# 三维视觉

# 两张图片恢复三维世界

# 原理+代码

# 运动恢复结构SfM

**已知:**

$n$个3D点$X_j$在$m$张图像中的对应点的像素坐标 $\boldsymbol{x}_{ij}$ $(i = 1, \dots, m; j = 1, \dots, n)$

且$\boldsymbol{x}_{ij} = \boldsymbol{M}_i X_j = \boldsymbol{K}_i [\boldsymbol{R}_i \ \boldsymbol{T}_i] X_j \quad i = 1, \dots, \boxed{m}\ ;\ j = 1, \dots, \boxed{n}$

图像个数　　3D点个数

其中，$\boldsymbol{M}_i$，$\boldsymbol{K}_i$，$[\boldsymbol{R}_i \ \boldsymbol{T}_i]$为第$i$张图像对应的摄像机的投影矩阵、内参数及外参数矩阵

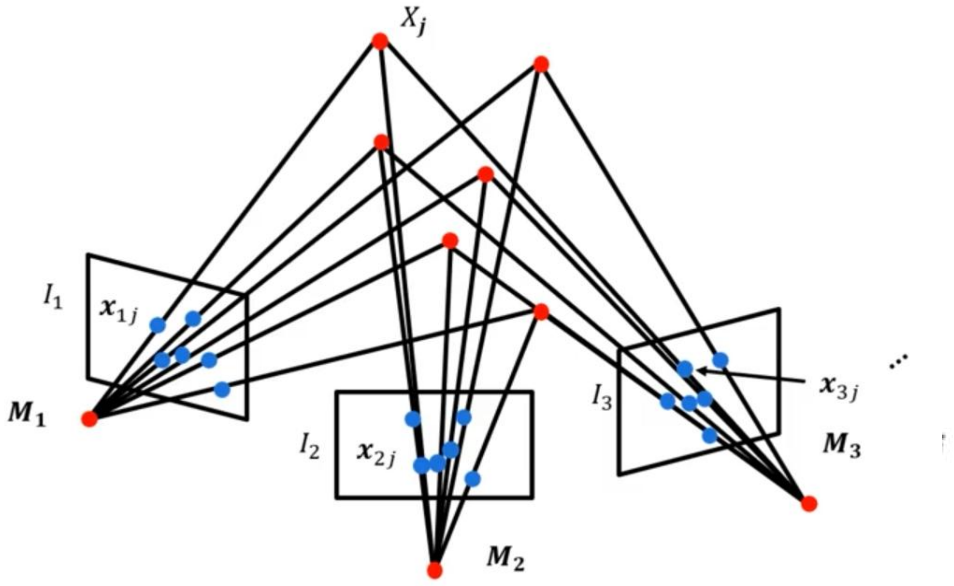
**求解:**

$m$个摄像机投影矩阵$\boldsymbol{M}_i$ $(i = 1, \cdots, m)$；

运动(motion)

$n$个三维点$X_j (j = 1, \cdots, n)$的坐标。

结构(structure)




因此，该类问题也称为“运动恢复结构问题”！

# 欧式结构恢复问题

实际情况是已知m张图片

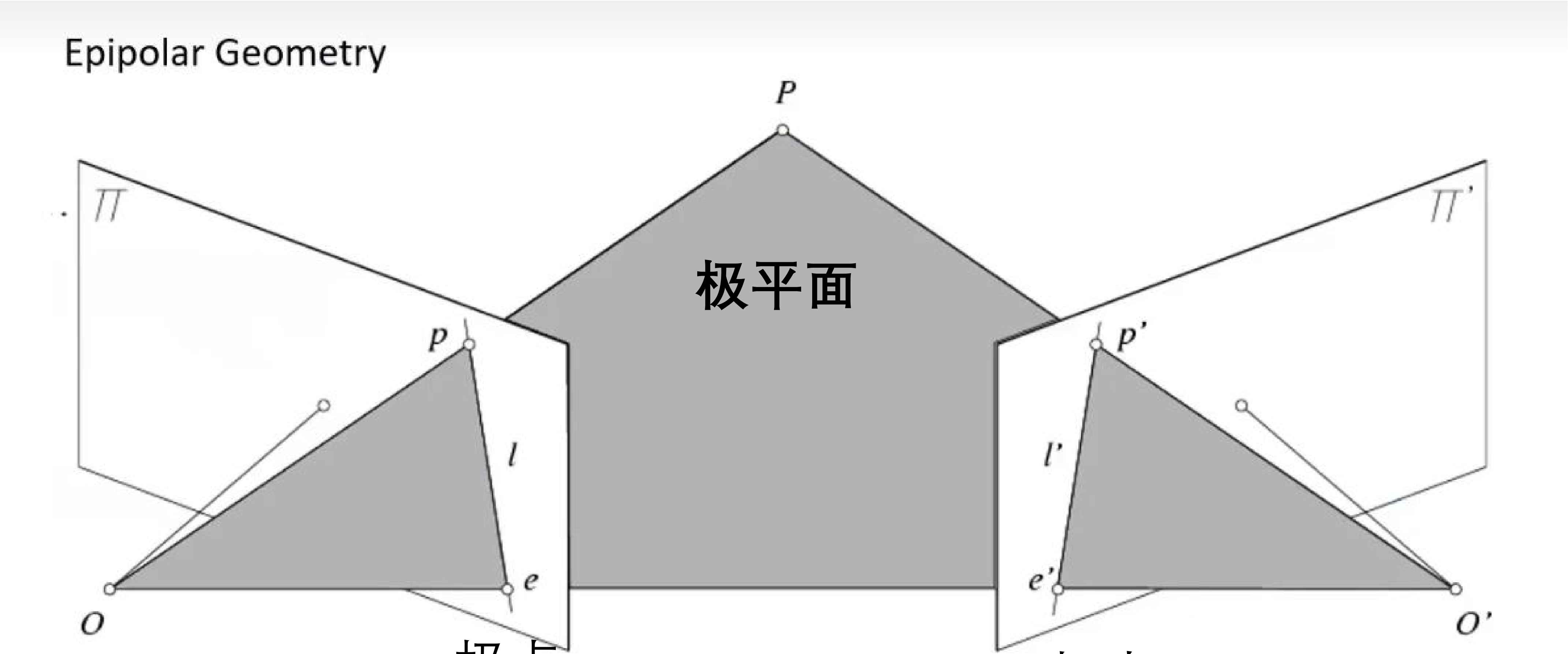
- 已知：
  - n个三维点$X_j(j=1,..,n)$在m张图片中的对应点的像素坐标$x_{ij}$
  - m张图片对应的摄像机内参矩阵$K_i(i=1,..,m)$和外参矩阵$[R_i\ T_i]$

$$x_{ij} = M_i X_j = K_i[R_i\ T_i]X_j \quad i = 1,..,m; j = 1,..,n$$

- 求解：
  - n个三维点$X_j$的坐标
  - m个摄像机的外参$R_i$和$T_i$

# 对极几何

描述**两幅视图**之间的**内在射影关系**




***e***极点

***l***极线

***OO'***基线

***e'***极点

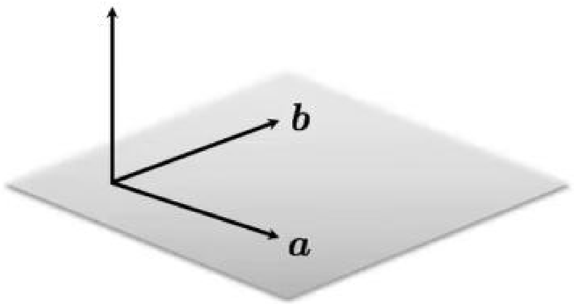
***l'***极线

## 复习线性代数的“叉积”

### Vector (cross) product

takes two vectors and returns a vector <u>perpendicular</u> to both

$$\boldsymbol{c} = \boldsymbol{a} \times \boldsymbol{b}$$




$$\boldsymbol{c} \cdot \boldsymbol{a} = 0 \qquad \boldsymbol{c} \cdot \boldsymbol{b} = 0$$

$$\boldsymbol{a} \times \boldsymbol{b} = \begin{bmatrix} a_2b_3 - a_3b_2 \\ a_3b_1 - a_1b_3 \\ a_1b_2 - a_2b_1 \end{bmatrix}$$

cross product of two vectors in the same direction is zero vector

$$\boldsymbol{a} \times \boldsymbol{a} = 0$$

remember this!!!

**cross product**

$$\boldsymbol{a} \times \boldsymbol{b} = \begin{bmatrix} a_2b_3 - a_3b_2 \\ a_3b_1 - a_1b_3 \\ a_1b_2 - a_2b_1 \end{bmatrix}$$

Can also be written as a matrix multiplication

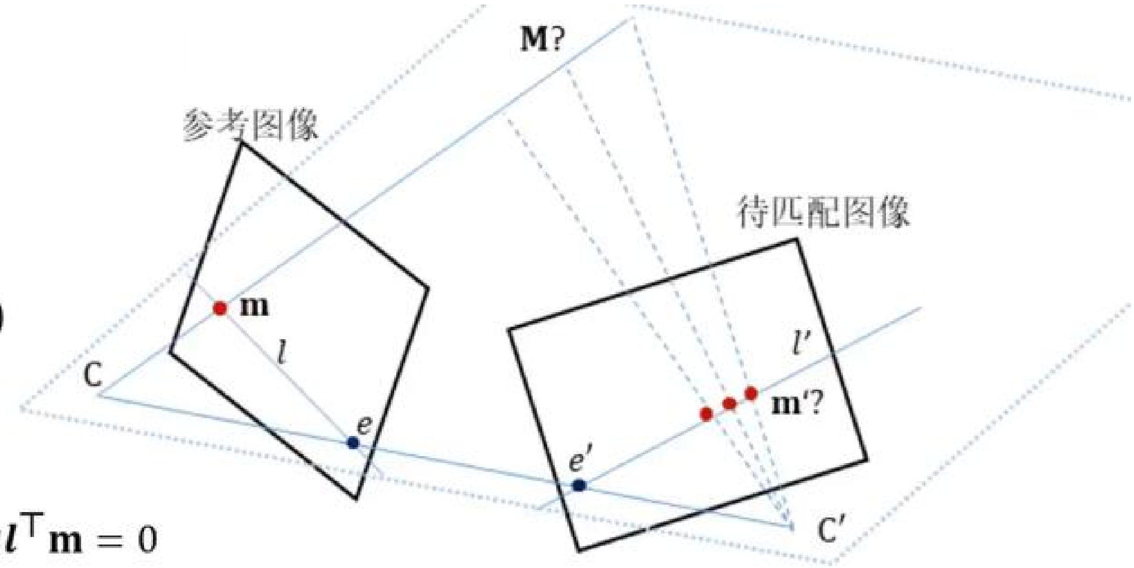
$$\boldsymbol{a} \times \boldsymbol{b} = [\boldsymbol{a}]_\times \boldsymbol{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

**Skew symmetric**
**反对称矩阵**

# 基础矩阵（Fundamental Matrix）

1, 基础矩阵
Fundamental matrix

(Faugeras and Luong, 1992)




点$\mathbf{m}$在线$\boldsymbol{l}$上可以表示为$\boldsymbol{l}^{\mathsf{T}}\mathbf{m}=0$

线$\boldsymbol{l}$穿过点$\mathbf{m}$和极点$\mathbf{e}$ ⇨ $\boldsymbol{l}\sim[\mathbf{e}]_{\times}\mathbf{m}$

极线$l$和另一幅影像上的极线$\boldsymbol{l}'$满足$\boldsymbol{l}'\sim\mathbf{H}^{-\mathsf{T}}\boldsymbol{l}$

$\mathbf{H}^{-\mathsf{T}}$的引入是根据下面推导出来

$\boldsymbol{m}'\sim\mathbf{H}\boldsymbol{m}\,,\ \boldsymbol{l}^{\mathsf{T}}\boldsymbol{m}=\ \boldsymbol{l}'^{\mathsf{T}}\boldsymbol{m}'=\mathbf{0}$

H（Homography Matrix，单应性矩阵）：描述两个平面之间的相对位置关系，建立m和m’之间的联系。

$$\boldsymbol{l}'\sim\mathbf{H}^{-\mathsf{T}}[\mathbf{e}]_{\times}\mathbf{m} \qquad \boldsymbol{l}'\sim\mathbf{F}\mathbf{m} \qquad \mathbf{m}'^{\mathsf{T}}\mathbf{F}\mathbf{m}=0$$

F（Fundamental Matrix，基础矩阵）：建立m和极线l’之间的联系。

**F叫做基础矩阵**

# 本质矩阵（Essential Matrix）

## F vs E

本质矩阵**E**是面向像空间坐标系（摄像机坐标系），不包括摄像机的内参信息。
但是研究像素点在另一个视图上的对极线，需要用摄像机的内参信息将摄像机坐标系和像平面坐标系联系起来。
设$\mathbf{K}_0$和$\mathbf{K}_1$分别为两个摄像机内参数矩阵，归一化的摄像机像空间坐标到像平面像素坐标表示可以通过$\mathbf{m}_0 = \mathbf{K}_0\mathbf{M}^0$计算。
基础矩阵**F**中编码了两幅影像的对极几何关系，可以推导**E**和基础矩阵**F**的关联

$$(\mathbf{K}_1\mathbf{M}^1)^\top\,\mathbf{F}\,(\mathbf{K}_0\mathbf{M}^0) = (\mathbf{M}^1)^\top\,\mathbf{E}\,\mathbf{M}^0 = 0$$

$$(\mathbf{K}_1\mathbf{M}^1)^\top\,\mathbf{F}\,\mathbf{K}_0 = (\mathbf{M}^1)^\top\,\mathbf{E}$$

$$\mathbf{F} = \mathbf{K}_1^{-\top}\,\mathbf{E}\,\mathbf{K}_0^{-1}$$

$$\mathbf{E} = \mathbf{K}_1^\top\mathbf{F}\mathbf{K}_0$$

本质矩阵**E**在成像坐标上作用
基础矩阵**F**在像素坐标上作用

基本矩阵是基本矩阵的推广，其中去掉了单位矩阵的假设

# 8-point algorithm

*How would you solve for F?*

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

8点法求基础矩阵F

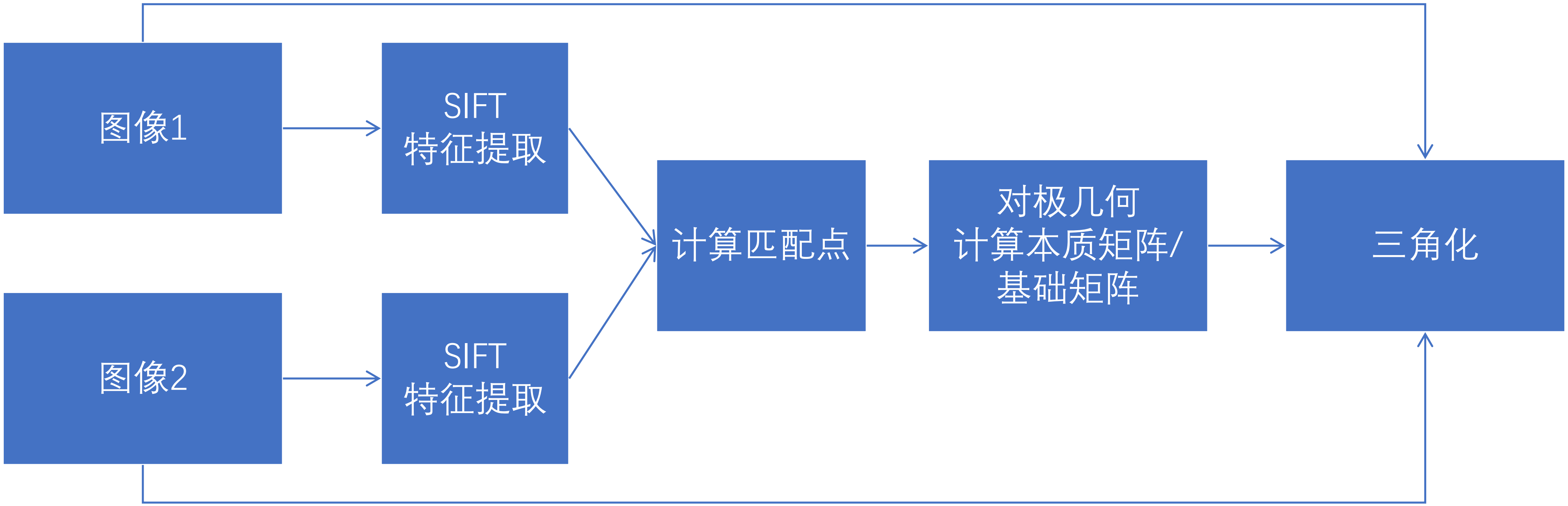
The Eight-Point Algorithm
(Longuet-Higgins. 1981)

$$(u, v, 1)\begin{pmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{pmatrix}\begin{pmatrix} u' \\ v' \\ 1 \end{pmatrix} = 0 \quad \Rightarrow \quad (uu', uv', u, vu', vv', v, u', v', 1)\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = 0$$

The 8-point algorithm

# 3D Reconstruction

- 两张图片重建三维点



# 欧式结构恢复问题

- **求解步骤：**
  - 1. 计算配准点，通过基于SIFT特征的图像配准方法
  - 2. 计算基础矩阵$F$，通过归一化八点法
  - 3. 计算本质矩阵$E = {K_2}^T F K_1$
  - 4. 本质矩阵分解，$E -> R$、$T$
  - 5. 三角化求解三维点$X_j$坐标

➢ 线性法：

$$\begin{cases} x_{1j} = M_1 X_j = K_1[I\ 0]X_j \\ x_{2j} = M_2 X_j = K_2[R\ T]X_j \end{cases}$$

➢ 非线性法：

$$P_j^* = \underset{P}{\operatorname{argmin}}\left(d\left(p_{1j}, M_1 X_j\right) + d\left(p_{2j}, M_2 X_j\right)\right)$$

**步骤1：SVD分解** $E = U\mathrm{diag}(1,1,0)V^T$

$$W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

**步骤2：** $\det UWV^T)UWV^T$ 或 $(\det UW^TV^T)UW^TV^T$

$\mathrm{T} = \pm u_3$

**步骤3：**

$$\boxed{\begin{cases} R = (\det UWV^T)UWV^T & T = u_3 \\ R = (\det UWV^T)UWV^T & T = -u_3 \\ R = (\det UW^TV^T)UW^TV^T & T = u_3 \\ R = (\det UW^TV^T)UW^TV^T & T = -u_3 \end{cases}}$$

**步骤4：通过重建单个或多个点找出正确解**

# 三角化

## 三角化（线性解法）

**问题：已知$p$和$p'$，$K$和$K'$以及$R,T$**

**求解：$P$点的三维坐标？**

**寻找$P$，满足：**

$$\begin{cases} p = MP = K[I\ 0]P \\ p' = M'P = K'[R\ T]P \end{cases} \Rightarrow$$

$$u = \frac{m_1P}{m_3P} \quad \rightarrow m_1P - u(m_3P) = 0$$

$$v = \frac{m_2P}{m_3P} \quad \rightarrow m_2P - v_i(m_3P) = 0$$

$$u' = \frac{m_1'P}{m_3P} \quad \rightarrow m_1'P - u'(m_3'P) = 0$$

$$v' = \frac{m_2'P}{m_3'P} \quad \rightarrow m_2'P - v'(m_3'P) = 0$$

$$\Rightarrow AP = 0$$

$$A = \begin{bmatrix} um_3 - m_1 \\ vm_3 - m_2 \\ u'm_3' - m_1' \\ v'm_3' - m_2' \end{bmatrix}$$

- 方程数4个
- 未知参数3个

超定齐次线性方程组

最小二乘解：1. 矩阵 $A$ 进行奇异值分解 $A = UDV^T$

2. $P$ 为 $V$ 矩阵的最后一列

谢谢观看