VIO-hw2-提升作业

`VIO`

Spline Fusion: A continuous-time representation for visual-inertial fusion with application to rolling shutter cameras

之前程序里所做的欧拉积分和中值积分，本质都是对数据的近似来进行积分。欧拉积分是一种矩形近似，而中值积分是一阶近似，显然中值积分从理论上来说要优于欧拉积分（作业1的编程也验证了这一点。）
但是再如何近似，这些数据也是离散的，因此本文使用B样条插值拟合曲线，这样即可在小范围内将离散数据变成连续数据。
本文的另一个应用是通过最小二乘法标定不同步的多传感器之间的内参和外参。

2.Continuous-time representation

在李群空间下进行平滑

2.1.Camera pose transformations

$$\mathbf{T}_{b,a}=\begin{bmatrix}\mathbf{R}_{b,a} & \mathbf{a}_b \\ \mathbf{0}^T & 1\end{bmatrix},\quad \mathbf{T}_{b,a}\in\mathbb{SE}\mathbf{3},\quad \mathbf{R}_{b,a}\in\mathbb{SO}\mathbf{3}$$

> 两两帧图像之间的T矩阵。没什么写的。从a帧变换到b帧。
> 左上角是旋转矩阵，右上角是平移量，和正常的Transform矩阵定义一样。
> 其中需要注意的是$\Omega=\frac{1}{\Delta t}\log\left(\mathbf{T}_{b,a}\right),\Omega\in\mathbb{R}^{4\times4}$，这里用矩阵对数的形式，来定义在a，b两帧图像之间$\mathbf{T}_{b,a}$代表的变换在Δt时刻内的变化，用来描述角度和线速度？？？

2.2 C^2 -continuous curves in SE3

选择使用李代数形成的累积基函数来参数化连续轨迹

2.2.1 Representing B-Splines with cumulative basis functions

$$\mathbf{p}(t)=\sum_{i=0}^{n}\mathbf{p}_i B_{i,k}(t)$$

> 这里提到了`B样条基函数`，个人理解其实和矩阵里的基的概念理解起来是一样的。这里使用一个$f(x)=\mathbf{C}\begin{bmatrix}1\\x\\x^2\\x^3\end{bmatrix}$的三次方程来拟合曲线。其中参数`C`是未知的，自然带入四个点$P(t)$即可构建方程，求出经过四个点的三次函数，那么点与点之间的连接被一个平滑的曲线连接。这种基确定，但是前面的参数未知的形式，和矩阵的基类似，因此基函数的概念也就好理解了。

$$\mathbf{p}(t)=\mathbf{p}_0\tilde{B}_{0,k}(t)+\sum_{i=1}^{n}(\mathbf{p}_i-\mathbf{p}_{i-1})\tilde{B}_{i,k}(t)$$

> 这里是Δt对应的T之间的变换的叠加，这里使用前后p点之间的位置差×对应的T变换来进行叠加，相当于是小段小段的位移叠加到一起。整个的累积一个i=1到i=n这么多的数量。

$$\mathbf{T}_{w,s}(t)=\exp\left(\tilde{B}_{0,k}(t)\log\left(\mathbf{T}_{w,0}\right)\right)\prod_{i=1}^{n}\exp\left(\tilde{B}_{i,k}(t)\Omega_i\right)$$

> 这里是整个i=1到i=n的T变换的叠加，只不过使用了李代数的形式。

2.2.2 Cumulative cubic B-Splines

> - 注意这里是`B-Splines(k = 4)`,k=4代表有四个控制点需要带入，因为三次样条有四个未知数。
> - 在$t\in[t_i,t_{i+1})$区间内构建连续的曲线，则使用$[t_{i-1},t_i,t_{i+1},t_{i+2}]$时刻四个点带入求解

$$\tilde{\mathbf{B}}(u)=\mathbf{C}\begin{bmatrix}1\\u\\u^2\\u^3\end{bmatrix},\quad \dot{\tilde{\mathbf{B}}}(u)=\frac{1}{\Delta t}\mathbf{C}\begin{bmatrix}0\\1\\2u\\3u^2\end{bmatrix},\quad \ddot{\tilde{\mathbf{B}}}(u)=\frac{1}{\Delta t^2}\mathbf{C}\begin{bmatrix}0\\0\\2\\6u\end{bmatrix},\quad \mathbf{C}=\frac{1}{6}\begin{bmatrix}6&0&0&0\\5&3&-3&1\\1&3&3&-2\\0&0&0&1\end{bmatrix}$$

> 上面的简述表达的是一维情况下曲线的拟合，但是我们现在是对$SE(3)$，也即是一个四维矩阵进行曲线拟合，所以C变成了一个4*4的矩阵形式

$$\mathbf{T}_{w,s}(u)=\mathbf{T}_{w,i-1}\prod_{j=1}^{3}\exp\left(\tilde{\mathbf{B}}(u)_j\Omega_{i+j}\right)$$

> 这里将几个时刻间的矩阵增量以李代数的形式表示。在`i-1`时刻下，经过三个时间段，到`i+2`时刻代表的位姿，以连乘的形式表示。

$$\dot{\mathbf{T}}_{w,s}(u)=\mathbf{T}_{w,i-1}\left(\dot{\mathbf{A}}_0\mathbf{A}_1\mathbf{A}_2+\mathbf{A}_0\dot{\mathbf{A}}_1\mathbf{A}_2+\mathbf{A}_0\mathbf{A}_1\dot{\mathbf{k}}_2\right)$$

> 这个地方，相比于上一个等式，进行一阶求导，前面的$T_{w,i-1}$不变，对后面的连乘积进行一阶展开。

$$\ddot{\mathbf{T}}_{w,s}(u)=\mathbf{T}_{w,i-1}\begin{pmatrix}\ddot{\mathbf{A}}_0\mathbf{A}_1\mathbf{A}_2+\mathbf{A}_0\ddot{\mathbf{A}}_1\mathbf{A}_2+\mathbf{A}_0\mathbf{A}_1\ddot{\mathbf{A}}_2+\\2\left(\dot{\mathbf{A}}_0\dot{\mathbf{A}}_1\mathbf{A}_2+\dot{\mathbf{A}}_0\mathbf{A}_1\dot{\mathbf{A}}_2+\mathbf{A}_0\dot{\mathbf{A}}_1\dot{\mathbf{A}}_2\right)\end{pmatrix}$$

> 同理，这个地方是二阶展开。

$$\mathbf{A}_j=\exp\left(\Omega_{i+j}\tilde{\mathbf{B}}(u)_j\right),\quad \dot{\mathbf{A}}_j=\mathbf{A}_j\Omega_{i+j}\hat{\mathbf{B}}(u)$$

$$\ddot{\mathbf{A}}_j=\mathbf{A}_j\Omega_{i+j}\dot{\tilde{\mathbf{B}}}(u)_j+\mathbf{A}_j\Omega_{i+j}\ddot{\mathbf{B}}(u)_j$$

> 这个地方是对上述展开式的补充，为了表达简洁，一些地方用了简写，详细形式在这里进行了说明。

3.Generative model of visual-inertial data

3.1 Parameterization

$$\mathbf{p}_b=\mathcal{W}(\mathbf{p}_a;\mathbf{T}_{b,a},\rho)=\pi\left([\mathbf{K}_b|\mathbf{0}]\mathbf{T}_{b,a}\left[\mathbf{K}_a^{-1}\begin{bmatrix}\mathbf{p}_a\\1\end{bmatrix};\rho\right]\right)$$

> 这个公式表达的含义是，在相机的a帧观察到的p点，结合深度信息，经过相机的内参，和a，b两帧之间的变换矩阵，能够投影到b帧的画面上。($\pi(*)$是投影函数，在slam相关论文中经常出现)

$$\mathrm{Gyro}(u)=\mathbf{R}_{w,s}^{\top}(u)\cdot\dot{\mathbf{R}}_{w,s}(u)+bias$$

$$\mathrm{Accel}(u)=\mathbf{R}_{w,s}^{\top}(u)\cdot(\ddot{\mathbf{s}}_w(u)+g_w)+bias$$

> 这个地方就是作业里面的欧拉积分那一部分了，分别利用陀螺仪和加速度计进行积分，用获取的结果来进行修正。

3.2 Minimization

> 构建优化问题的模型---即对位置，速度和加速度三个部分加在一起构建目标函数。文中提到以信息矩阵来决定三个部分的权重，也即$\boldsymbol{\Sigma}$，在《SLAM14讲》中也有提到，实际是将最小二乘问题构建的二范数，利用协方差矩阵，将欧氏距离变成了马氏距离。文中也有说明。

$$E(\boldsymbol{\theta})=\sum_{\mathbf{p}_m}\left(\hat{\mathbf{p}}_m-\mathcal{W}\left(\mathbf{p}_r;\mathbf{T}_{c,s}\mathbf{T}_{w,s}(u_m)^{-1}\mathbf{T}_{w,s}(u_r)\mathbf{T}_{s,c},\rho\right)\right)_{\Sigma_p}^2+\sum_{\hat{\omega}_m}(\hat{\omega}_m-\mathrm{Gyro}(u_m))_{\Sigma_\omega}^2+\sum_{\hat{\mathbf{a}}_m}(\hat{\mathbf{a}}_m-\mathrm{Accel}(u_m))_{\Sigma_{\mathbf{a}}}^2$$

> 优化信息的长度，可以是全部数据，也可以是以滑动窗口的形式。

4.Projection into a rolling shutter camera

> 这里考虑的是滚动快门相机和全局快门相机的问题，二者的区别网络可查。在考虑滚动快门相机模型的时候，因为像素值是逐行读取，因此随着相机的运动，呈像的整个画面会出现失真。如何将运动信息考虑在内，将失真的图像结合运动估计，恢复成真实图像，是本段主要介绍的部分。

$$\mathbf{p}_b(t)=\begin{bmatrix}x_b(t)\\y_b(t)\end{bmatrix}=\mathcal{W}\left(\mathbf{p}_a;\mathbf{T}_{b,a}(t),\boldsymbol{\rho}\right)$$

> 这个公式前面提过，投影函数，将三维坐标经过T变换到另一帧图像的uv坐标系下。

$$\mathbf{p}_b(t+\Delta t)=\mathcal{W}\left(\mathbf{p}_a;\mathbf{T}_{b,a}(t),\rho\right)+\Delta t\,\frac{d\mathcal{W}\left(\mathbf{p}_a;\mathbf{T}_{b,a}(t),\rho\right)}{dt}$$

$$y_{b(t+\Delta)}=\frac{h(t+\Delta t-s)}{e-s},\quad \Delta t=-\frac{h\cdot t_0+s.\,(y_b(t)-h)-e\cdot y_b(t)}{(s-e)\,\frac{d\mathcal{W}_y\left(\mathbf{p}_a;\mathbf{T}_{b,a}(t),\rho\right)}{dt}+h}$$

> 这里结合P在图像画面里面的运动估计，来对滚动快门相机进行补偿，图像里点坐标的运动为$y_{b(t)} = h(t-s)$，$/(e-s)$，其中p在图像里面的运动估计，需要在结合其在图像坐标中的位置对时间的求导(也即在图像里面运动的速度)，再乘Δt，得到一个微小的位移，再进行估计补偿。

5.Experiments

不作分析

5.1 Simulated rolling shutter visual odometry

不作分析

5.2 Self-calibration and scale

不作分析

6.Conclusions and future work

不作分析