

VINS-Mono with Line Feature

贺一家

December 3, 2019

Abstract

这是 2017 年写 PL-VIO 时的笔记，该笔记主要是在 VINS-Mono 的基础上添加直线特征，对直线的各类雅克比公式进行推导整理，方便以后代码实现和讨论交流。

1 直线参数化方法

三维空间中的直线可以用两个过直线的三维空间点或者两个平面来定义。但是，相对于刚体变换在三维空间中的6个自由度（3个旋转，3个平移），直线在三维空间中只有4个自由度，可以简单理解为直线绕自身直线方向的旋转和沿着直线方向的平移这两个自由度没有了。在本节中将介绍直线的两个参数化方法：普吕克直线参数法和正交表达法。其中普吕克直线参数法通过两个平面或者两个三维空间点得到，用该方法可以方便直线参与几何运算，而正交表达法则用于直线参与优化运算。

1.1 普吕克直线表示方法

假设三维空间中两个点 $\mathbf{P}_1, \mathbf{P}_2 \in \mathbb{R}^3$ ，两个点的齐次坐标形式为 $\bar{\mathbf{P}}_1 = [x_1, y_1, z_1, w_1]^T, \bar{\mathbf{P}}_2 = [x_2, y_2, z_2, w_2]^T$ 。其中 w_1, w_2 是为了齐次坐标的更通用的表达，表示三维空间点时有 $w_1 = w_2 = 1$ 。空间中过这两个点的直线方程可以用6个参数的普吕克坐标表达 $\mathbf{L} = [L_1, L_2, L_3, L_4, L_5, L_6]^T$ ：

$$\begin{aligned} L_1 &= y_1 z_2 - z_1 y_2 & L_4 &= w_1 x_2 - w_2 x_1 \\ L_2 &= z_1 x_2 - x_1 z_2 & L_5 &= w_1 y_2 - w_2 y_1 \\ L_3 &= x_1 y_2 - y_1 x_2 & L_6 &= w_1 z_2 - w_2 z_1 \end{aligned} \tag{1}$$

当 $w_1 = w_2 = 1$ 时，直线的方向向量为：

$$\mathbf{d} = \mathbf{P}_2 - \mathbf{P}_1 = [L_4, L_5, L_6]^T \tag{2}$$

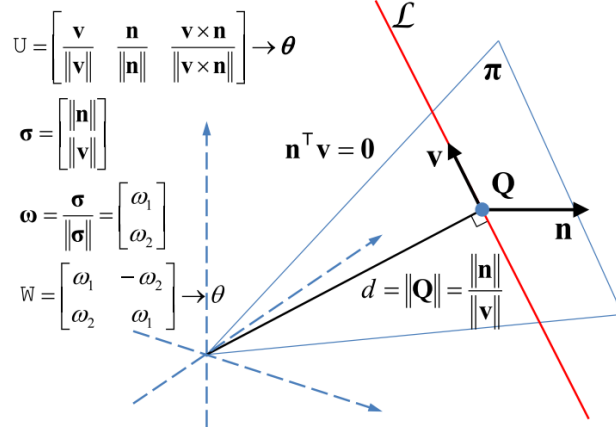


Figure 1: 直线在三维空间中的两种参数化方法。普吕克直线坐标是由平面法向 \mathbf{n} 和直线方向 \mathbf{v} 组成。直线的正交表达由 \mathbf{U} 和 \mathbf{W} 组成，其中 $\mathbf{U} \in SO(3), \mathbf{W} \in SO(2)$

由坐标系原点 \mathbf{O} 和 $\mathbf{P}_1, \mathbf{P}_2$ 三点构成的平面的法向量为：

$$\mathbf{n} = \mathbf{P}_1 \times \mathbf{P}_2 = [L_1, L_2, L_3]^T \quad (3)$$

所以普吕克坐标可以由一个方向方向量和过坐标原点及直线的平面的法向量组成 $\mathbf{L} = [\mathbf{n}^T, \mathbf{d}^T]^T$ ，如图1所示。如果选择另外两个在直线上的点来表示直线：

$$\mathbf{P}'_1 = \lambda \mathbf{P}_1 + (1 - \lambda) \mathbf{P}_2, \quad \mathbf{P}'_2 = \mu \mathbf{P}_1 + (1 - \mu) \mathbf{P}_2$$

新的方向向量和法向量为：

$$\mathbf{n}' = \mathbf{P}'_1 \times \mathbf{P}'_2 = (\lambda - \mu) \mathbf{n}$$

$$\mathbf{d}' = \mathbf{P}'_2 - \mathbf{P}'_1 = (\lambda - \mu) \mathbf{d}$$

由此可以知道同一直线的不同普吕克坐标参数相差一个尺度因子，即虽然普吕克坐标是6个参数，但只有5个自由度。另外 $\mathbf{n}^T \mathbf{d} = 0$ 构建的一个约束将直线在空间中的自由度降低到4个。需要注意的是普吕克坐标中的法向量和单位向量都不必是单位向量。另外，假设坐标原点和直线垂直相交于点 \mathbf{Q} ，有

$$\|\mathbf{n}\| = \|\mathbf{Q} \times \mathbf{d}\| = \|\mathbf{Q}\| \|\mathbf{d}\| \sin(\pi/2) \rightarrow \|\mathbf{Q}\| = \frac{\|\mathbf{n}\|}{\|\mathbf{d}\|} \quad (4)$$

即原点到直线的距离 $d = \|\mathbf{Q}\|$ 。

然而在SLAM过程中，通过图像往往无法得到过直线的两个点的三维坐标，而只能通过两个不同时刻的图像坐标获取直线所在的平面。因此在初始化直线普吕克坐标时，采用两平面交线的形式来计算，如图2所示。假

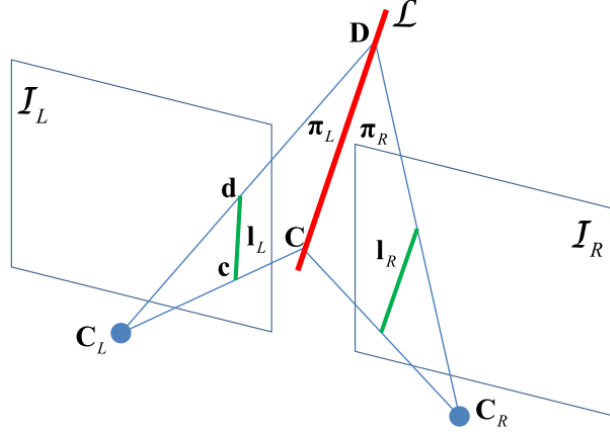


Figure 2: 空间直线 \mathbf{L} 可以由两个平面 π_L, π_R 的相交线得到。

设归一化图像平面 \mathbf{I} 上检测到的线段的两个端点坐标为 $\mathbf{s} = [u_s, v_s, 1]^T$, $\mathbf{e} = [u_e, v_e, 1]^T$ 以及该时刻相机坐标系的原点 $\mathbf{o} = [0, 0, 0]^T$, 该三点确定的平面系数 $\pi = [a, b, c, d]^T$, 其中

$$a(x - x_0) + b(y - y_0) + c(z - z_0) = ax + by + cz + d = 0 \quad (5)$$

其中 (a, b, c) 为平面的法向量, $d = ax_0 + by_0 + cz_0$:

$$\mathbf{n} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} = (\mathbf{s} - \mathbf{o}) \times (\mathbf{e} - \mathbf{o}), \quad d = \mathbf{n}^T \mathbf{o}$$

对于双目, 同一时刻的左目和右目同时检测到直线线段, 因此平面 π_L, π_R 可以得到对偶的普吕克矩阵 \mathcal{L}^* :

$$\mathcal{L}^* = \begin{bmatrix} [\mathbf{d}]_{\times} & \mathbf{n} \\ -\mathbf{n}^T & 0 \end{bmatrix} = \pi_L \pi_R^T - \pi_R \pi_L^T \in \mathbb{R}^{4 \times 4} \quad (6)$$

可以直接从矩阵 \mathcal{L}^* 提取空间三维直线的普吕克坐标 \mathbf{L} 。

1.2 正交表达法

普吕克坐标表达法自身参数之间存在约束, 不利于参数优化。正交表达只需要4个变量 $(\boldsymbol{\theta}, \phi) \in \mathbb{R}^3 \times \mathbb{R}^1$ 来表示空间直线, 对应两个正交矩阵 $(\mathbf{U}, \mathbf{W}) \in SO(3) \times SO(2)$ 。正交表达可以通过对普吕克坐标 $\mathcal{L} = [\mathbf{n}^T, \mathbf{d}^T]^T$ 进行QR分解可

以得到。但在实际应用过程中，可以很轻易的构造如下矩阵：

$$\begin{bmatrix} \mathbf{n} & \mathbf{d} \end{bmatrix} = \begin{bmatrix} \frac{\mathbf{n}}{\|\mathbf{n}\|} & \frac{\mathbf{d}}{\|\mathbf{d}\|} & \frac{\mathbf{n} \times \mathbf{d}}{\|\mathbf{n} \times \mathbf{d}\|} \end{bmatrix} \begin{bmatrix} \|\mathbf{n}\| & 0 \\ 0 & \|\mathbf{d}\| \\ 0 & 0 \end{bmatrix} \quad (7)$$

因此可以很容易得到一个正交矩阵：

$$\begin{aligned} \mathbf{U} &= \mathbf{R}_x(\theta_1) \mathbf{R}_y(\theta_2) \mathbf{R}_z(\theta_3) = \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \mathbf{u}_3 \end{bmatrix} \\ &= \begin{bmatrix} \frac{\mathbf{n}}{\|\mathbf{n}\|} & \frac{\mathbf{d}}{\|\mathbf{d}\|} & \frac{\mathbf{n} \times \mathbf{d}}{\|\mathbf{n} \times \mathbf{d}\|} \end{bmatrix} \end{aligned} \quad (8)$$

对于后面矩阵的两个元素 $\|\mathbf{n}\|, \|\mathbf{d}\|$ 实际上只有一个自由度，把它们归一化以后，就能用一个 $SO(2)$ 来更新它

$$\begin{aligned} \mathbf{W} &= \begin{bmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{bmatrix} = \begin{bmatrix} w_1 & -w_2 \\ w_2 & w_1 \end{bmatrix} \\ &= \frac{1}{\sqrt{(\|\mathbf{n}\|^2 + \|\mathbf{d}\|^2)}} \begin{bmatrix} \|\mathbf{n}\| & -\|\mathbf{d}\| \\ \|\mathbf{d}\| & \|\mathbf{n}\| \end{bmatrix} \end{aligned} \quad (9)$$

原点到直线的距离 $d = \frac{w_1}{w_2} = \frac{\|\mathbf{n}\|}{\|\mathbf{d}\|}$ ，所以 \mathbf{W} 里编码了距离信息 d 。由正交表达也很容易得到普吕克坐标：

$$\mathcal{L}' \sim [w_1 \mathbf{u}_1^T, w_2 \mathbf{u}_2^T]^T \quad (10)$$

注意这里不是等号，他们之间在数值上差了一个尺度，但是对应同一条三维空间直线。

$$\mathcal{L}' = \frac{1}{\sqrt{(\|\mathbf{n}\|^2 + \|\mathbf{d}\|^2)}} \mathcal{L} \quad (11)$$

2 VINS系统里的直线观测模型

该部分主要对直线的重投影误差和相关的雅克比进行推导。

2.1 重投影误差

在介绍直线的重投影误差前，需要知道三维直线的普吕克坐标在各个坐标系之间的转换矩阵，假设世界坐标系到相机坐标系之间的旋转和平移对应为 $\mathbf{R}_{cw}, \mathbf{t}_{cw}$ ，对于世界坐标系下的三维直线 \mathcal{L}_w 可以利用如下方程转换到相机坐标系下：

$$\mathcal{L}_c = \begin{bmatrix} \mathbf{n}_c \\ \mathbf{d}_c \end{bmatrix} = \mathcal{T}_{cw} \mathcal{L}_w = \begin{bmatrix} \mathbf{R}_{cw} & [\mathbf{t}_{cw}]_{\times} \mathbf{R}_{cw} \\ \mathbf{0} & \mathbf{R}_{cw} \end{bmatrix} \mathcal{L}_w \quad (12)$$

同理，可以定义逆变换矩阵，将直线普吕克坐标系从相机坐标系转换到世界坐标系下：

$$\mathcal{T}_{cw}^{-1} = \begin{bmatrix} \mathbf{R}_{cw}^T & [-\mathbf{R}_{cw}^T \mathbf{t}_{cw}]_{\times} \mathbf{R}_{cw}^T \\ \mathbf{0} & \mathbf{R}_{cw}^T \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{cw}^T & -\mathbf{R}_{cw}^T [\mathbf{t}_{cw}]_{\times} \\ \mathbf{0} & \mathbf{R}_{cw}^T \end{bmatrix} \quad (13)$$

从相机坐标系到投影到图像平面的投影矩阵为：

$$\mathbf{l} = \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix} = \mathcal{K} \mathbf{n}_c = \begin{bmatrix} f_y & 0 & 0 \\ 0 & f_x & 0 \\ -f_y c_x & -f_x c_y & f_x f_y \end{bmatrix} \mathbf{n}_c \quad (14)$$

其中 \mathcal{K} 为相机内参数组成的矩阵。

定义重投影误差为检测的直线线段的两个端点到重投影直线上的距离：

$$\mathbf{e}_l = \begin{bmatrix} d(\mathbf{x}_s, \mathbf{l}) \\ d(\mathbf{x}_e, \mathbf{l}) \end{bmatrix} \quad (15)$$

其中 $d(\mathbf{x}, \mathbf{l})$ 表示点到直线的距离：

$$d(\mathbf{x}, \mathbf{l}) = \frac{\mathbf{x}^T \mathbf{l}}{\sqrt{l_1^2 + l_2^2}} \quad (16)$$

$\mathbf{x}_s, \mathbf{x}_e$ 表示线段在图像平面的两个端点。

2.2 重投影误差对状态量的雅克比

2.2.1 重投影误差对直线正交参数增量的雅克比

$$\frac{\partial \mathbf{e}_l}{\partial \mathbf{l}} \frac{\partial \mathbf{l}}{\partial \mathcal{L}_c} \frac{\partial \mathcal{L}_c}{\partial \mathcal{L}_w} \frac{\partial \mathcal{L}_w}{\partial \mathcal{O}} \frac{\partial \mathcal{O}}{\partial \begin{bmatrix} \delta \boldsymbol{\theta} \\ \delta \phi \end{bmatrix}} \quad (17)$$

其中普吕克坐标 \mathcal{L} 只跟正交矩阵 \mathbf{U}, \mathbf{W} 中部分变量有关 $\mathcal{O} = [\mathbf{u}_1, \mathbf{u}_2, w_1, w_2]^T$ 。

首先计算重投影误差对重投影直线的导数

$$\begin{aligned} \frac{\partial \mathbf{e}_l}{\partial \mathbf{l}} &= \begin{bmatrix} \frac{\partial e_1}{\partial \mathbf{l}} \\ \frac{\partial e_2}{\partial \mathbf{l}} \end{bmatrix} = \begin{bmatrix} \frac{\partial e_1}{\partial l_1} & \frac{\partial e_1}{\partial l_2} & \frac{\partial e_1}{\partial l_3} \\ \frac{\partial e_2}{\partial l_1} & \frac{\partial e_2}{\partial l_2} & \frac{\partial e_2}{\partial l_3} \end{bmatrix} \\ &= \begin{bmatrix} \frac{-l_1 \mathbf{x}_s^T \mathbf{l}}{(l_1^2 + l_2^2)^{(\frac{3}{2})}} + \frac{u_s}{(l_1^2 + l_2^2)^{(\frac{1}{2})}} & \frac{-l_2 \mathbf{x}_s^T \mathbf{l}}{(l_1^2 + l_2^2)^{(\frac{3}{2})}} + \frac{v_s}{(l_1^2 + l_2^2)^{(\frac{1}{2})}} & \frac{1}{(l_1^2 + l_2^2)^{(\frac{1}{2})}} \\ \frac{-l_1 \mathbf{x}_e^T \mathbf{l}}{(l_1^2 + l_2^2)^{(\frac{3}{2})}} + \frac{u_e}{(l_1^2 + l_2^2)^{(\frac{1}{2})}} & \frac{-l_2 \mathbf{x}_e^T \mathbf{l}}{(l_1^2 + l_2^2)^{(\frac{3}{2})}} + \frac{v_e}{(l_1^2 + l_2^2)^{(\frac{1}{2})}} & \frac{1}{(l_1^2 + l_2^2)^{(\frac{1}{2})}} \end{bmatrix}_{2 \times 3} \end{aligned} \quad (18)$$

投影直线 \mathbf{l} 对相机坐标系下的三维直线 \mathcal{L}_c 的雅克比

$$\frac{\partial \mathbf{l}}{\partial \mathcal{L}_c} = \begin{bmatrix} \mathcal{K} & \mathbf{0} \end{bmatrix}_{3 \times 6} \quad (19)$$

相机坐标系下的三维直线 \mathcal{L}_c 对世界坐标系下的坐标 \mathcal{L}_w 的雅克比

$$\frac{\partial \mathcal{L}_c}{\partial \mathcal{L}_w} = \mathcal{T}_{wc}^{-1} \quad (20)$$

世界坐标系下的三维直线普吕克坐标 \mathcal{L}_w 对正交表达 $\mathcal{O} = [\mathbf{u}_1, \mathbf{u}_2, w_1, w_2]^T$ 的求导

$$\begin{aligned} \frac{\partial \mathcal{L}_w}{\partial \mathcal{O}} &= \frac{\partial \begin{bmatrix} w_1 \mathbf{u}_1 \\ w_2 \mathbf{u}_2 \end{bmatrix}}{\partial \mathcal{O}} = \begin{bmatrix} \frac{\partial \mathcal{L}_w}{\partial \mathbf{u}_1} & \frac{\partial \mathcal{L}_w}{\partial \mathbf{u}_2} & \frac{\partial \mathcal{L}_w}{\partial w_1} & \frac{\partial \mathcal{L}_w}{\partial w_2} \end{bmatrix}_{6 \times (3+3+1+1)} \\ &= \begin{bmatrix} w_1 \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{u}_1 & \mathbf{0}_{3 \times 1} \\ \mathbf{0}_{3 \times 3} & w_2 \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 1} & \mathbf{u}_2 \end{bmatrix}_{6 \times 8} \end{aligned} \quad (21)$$

接下来计算正交表达自身的雅克比矩阵:

$$\frac{\partial \mathcal{O}}{\partial \begin{bmatrix} \delta \boldsymbol{\theta} \\ \delta \phi \end{bmatrix}} = \begin{bmatrix} \frac{\partial \mathbf{u}_1}{\partial \delta \boldsymbol{\theta}} & \frac{\partial \mathbf{u}_1}{\partial \delta \phi} \\ \frac{\partial \mathbf{u}_2}{\partial \delta \boldsymbol{\theta}} & \frac{\partial \mathbf{u}_2}{\partial \delta \phi} \\ \frac{\partial w_1}{\partial \delta \boldsymbol{\theta}} & \frac{\partial w_1}{\partial \delta \phi} \\ \frac{\partial w_2}{\partial \delta \boldsymbol{\theta}} & \frac{\partial w_2}{\partial \delta \phi} \end{bmatrix}_{8 \times 4} \quad (22)$$

在求解之前, 需要定义好变量的增量更新形式。由于 $\mathbf{U} \in SO(3)$, $\mathbf{W} \in SO(2)$, 可以用李代数那套来更新他们, 分别定义他们的增量为 $\delta \boldsymbol{\theta}, \delta \phi$ 。

$$\begin{aligned} \mathbf{U}' &= \mathbf{U} \exp([\delta \boldsymbol{\theta}]_{\times}) \approx \mathbf{U}(\mathbf{I} + [\delta \boldsymbol{\theta}]_{\times}) \\ \mathbf{W}' &= \mathbf{W} \exp([\delta \phi]_{\times}) = \mathbf{W} \exp\left(\begin{bmatrix} 0 & -\delta \phi \\ \delta \phi & 0 \end{bmatrix}\right) \approx \mathbf{W}\left(\mathbf{I} + \begin{bmatrix} 0 & -\delta \phi \\ \delta \phi & 0 \end{bmatrix}\right) \end{aligned} \quad (23)$$

雅克比的计算就比较容易了

$$\frac{\partial \mathbf{U}'}{\partial \delta \boldsymbol{\theta}} = \frac{\partial \mathbf{U}[\delta \boldsymbol{\theta}]_{\times}}{\partial \delta \boldsymbol{\theta}} = \frac{\delta \mathbf{U}(\delta \theta_1 \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} + \delta \theta_2 \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} + \delta \theta_3 \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix})}{\partial \delta \boldsymbol{\theta}} \quad (24)$$

上面矩阵元素较多，直接展开的话，篇幅较大，先列出对每个增量分量的雅克比：

$$\begin{aligned}
\frac{\partial \mathbf{U}'}{\partial \delta \theta_1} &= \mathbf{U} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{u}_3 & -\mathbf{u}_2 \end{bmatrix}_{3 \times 3} \\
\frac{\partial \mathbf{U}'}{\partial \delta \theta_2} &= \mathbf{U} \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} -\mathbf{u}_3 & \mathbf{0} & \mathbf{u}_1 \end{bmatrix}_{3 \times 3} \\
\frac{\partial \mathbf{U}'}{\partial \delta \theta_3} &= \mathbf{U} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{u}_2 & -\mathbf{u}_1 & \mathbf{0} \end{bmatrix}_{3 \times 3}
\end{aligned} \tag{25}$$

在前面的变量 $\mathcal{O} = [\mathbf{u}_1, \mathbf{u}_2, w_1, w_2]^T$ 只涉及到了 $\mathbf{u}_1, \mathbf{u}_2$ ，因此我们把上面公式求得的各分量雅克比取对应的列，就能得到：

$$\begin{aligned}
\frac{\partial \mathbf{u}_1}{\partial \delta \boldsymbol{\theta}} &= \begin{bmatrix} \frac{\partial \mathbf{u}_1}{\partial \delta \theta_1} & \frac{\partial \mathbf{u}_1}{\partial \delta \theta_2} & \frac{\partial \mathbf{u}_1}{\partial \delta \theta_3} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & -\mathbf{u}_3 & \mathbf{u}_2 \end{bmatrix}_{3 \times 3} \\
\frac{\partial \mathbf{u}_2}{\partial \delta \boldsymbol{\theta}} &= \begin{bmatrix} \frac{\partial \mathbf{u}_2}{\partial \delta \theta_1} & \frac{\partial \mathbf{u}_2}{\partial \delta \theta_2} & \frac{\partial \mathbf{u}_2}{\partial \delta \theta_3} \end{bmatrix} = \begin{bmatrix} \mathbf{u}_3 & \mathbf{0} & -\mathbf{u}_1 \end{bmatrix}_{3 \times 3}
\end{aligned} \tag{26}$$

w_1, w_2 对 $\delta \phi$ 的雅克比：

$$\frac{\partial \mathbf{W}'}{\partial \delta \phi} = \frac{\frac{\partial \mathbf{W}}{\partial \delta \phi}}{\partial \delta \phi} = \mathbf{W} \begin{bmatrix} 0 & -\delta \phi \\ \delta \phi & 0 \end{bmatrix} = \mathbf{W} \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} -w_2 & -w_1 \\ w_1 & -w_2 \end{bmatrix} \tag{27}$$

有

$$\begin{aligned}
\frac{\partial w_1}{\partial \delta \phi} &= -w_2 \\
\frac{\partial w_2}{\partial \delta \phi} &= w_1
\end{aligned} \tag{28}$$

上面的求导过程相对来讲是比较通用的求导方式，当然可以直接把 $\mathbf{u}'_1, \mathbf{u}'_2, w'_1, w'_2$ 他们的表达式从矩阵 \mathbf{U}', \mathbf{W}' 的近似公式中抠出来，然后再计算雅克比，那样更直观。利用上面计算的结果，就很容易得到：

$$\frac{\partial \mathcal{O}}{\partial \begin{bmatrix} \delta \boldsymbol{\theta} \\ \delta \phi \end{bmatrix}} = \begin{bmatrix} \frac{\partial \mathbf{u}_1}{\partial \delta \boldsymbol{\theta}} & \frac{\partial \mathbf{u}_1}{\partial \delta \phi} \\ \frac{\partial \mathbf{u}_2}{\partial \delta \boldsymbol{\theta}} & \frac{\partial \mathbf{u}_2}{\partial \delta \phi} \\ \frac{\partial w_1}{\partial \delta \boldsymbol{\theta}} & \frac{\partial w_1}{\partial \delta \phi} \\ \frac{\partial w_2}{\partial \delta \boldsymbol{\theta}} & \frac{\partial w_2}{\partial \delta \phi} \end{bmatrix}_{8 \times 4} = \begin{bmatrix} \mathbf{0} & -\mathbf{u}_3 & \mathbf{u}_2 & \mathbf{0} \\ \mathbf{u}_3 & \mathbf{0} & -\mathbf{u}_1 & \mathbf{0} \\ 0 & 0 & 0 & -w_2 \\ 0 & 0 & 0 & w_1 \end{bmatrix}_{8 \times 4} \tag{29}$$

$$\begin{aligned}
\frac{\partial \mathcal{L}_w}{\partial \mathcal{O}} \frac{\partial \mathcal{O}}{\partial \begin{bmatrix} \delta \boldsymbol{\theta} \\ \delta \phi \end{bmatrix}} &= \begin{bmatrix} w_1 \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{u}_1 & \mathbf{0}_{3 \times 1} \\ \mathbf{0}_{3 \times 3} & w_2 \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 1} & \mathbf{u}_2 \end{bmatrix}_{6 \times 8} \begin{bmatrix} \mathbf{0} & -\mathbf{u}_3 & \mathbf{u}_2 & \mathbf{0} \\ \mathbf{u}_3 & \mathbf{0} & -\mathbf{u}_1 & \mathbf{0} \\ 0 & 0 & 0 & -w_2 \\ 0 & 0 & 0 & w_1 \end{bmatrix}_{8 \times 4} \\
&= \begin{bmatrix} \mathbf{0} & -w_1 \mathbf{u}_3 & w_1 \mathbf{u}_2 & -w_2 \mathbf{u}_1 \\ w_2 \mathbf{u}_3 & \mathbf{0} & -w_2 \mathbf{u}_1 & w_1 \mathbf{u}_2 \end{bmatrix}_{6 \times 4}
\end{aligned} \tag{30}$$

2.2.2 重投影误差对位姿状态量的雅克比

$$\frac{\partial \mathbf{e}_l}{\partial \mathbf{l}} \frac{\partial \mathbf{l}}{\partial \mathcal{L}_c} \frac{\partial \mathcal{L}_c}{\partial \begin{bmatrix} \delta \boldsymbol{\theta} \\ \delta \mathbf{t} \end{bmatrix}} \tag{31}$$

其中 $\boldsymbol{\theta}, \mathbf{t}$ 表示外参数的最小表示，具体对应外参数 $\mathbf{q}_{bc}, \mathbf{t}_{bc}$ 或者imu机体坐标系 $\mathbf{q}_{wb}, \mathbf{t}_{wb}$ 的增量。上面的链式求导中，只有最后一项还没有推导过，最后一项涉及到公式为：

$$\mathcal{L}_c = \mathcal{T}_{bc}^{-1} \mathcal{T}_{wb}^{-1} \mathcal{L}_w \tag{32}$$

在计算雅克比前，需要定义变量的更新方式：

$$\begin{aligned}
\mathbf{q}' &= \mathbf{q} \otimes \begin{bmatrix} 1 \\ \frac{1}{2} \delta \boldsymbol{\theta} \end{bmatrix} \leftrightarrow \mathbf{R}' = \mathbf{R} \exp([\delta \boldsymbol{\theta}]_{\times}) \\
\mathbf{t}' &= \mathbf{t} + \delta \mathbf{t}
\end{aligned} \tag{33}$$

特别注意，上面的更新方式是沈老师VINS代码采用的形式。实际上，位姿的更新方式还有很多种，比如李代数里的左乘和右乘：

$$\begin{aligned}
\mathbf{T}'_{cw} &= \exp([\boldsymbol{\xi}_{c'c}]_{\times}) \mathbf{T}_{cw} \\
\mathbf{T}'_{wc} &= \mathbf{T}_{wc} \exp([\boldsymbol{\xi}_{cc'}]_{\times})
\end{aligned} \tag{34}$$

而SVO2以及他的预积分的代码中采用的右乘更新的形式，把右乘更新的矩阵展开，容易得到：

$$\begin{aligned}
\mathbf{R}'_{wc} &= \mathbf{R}_{wc} \exp([\delta \boldsymbol{\theta}_{cc'}]_{\times}) \\
\mathbf{t}'_{wc} &= \mathbf{t}_{wc} + \mathbf{R}_{wc} \delta \mathbf{t}_{cc'}
\end{aligned} \tag{35}$$

现在回到雅克比的计算，首先推导重投影误差对外参数的雅克比：

$$\begin{aligned}
\mathcal{L}_c &= \mathcal{T}_{bc}^{-1} \mathcal{T}_{wb}^{-1} \mathcal{L}_w \\
&= \mathcal{T}_{bc}^{-1} \mathcal{L}_b \\
&= \begin{bmatrix} \mathbf{R}_{bc}^T & -\mathbf{R}_{bc}^T [\mathbf{t}_{bc}]_{\times} \\ \mathbf{0} & \mathbf{R}_{bc}^T \end{bmatrix} \mathcal{L}_b \\
&= \begin{bmatrix} \mathbf{R}_{bc}^T (\mathbf{n}_b + [\mathbf{d}_b]_{\times} \mathbf{t}_{bc}) \\ \mathbf{R}_{bc}^T \mathbf{d}_b \end{bmatrix}_{6 \times 1}
\end{aligned} \tag{36}$$

对外参旋转的雅克比：

$$\begin{aligned}
\frac{\partial \mathcal{L}_c}{\partial \delta \boldsymbol{\theta}_{cc'}} &= \begin{bmatrix} \frac{\partial (\mathbf{I} - [\delta \boldsymbol{\theta}_{cc'}]_{\times}) \mathbf{R}_{bc}^T (\mathbf{n}_b + [\mathbf{d}_b]_{\times} \mathbf{t}_{bc})}{\frac{\partial \delta \boldsymbol{\theta}_{cc'}}{\partial \delta \boldsymbol{\theta}_{cc'}}}} \\ \frac{\partial (\mathbf{I} - [\delta \boldsymbol{\theta}_{cc'}]_{\times}) \mathbf{R}_{bc}^T \mathbf{d}_b}{\frac{\partial \delta \boldsymbol{\theta}_{cc'}}{\partial \delta \boldsymbol{\theta}_{cc'}}}} \end{bmatrix} \\
&= \begin{bmatrix} [\mathbf{R}_{bc}^T (\mathbf{n}_b + [\mathbf{d}_b]_{\times} \mathbf{t}_{bc})]_{\times} \\ [\mathbf{R}_{bc}^T \mathbf{d}_b]_{\times} \end{bmatrix}_{6 \times 3}
\end{aligned} \tag{37}$$

对外参平移的雅克比：

$$\begin{aligned}
\frac{\partial \mathcal{L}_c}{\partial \delta \mathbf{t}_{cc'}} &= \begin{bmatrix} \frac{\partial \mathbf{R}_{bc}^T (\mathbf{n}_b + [\mathbf{d}_b]_{\times} (\mathbf{t}_{bc} + \delta \mathbf{t}_{cc'}))}{\frac{\partial \delta \mathbf{t}_{cc'}}{\partial \delta \mathbf{t}_{cc'}}}} \\ \frac{\partial \mathbf{R}_{bc}^T \mathbf{d}_b}{\frac{\partial \delta \mathbf{t}_{cc'}}{\partial \delta \mathbf{t}_{cc'}}}} \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{R}_{bc}^T [\mathbf{d}_b]_{\times} \\ \mathbf{0}_{3 \times 3} \end{bmatrix}_{6 \times 3}
\end{aligned} \tag{38}$$

同理，可以很容易推导出，重投影误差对imu位姿的导数

$$\begin{aligned}
\frac{\partial \mathcal{L}_c}{\partial \delta \boldsymbol{\theta}_{bb'}} &= \mathcal{T}_{bc}^{-1} \begin{bmatrix} [\mathbf{R}_{wb}^T (\mathbf{n}_w + [\mathbf{d}_w]_{\times} \mathbf{t}_{wb})]_{\times} \\ [\mathbf{R}_{wb}^T \mathbf{d}_w]_{\times} \end{bmatrix}_{6 \times 3} \\
\frac{\partial \mathcal{L}_c}{\partial \delta \mathbf{t}_{bb'}} &= \mathcal{T}_{bc}^{-1} \begin{bmatrix} \mathbf{R}_{wb}^T [\mathbf{d}_w]_{\times} \\ \mathbf{0}_{3 \times 3} \end{bmatrix}_{6 \times 3}
\end{aligned} \tag{39}$$

至此所有的雅克比矩阵已经推导完毕。

References