

Visual SLAM

基于OpenCV项目实施

大连理工大学 电信学部

郁森



前景提要

◆ 根据项目要求，

◆ **前端特征检测**

使用OpenCV提供的SIFT特征检测器

◆ **前端特征点匹配**

进行SIFT和光流匹配筛选有效点

◆ **对极几何运动求解**

使用极线几何8点求解摄像头运动位姿

◆ **运动距离估算**

求解最优地平面对运动距离

◆ **帧运动骨架优化**

使用FrameSLAM

前端特征检测



Dataset: 00 / Index: 000001

- ◆ 使用SIFT(Scale Invariant Feature Transform)进行KeyPoint检测

$KeyPoints_i = SiftDetector.detect(Image_i, number\ of\ Features)$

前端特征匹配-1



$$Match_{i,j} = SiftDetector.match(KeyPoints_i, KeyPoints_j)$$

前端特征匹配-2



◆ 使用光流算法对原有匹配点进行过滤

$Guess_j = OpticalFlow(Image_i, Image_j, KeyPoints_i \text{ of } Match_{i,j})$

for $Pair_k$ in $Match_{i,j}$, $k < Match.size()$

if $\|Pair^k.second - Guess_j^k\|^2 < Threshold$ then

retain

else **remove**

运动估算

◆ 考虑摄像头第 i 帧的姿态为 X_i

$$X_i = \begin{bmatrix} Dir_i & Pos_i \\ 0 & 1 \end{bmatrix}, \text{ 其中:}$$

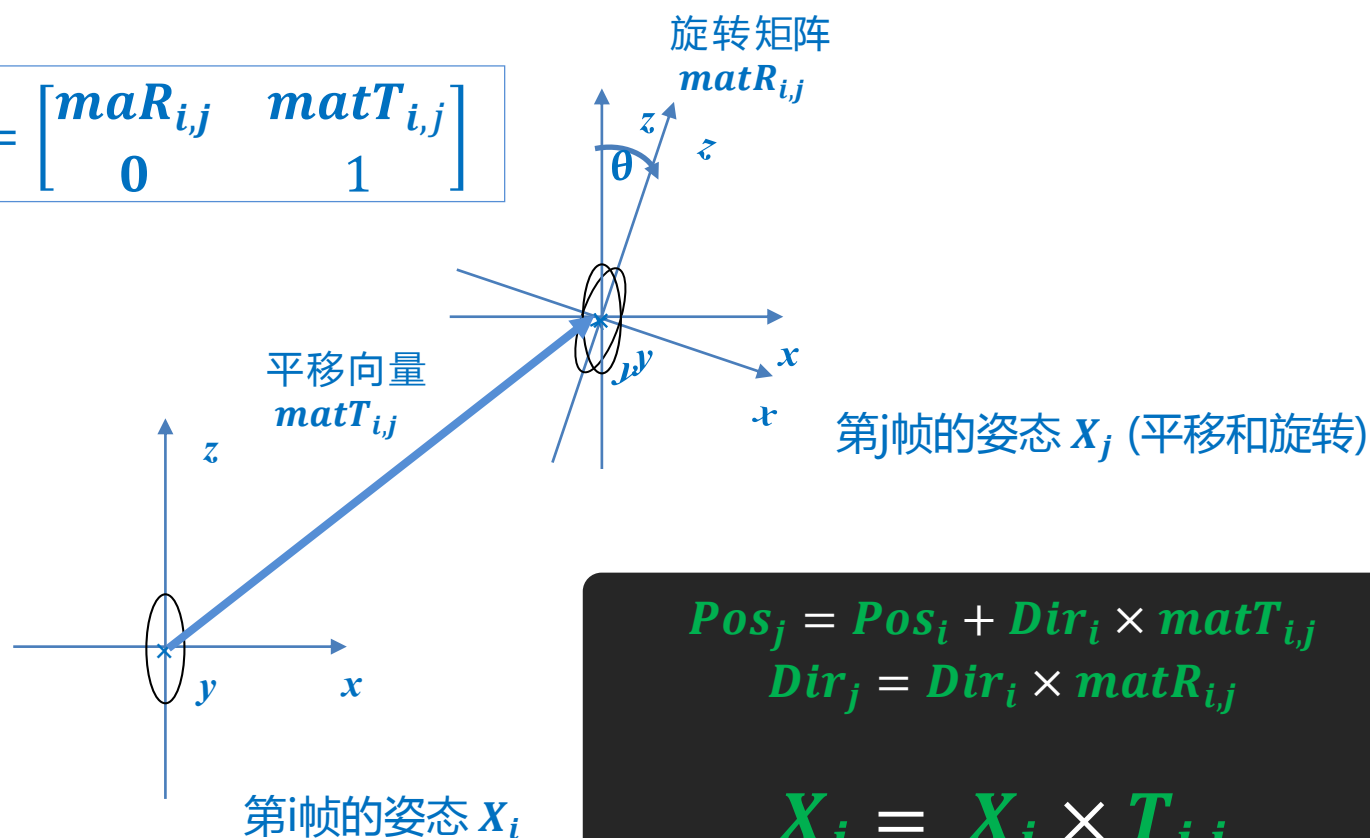
$$Pos_i = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \text{ 为当前摄像头空间位置}$$

$$Dir_i = \begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix} \text{ 为从 } \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \text{ 方向旋转到当前摄像头方向的旋转矩阵}$$

运动估算

◆ 考虑摄像头姿态 X_i 和 X_j 之间的运动关系 $T_{i,j}$, 有

$$T_{i,j} = \begin{bmatrix} \text{mat}R_{i,j} & \text{mat}T_{i,j} \\ 0 & 1 \end{bmatrix}$$



$$\begin{aligned} \text{Pos}_j &= \text{Pos}_i + \text{Dir}_i \times \text{mat}T_{i,j} \\ \text{Dir}_j &= \text{Dir}_i \times \text{mat}R_{i,j} \end{aligned}$$

$$X_j = X_i \times T_{i,j}$$

运动估算

- ◆ 考虑同一个空间点在第*i*和第*j*帧图像的二维映射关系，有

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix}_i = \mathbf{Proj}_{camera} \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \dots\dots\dots \text{对于第 } i \text{ 帧}$$

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix}_j = \mathbf{Proj}_{camera} \times \begin{bmatrix} \mathbf{matR}^T & -\mathbf{matR}^T \mathbf{matT} \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \dots\dots\dots \text{对于第 } j \text{ 帧}$$

- ◆ 以上 \mathbf{matR} 和 \mathbf{matT} 表示上一页中的标识，对于同一空间点从第*i*坐标系转移到第*j*坐标系需要进行的变换。

运动估算

- 1. 给定一些(≥ 8)的平面对应点对, 可以求解出其极几何意义上的 *Fundamental Matrix*
- 2. 结合摄像头内参 $Proj_{camera}$ 可以得到 *Essential Matrix*
- 3. 并可以求解出对应的运动 $matR$ 和 $matT$

$$\begin{aligned} matF &= cvFindFundamentalMat(Match_{i,j}) \\ matE &= Proj_{camera}^T \times matF \times Proj_{camera} \\ matE &= matU \times matS \times matVt \quad // \text{奇异值分解} \end{aligned}$$

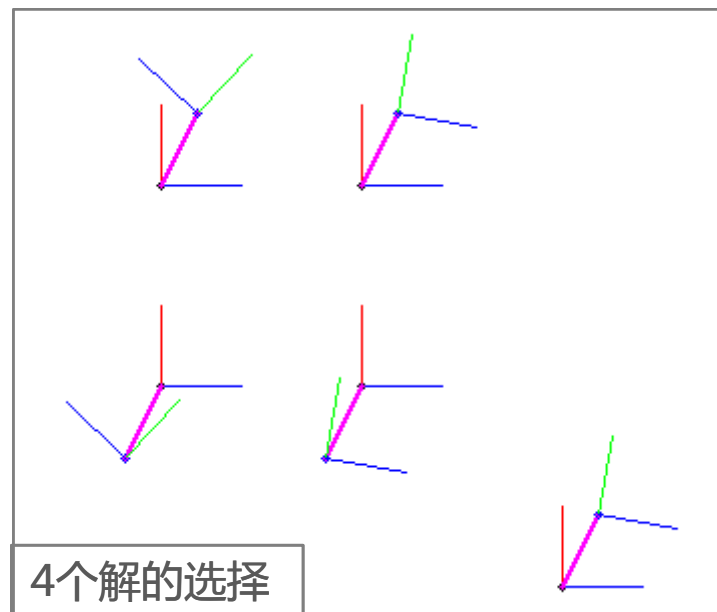
$$R = \begin{cases} matU \times matW^T \times matVt \\ matU \times matW \times matVt \end{cases}, \quad matW = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$
$$T = \begin{cases} matU \times [0 & 0 & 1]^T \\ matU \times [0 & 0 & -1]^T \end{cases}$$

运动估算

- ◆ 以上 $matR$ 和 $matT$ 表示上2页中的标识，对于同一空间点从第 i 坐标系转移到第 j 坐标系需要进行的变换。
- ◆ 而以上 R 和 T 表示上一页中的标识，对于同一空间点从第 j 坐标系转移到第 i 坐标系需要进行的变换。

$$\begin{aligned} matR &= R^T \\ matT &= -R^T \times T \end{aligned}$$

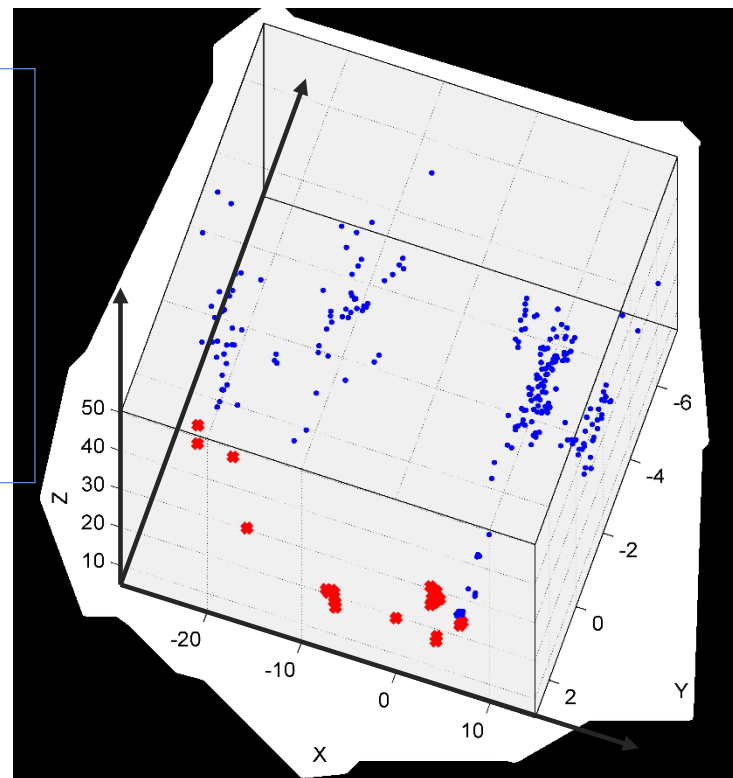
- ◆ 对于4个解的选择，按照每个解把平面点对投影成三维点，计算两个摄像头能看到的点的数量，选择点数量最多的那个解



距离估算

- 在上一步求解中，计算得到的 T 位移向量是未知长度的，因此，需要通过其他途径来计算位移的距离
- 考虑以下相似关系，即只要求地平面到摄像头中心的距离即可

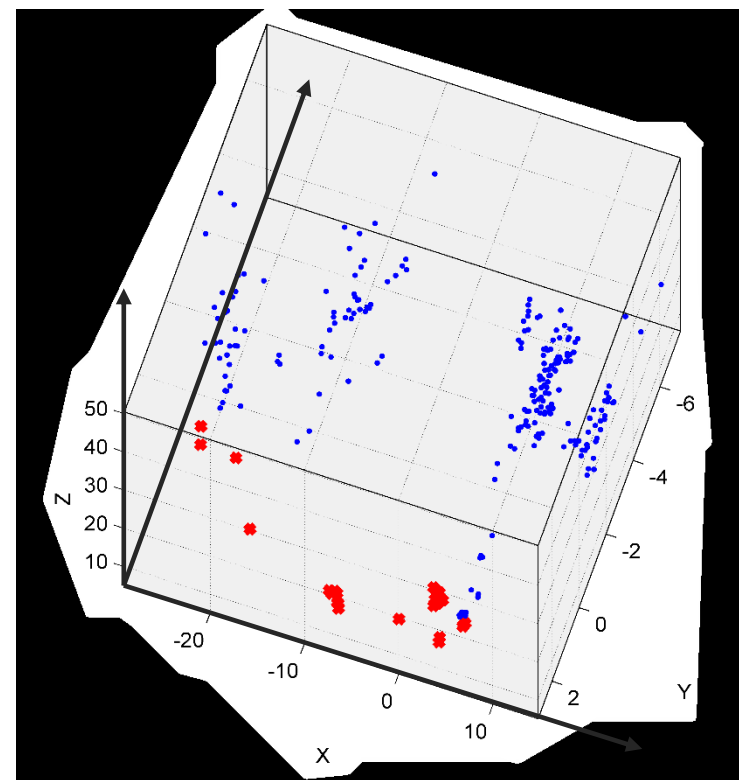
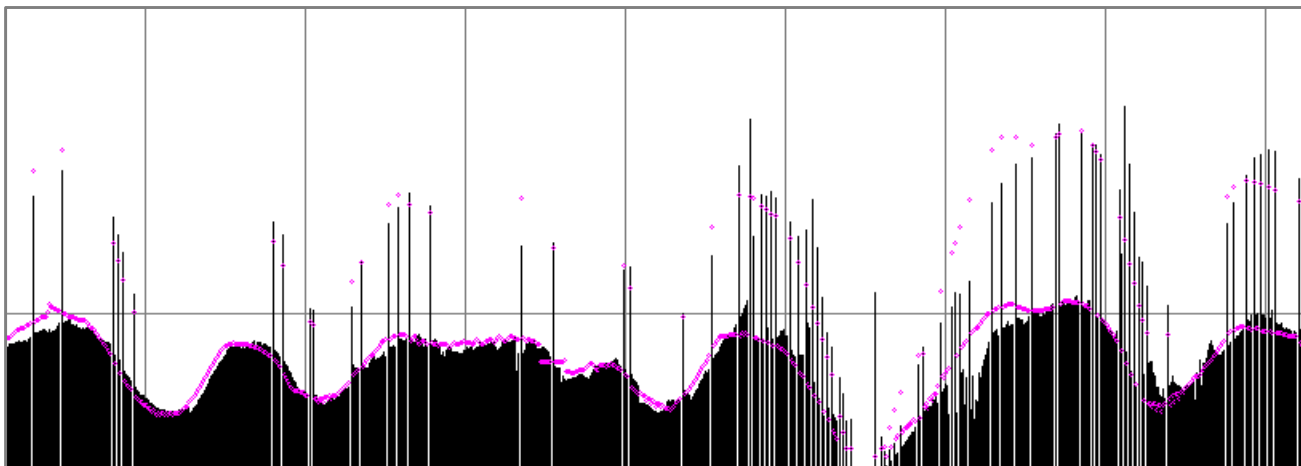
$$\frac{1.65m \text{ 摄像头标定距离}}{\|T\| \text{ 运动距离}} = \frac{H_{ground} \text{ 地平面到摄像头的距离}}{1.0}$$



距离估算

- 在上一步求解中，计算得到的 T 位移向量是未知长度的，因此，需要通过其他途径来计算位移的距离
- 考虑以下相似关系，即只要求地平面对摄像头中心的距离即可

图为Ground Truth的运动距离和我估算的运动距离
竖线为（跳帧）



骨架优化

- ◆ 由于每次都只考虑连续帧间运动(即第 i 和第 $i + 1$ 帧), 会导致误差的累计, 导致后续结果差异巨大。因此考虑增加多帧间运动约束(来源于FrameSLAM)

$X_j = X_i \times T_{i,j}$ 对 i, j 进行运动估算得到的约束

记 \hat{X}_k 为最优估计值, 则对约束 i, j 有一个误差, 记为 $e_{i,j}$

$$e_{i,j} = X_j - X_i \times T_{i,j}$$

那么考虑优化所有 \hat{X}_k 使得代价和函数 $\sum \|e_{i,j}\|_2^2$ 最小

骨架优化

◆ 对上式应用最小二程法可得

$$\begin{aligned} Cost &= \sum \|e_{i,j}\|_2^2 \\ &= \sum e_{i,j}^T \times e_{i,j} \\ &= \sum (X_j^T - T_{i,j}^T X_i^T) \times (X_j - X_i T_{i,j}) \end{aligned}$$

◆ 求偏导数得到

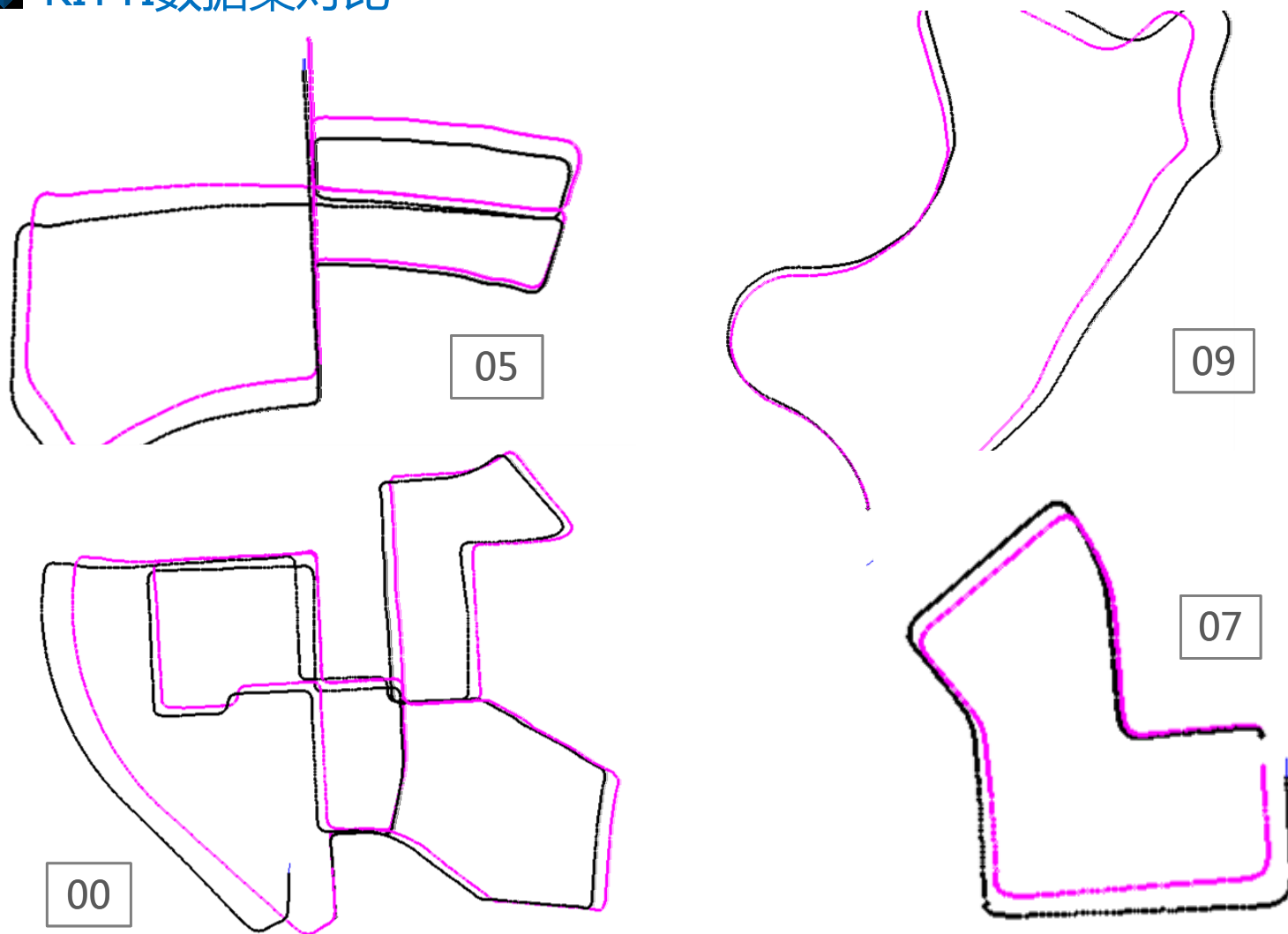
$$\frac{\partial \|e_{i,j}\|_2^2}{\partial X_j} = 2(X_j - X_i T_{i,j}) \quad \frac{\partial \|e_{i,j}\|_2^2}{\partial X_i} = 2(-X_j T_{i,j}^T + X_i T_{i,j}^T T_{i,j}^T)$$

◆ 然而对于 X_k 做梯度下降时，修正一次就不是旋转矩阵，因此需要使用 *Lagrange* 乘数法加入约束

$$Dir^T \times Dir = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

项目演示

◆ KITTI数据集对比





谢 谢

欢迎各位老师同学批评指正

大连理工大学 电信学部
郁森