begin{pmatrix} x_{r1}x_{l1}, x_{r1}y_{l1}, x_{r1}, y_{r1}x_{l1}, y_{r1}y_{l1}, y_{r1}, x_{l1}, y_{l1}, 1 \\ \vdots \\ x_{rn}x_{ln}, x_{rn}y_{ln}, x_{rn}, y_{rn}x_{ln}, y_{rn}y_{ln}, y_{rn}, x_{ln}, y_{ln}, 1 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix}}_{\mathbf{f}} = \mathbf{0}$$

```python
# build constraint matrix
A = np.stack([x2[:, 0]*x1[:, 0], x2[:, 0]*x1[:, 1], \
   x2[:, 0], x2[:, 1]*x1[:, 0], x2[:, 1]*x1[:, 1], x2[:, 1], \
   x1[:, 0], x1[:, 1], np.ones(npts)], axis=1)

# compute SVD matrix factorization
U, D, V = np.linalg.svd(A)

# extract Fundamental Matrix from the column of V
# corresponding to the smallest singular value
F = V[:, -1]
F = F.reshape(3, 3).T

# Enforce rank 2 constraint
U, D, V = np.linalg.svd(F)
F = U @ np.diag([D[0], D[1], 0]) @ V.T
```

$$\begin{pmatrix} F_{11} & F_{21} & F_{31} \\ F_{12} & F_{22} & F_{32} \\ F_{13} & F_{23} & F_{33} \end{pmatrix}$$

```
# build constraint matrix
A = np.stack([x2[:, 0]*x1[:, 0], x2[:, 0]*x1[:, 1], \
    x2[:, 0], x2[:, 1]*x1[:, 0], x2[:, 1]*x1[:, 1], x2[:, 1], \
    x1[:, 0], x1[:, 1], np.ones(npts)], axis=1)

# compute SVD matrix factorization
U, D, V = np.linalg.svd(A)

# extract Fundamental Matrix from the column of V
# corresponding to the smallest singular value
F = V[:, -1]
F = F.reshape(3, 3).T

# Enforce rank 2 constraint
U, D, V = np.linalg.svd(F)
F = U @ np.diag([D[0], D[1], 0]) @ V.T
```

$$\begin{pmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{pmatrix}$$

```python
# build constraint matrix
A = np.stack([x2[:, 0]*x1[:, 0], x2[:, 0]*x1[:, 1], \
    x2[:, 0], x2[:, 1]*x1[:, 0], x2[:, 1]*x1[:, 1], x2[:, 1], \
    x1[:, 0], x1[:, 1], np.ones(npts)], axis=1)

# compute SVD matrix factorization
U, D, V = np.linalg.svd(A)

# extract Fundamental Matrix from the column of V
# corresponding to the smallest singular value
F = V[:, -1]
F = F.reshape(3, 3).T

# Enforce rank 2 constraint
U, D, V = np.linalg.svd(F)
F = U @ np.diag([D[0], D[1], 0]) @ V.T
```

```python
# build constraint matrix
A = np.stack([x2[:, 0]*x1[:, 0], x2[:, 0]*x1[:, 1], \
    x2[:, 0], x2[:, 1]*x1[:, 0], x2[:, 1]*x1[:, 1], x2[:, 1], \
    x1[:, 0], x1[:, 1], np.ones(npts)], axis=1)

# compute SVD matrix factorization
U, D, V = np.linalg.svd(A)

# extract Fundamental Matrix from the column of V
# corresponding to the smallest singular value
F = V[:, -1]
F = F.reshape(3, 3).T

# Enforce rank 2 constraint
U, D, V = np.linalg.svd(F)
F = U @ np.diag([D[0], D[1], 0]) @ V.T
```

```
# build constraint matrix
A = np.stack([x2[:, 0]*x1[:, 0], x2[:, 0]*x1[:, 1], \
    x2[:, 0], x2[:, 1]*x1[:, 0], x2[:, 1]*x1[:, 1], x2[:, 1], \
    x1[:, 0], x1[:, 1], np.ones(npts)], axis=1)

# compute SVD matrix factorization
U, D, V = np.linalg.svd(A)

# extract Fundamental Matrix from the column of V
# corresponding to the smallest singular value
F = V[:, -1]
F = F.reshape(3, 3).T

# Enforce rank 2 constraint
U, D, V = np.linalg.svd(F)
F = U @ np.diag([D[0], D[1], 0]) @ V.T
```

8点算法

```python
# build constraint matrix
A = np.stack([x2[:, 0]*x1[:, 0], x2[:, 0]*x1[:, 1], \
    x2[:, 0], x2[:, 1]*x1[:, 0], x2[:, 1]*x1[:, 1], x2[:, 1], \
    x1[:, 0], x1[:, 1], np.ones(npts)], axis=1)

# compute SVD matrix factorization
U, D, V = np.linalg.svd(A)

# extract Fundamental Matrix from the column of V
# corresponding to the smallest singular value
F = V[:, -1]
F = F.reshape(3, 3).T

# Enforce rank 2 constraint
U, D, V = np.linalg.svd(F)
F = U @ np.diag([D[0], D[1], 0]) @ V.T
```

Python时间