

# 基于卷帘快门 RGB-D 相机的视觉惯性 SLAM 方法

曹力科, 肖晓晖

(武汉大学动力与机械学院, 湖北 武汉 430072)

**摘要:** 针对当前单目视觉惯性 SLAM (同步定位与地图创建) 中初始化需要加速度激励以及高 IMU (惯性测量单元) 噪声条件下系统精度下降的问题, 提出一种基于卷帘快门 RGB-D 相机的视觉惯性 SLAM 方法——VINS-RSD 方法. VINS-RSD 方法联合卷帘快门 RGB-D 图像和 IMU 对系统进行初始化, 通过控制特征的速度对卷帘快门效应进行校正, 并采用一种带置信因子的损失核函数进行滑动窗口优化. 为了评测该方法, 在 WHU-RSVI 数据集的基础上制作了一个可以评价 RGB-D 视觉惯性 SLAM 算法的开源深度数据集并进行实验验证. 结果表明, 与 VINS-Mono 方法相比, VINS-RSD 方法的均方根误差平均值下降了 30.76%, 表明该方法能获得更高的跟踪精度.

**关键词:** 同步定位与地图创建; 视觉惯性; 卷帘快门; RGB-D

中图分类号: TP242

文献标识码: A

文章编号: 1002-0446(2021)-02-0193-10

## A Visual-Inertial SLAM Method Based on Rolling Shutter RGB-D Cameras

CAO Like, XIAO Xiaohui

(School of Power and Mechanical Engineering, Wuhan University, Wuhan 430072, China)

**Abstract:** For the current monocular visual-inertial SLAM (simultaneous localization and mapping) systems, the acceleration excitation is necessary, and the system accuracy will decrease in the case of high IMU (inertial measurement unit) noises. To solve those problems, a visual-inertial SLAM method based on rolling shutter RGB-D cameras is proposed, named VINS-RSD method, which combines rolling shutter RGB-D image and IMU to initialize the system. The rolling shutter effect is corrected by controlling the feature velocity, and a loss kernel function with confidence factor is applied to the sliding window optimization. An open-source depth dataset extended from WHU-RSVI dataset is developed to evaluate the RGB-D visual-inertial SLAM method. Experiments are performed on the dataset and the root mean square error of VINS-RSD method is reduced by 30.76% compared to VINS-Mono (monocular visual-inertial system) method, which demonstrates that the proposed method can achieve a higher tracking accuracy.

**Keywords:** simultaneous localization and mapping (SLAM); visual-inertial; rolling shutter; RGB-D

## 1 引言 (Introduction)

同步定位与地图创建 (SLAM) 是移动机器人、微型飞行器、虚拟现实、增强现实等技术需要的基础模块. 多种传感器都能实现 SLAM 方法, 但视觉和惯性测量单元 (IMU) 融合的 SLAM 可以在全球定位系统 (GPS) 缺失的环境中使用, 且鲁棒性好、成本低. 故近年来视觉惯性 SLAM 方法得到了大量的研究和应用. 相机按照扫描机制不同可分为全局快门相机和卷帘快门相机. 采用全局快门相机时同一幅图像中所有像素在同一时刻曝光, 而卷帘快门相机采用逐行扫描 (或逐列扫描) 的方式, 使得不同行或列上的像素曝光时间不同, 因此在相机运动的时候图像会发生扭曲, 这种效应称为卷帘快门效应 (果冻效应). 卷帘快门机制以其低廉的成本,

在消费级相机领域得到了广泛的使用.

视觉和 IMU 融合的 SLAM 方法可分为基于滤波和基于优化 2 种方式. 基于滤波的经典工作有 MSCKF 方法<sup>[1-2]</sup>和 ROVIO 方法<sup>[3]</sup>等. 基于优化的经典工作有 OKVIS 方法<sup>[4-5]</sup>、VINS-Mono 方法<sup>[6]</sup>和 VINS-Fusion 方法<sup>[7-9]</sup>等. 慕尼黑工业大学的 Kerl 等<sup>[10]</sup>利用样条建模提出了一种基于卷帘快门 RGB-D 相机的 SLAM 方法. Lovegrove 等<sup>[11-12]</sup>提出了一种基于样条融合的视觉惯性 SLAM 方法, 利用样条曲线处理卷帘快门效应. 另外 Lee 等<sup>[13]</sup>利用 IMU 测量得到的运动信息辅助卷帘快门相机进行位姿估计, Li 等<sup>[14]</sup>则结合卷帘快门相机和 IMU 提出了一种在连续位姿之间插值的方法, 并建立了一种特征测量误差模型来解决时延问题.

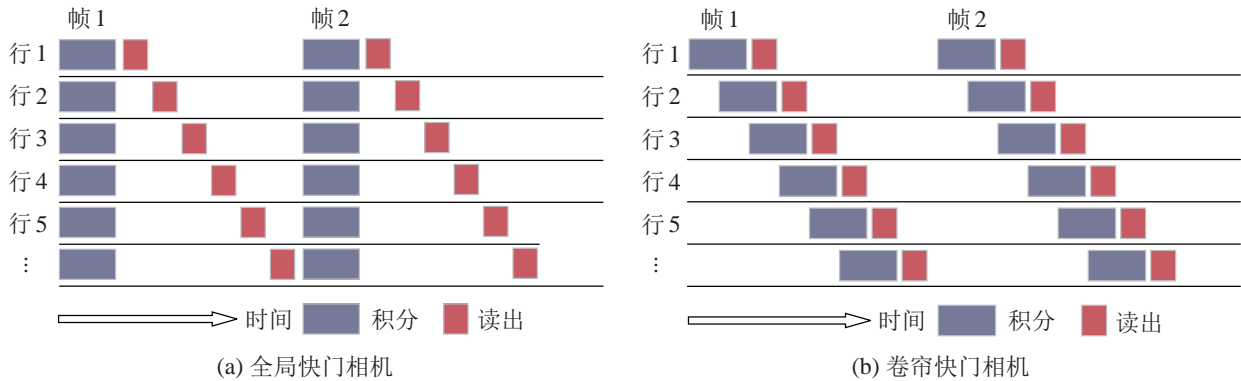


图1 全局快门相机和卷帘快门相机原理

Fig.1 The principle of the global shutter camera and the rolling shutter camera

融合全局快门 RGB 相机和 IMU 能得到较高的鲁棒性和定位精度, 然而卷帘快门效应仍然是主流视觉 SLAM 相关方法中的一个挑战<sup>[15]</sup>. 仅仅使用 RGB 相机很难得到稠密的地图, 另外单目视觉惯性 SLAM 中的初始化方法也比较复杂, 需要视觉与 IMU 联合才能完成初始化, 且需要在特定的运动条件下完成.

为了更好地处理上述问题, 本文提出一种基于卷帘快门 RGB-D 相机的视觉惯性 SLAM 方法, 由深度相机直接获得深度, 解决了视觉惯性 SLAM 需要在特定运动状态下完成初始化的问题, 而根据相机的深度图像, 可以直接创建稠密地图.

本文的贡献如下:

1) 扩展了武汉大学的卷帘快门视觉惯性数据集 WHU-RSVI (Wuhan University rolling shutter visual inertial), 对数据集中的 RGB 图像都生成了合成的深度数据.

2) 提出了一种基于卷帘快门 RGB-D 相机的视觉惯性 SLAM 方法.

本文开源的深度数据集下载地址为 <http://aric.whu.edu.cn/whu-rsvi-depth.html>.

## 2 WHU-RSVI 深度数据集 (WHU-RSVI depth dataset)

### 2.1 卷帘快门相机原理

所有使用电荷耦合器件 (charge-coupled device, CCD) 图像传感器的相机皆为全局快门相机, 在全局快门相机的图像中, 所有像素在同一时间进行曝光. 而使用互补金属氧化物半导体 (complementary metal oxide semiconductor, CMOS) 图像传感器的相机既有全局快门相机, 也有卷帘快门相机. 在消费级相机中, 绝大部分都使用 CMOS 卷帘快门相

机. 卷帘快门相机采用一种逐行 (也可以是列) 的扫描模式, 即不同行的像素曝光时间是不一致的, 其原理如图 1 所示.

### 2.2 深度图像

本文深度数据集是在 WHU-RSVI<sup>[16]</sup> 数据集的基础上扩展而来. 与 WHU-RSVI 数据集相同, 模型的尺寸是 800 cm×500 cm×250 cm, 图像的分辨率为 640×480, 相机的视场角为 90°. WHU-RSVI 数据集中的 RGB 图像是采用光线追踪软件 POV-Ray<sup>†</sup> 渲染得到的, 相应的深度图像可以采用 POV-Ray 软件中的雾介质得到<sup>[17]</sup>, 但实现起来比较复杂. 因此, 本文数据集中的深度图像是直接采用 POV-Ray 软件的扩展软件 MegaPOV<sup>‡</sup> 的后处理模块渲染得到的. 深度图与 RGB 图有相同的内参矩阵, 其右手坐标系下的内参矩阵为

$$\mathbf{K} = \begin{bmatrix} 320.0 & 0.0 & 319.5 \\ 0.0 & 320.0 & 239.5 \\ 0.0 & 0.0 & 1.0 \end{bmatrix} \quad (1)$$

与 TUM RGB-D 数据集<sup>[18]</sup> 一致, 本文深度图像为 16 位灰度 png 格式, 深度因子为 5 000, 即像素值 5 000 代表的距离为 1 m, 16 位灰度图最大像素值为 65 535, 所代表的最大距离为 13.107 m.

在主流商业 RGB-D 相机中, 图像深度是指相机坐标系下 Z 轴方向的距离<sup>[19-20]</sup>, 而非相机到光心的实际距离. 设  $\mathbf{P}_c = (X, Y, Z)^T$  是相机坐标系下的空间点的坐标, 根据针孔相机的投影模型有:

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \frac{1}{Z} \mathbf{K} \mathbf{P}_c = \frac{1}{Z} \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \quad (2)$$

<sup>†</sup><http://www.povray.org/>.

<sup>‡</sup><http://megapov.inetart.net/>.



图 2 数据集中的 RGB 图像和对应的深度图像概览

Fig.2 Overview of RGB images and corresponding depth images in the dataset

则在相机归一化坐标下:

$$\begin{cases} x = \frac{X}{Z} = \frac{1}{f_x}(u - c_x) \\ y = \frac{Y}{Z} = \frac{1}{f_y}(v - c_y) \end{cases} \quad (3)$$

则  $Z$  轴方向的深度与空间点到相机光心距离存在如下映射关系:

$$Z = \frac{d}{\sqrt{1+x^2+y^2}} \quad (4)$$

利用式 (4) 对渲染后的深度图像进行校正后, 可得到与主流 RGB-D 相机图像一致的格式. 本数据集中的深度图包含 6 个轨迹序列, 每个轨迹包含全局快门和卷帘快门 2 种模式. 研究人员结合 WHU-RSVI 数据集可以评测 RGB-D 视觉 SLAM 和 RGB-D 视觉惯性 SLAM 等相关的算法, 并对比全局快门图像序列和卷帘快门图像序列在各算法上的表现.

### 3 VINS-RSD 方法 (VINS-RSD method)

本文在 VINS-Mono 方法<sup>[6]</sup>的基础上提出了 VINS-RSD 方法. VINS-RSD 方法的测量预处理、系统初始化以及紧耦合视觉惯性里程计模块与 VINS-Mono 方法不同, 但在重定位模块和全局姿态图优化上是一致的.

本文用  $(\cdot)^w$  表示世界坐标系, 重力的方向与世界坐标系中的  $Z$  轴对齐.  $(\cdot)^b$  表示载体坐标系, IMU 坐标系与载体坐标系重合.  $(\cdot)^c$  表示相机坐标系, 特别地,  $b_k$  是第  $k$  帧图像的载体坐标系,  $c_k$  是第  $k$  帧图像的相机坐标系. 旋转矩阵  $\mathbf{R}$  和四元数  $\mathbf{q}$  皆用来表示旋转.  $\mathbf{q}_b^w$  和  $\mathbf{p}_b^w$  表示载体坐标系到世界坐标系的旋转和平移.  $\otimes$  表示 2 个四元数之间的乘法运算.  $\mathbf{g}^w = [0, 0, \mathbf{g}]^T$  为世界坐标系下的重力向量.  $(\cdot)$  表示某个有噪声的测量值或者估计值.

#### 3.1 数据预处理

与全局快门相机不同的是, 在测量值中, 还需要记录特征的速度以及特征像素的扫描行值. VINS-Mono 方法对卷帘快门相机与 IMU 进行时间同步误差的估计. 在现有技术下, 通过硬件同步采样技术, 可以实现图像帧和 IMU 帧的采样起始时间的同步化. 故在 VINS-RSD 方法中, 卷帘快门相机和 IMU 之间的时间被认为是同步的, 不需要估计. 数据集中卷帘快门相机为行扫描相机, 第 1 行像素的采样时间和 IMU 起始采样时间一致. 卷帘快门相机的读出时间为  $t_r$ , 假设图像帧的时间戳为图像中心行所在的时间, 则有:

$$t_c = t_i + \frac{1}{2}t_r \quad (5)$$

其中  $t_c$  为卷帘快门相机校正后的虚拟曝光时间,  $t_i$

为 IMU 在图像第 1 行曝光时的采样时间。

在相机坐标系下, 特征点在深度为 1 的相机平面上的运动速度为  $[v_{f_x}, v_{f_y}]^T$ , 假设在同一帧内特征点的运动都是线性的, 特征点在经过校正以后, 在图像中心行扫描时刻的位置  $\mathbf{p}_g$  为

$$\mathbf{p}_g = \mathbf{p}_i - \left(r_i - \frac{r_L}{2}\right) \frac{t_r}{r_L} \mathbf{v}_f \quad (6)$$

其中,  $r_i$  为当前像素所在的行数,  $r_L$  为扫描总线数 (图像总行数),  $\mathbf{p}_i$  为当前特征在卷帘快门图像实际成像时的位置. 此时, 卷帘快门相机被校正为全局快门相机.

### 3.2 IMU 预积分

本文采用文 [6, 21] 的方法对 IMU 进行预积分. 通常情况下, IMU 的测量值包含载体坐标系下的三轴加速度计和三轴陀螺仪 (角速度).  $t$  时刻的测量模型如下:

$$\begin{aligned} \hat{\mathbf{a}}_t &= \mathbf{a}_t + \mathbf{b}_{a_t} + \mathbf{R}_w^t \mathbf{g}^w + \mathbf{n}_a \\ \hat{\boldsymbol{\omega}}_t &= \boldsymbol{\omega}_t + \mathbf{b}_{\omega_t} + \mathbf{n}_{\omega} \end{aligned} \quad (7)$$

其中  $\hat{\mathbf{a}}_t$  表示加速度计的测量值,  $\hat{\boldsymbol{\omega}}_t$  表示陀螺仪的测量值,  $\mathbf{a}_t$  表示加速度计的真值,  $\mathbf{n}_a$  表示加速度计的噪声,  $\boldsymbol{\omega}_t$  表示陀螺仪的真值,  $\mathbf{n}_{\omega}$  表示陀螺仪的噪声.  $\mathbf{b}_{a_t}$  表示加速度计的偏置,  $\mathbf{b}_{\omega_t}$  表示陀螺仪的偏置, 加速度计和陀螺仪的噪声可以简化为高斯白噪声:

$$\mathbf{n}_a \sim N(0, \sigma_a^2), \quad \mathbf{n}_{\omega} \sim N(0, \sigma_{\omega}^2) \quad (8)$$

加速度计和陀螺仪的偏置可以简化为一个离散的随机游走模型, 其对应的连续模型称为维纳过程或布朗运动, 这个过程是由  $\sigma_{b_a}$  和  $\sigma_{b_{\omega}}$  强度的噪声积分组成:

$$\dot{\mathbf{b}}_{a_t} \sim N(0, \sigma_{b_a}^2), \quad \dot{\mathbf{b}}_{\omega_t} \sim N(0, \sigma_{b_{\omega}}^2) \quad (9)$$

在 2 个连续帧  $b_k$  和  $b_{k+1}$  之间存在数个 IMU 数据, 数据之间的时间间隔是  $[t_k, t_{k+1}]$ , 在偏置已知的情况下进行积分, 则位移  $\boldsymbol{\alpha}_{b_{k+1}}^{b_k}$ 、速度  $\boldsymbol{\beta}_{b_{k+1}}^{b_k}$  和旋转四元数  $\boldsymbol{\gamma}_{b_{k+1}}^{b_k}$  可表示为

$$\begin{aligned} \boldsymbol{\alpha}_{b_{k+1}}^{b_k} &= \iint_{t \in [t_k, t_{k+1}]} \mathbf{R}_t^{b_k} (\hat{\mathbf{a}}_t - \mathbf{b}_{a_t}) dt^2 \\ \boldsymbol{\beta}_{b_{k+1}}^{b_k} &= \int_{t \in [t_k, t_{k+1}]} \mathbf{R}_t^{b_k} (\hat{\mathbf{a}}_t - \mathbf{b}_{a_t}) dt \\ \boldsymbol{\gamma}_{b_{k+1}}^{b_k} &= \int_{t \in [t_k, t_{k+1}]} \frac{1}{2} \boldsymbol{\Omega} (\hat{\boldsymbol{\omega}}_t - \mathbf{b}_{\omega_t}) \boldsymbol{\gamma}_t^{b_k} dt \end{aligned} \quad (10)$$

其中

$$\boldsymbol{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} [\boldsymbol{\omega}]_{\times} & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^T & 0 \end{bmatrix}, \quad [\boldsymbol{\omega}]_{\times} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \quad (11)$$

### 3.3 初始化

纯单目视觉方法不能提供尺度信息, 因此必须联合惯性测量得到系统的尺度, 如果初始值不理想, 则很难将视觉测量和惯性测量融合到一起. 在 OKVIS 方法中, 假定了初始化开始时系统处于静止状态<sup>[4-5]</sup>, 但现实中很多情况都不满足该假设条件. 在 VIORB 方法<sup>[22]</sup>和 VINS-Mono 方法<sup>[6]</sup>的初始化中也需要加速度激励以测量尺度.

在只有一个相机的时候, 需要在不同图像帧之间采用三角化的方法来求特征点的深度, 在有深度图的基础上, 特征点的深度可以直接由测量值得到. 因此根据 RGB-D 传感器自带的深度信息, 仅依靠视觉测量即可得到系统的尺度. 本文初始化时, 直接采用 PnP 方法<sup>[23-25]</sup>对位姿进行求解. 在求解位姿之前, 需要把卷帘快门相机图像中的特征点位置通过式 (6) 对齐到同一时刻.

在窗口内有  $b_k$  和  $b_{k+1}$  两个连续的 IMU 帧, 通过视觉处理可得到  $\mathbf{q}_{b_k}^{c_0}$  和  $\mathbf{q}_{b_{k+1}}^{c_0}$  的旋转结果, 而通过 IMU 的预积分也可以得到对应的旋转为  $\hat{\boldsymbol{\gamma}}_{b_{k+1}}^{b_k}$ . 在没有误差的情况下, 通过视觉求得的各帧间的旋转与 IMU 预积分求得的各帧之间的旋转是一致的, 根据此关系, 可以构建代价函数来求解陀螺仪的偏置:

$$\min_{\delta \mathbf{b}_{\omega}} \sum_{k \in \mathcal{B}} \left\| (\mathbf{q}_{b_{k+1}}^{c_0})^{-1} \otimes \mathbf{q}_{b_k}^{c_0} \otimes \boldsymbol{\gamma}_{b_{k+1}}^{b_k} \right\|^2 \quad (12)$$

其中  $\mathcal{B}$  表示滑动窗口内的所有帧, 而  $\boldsymbol{\gamma}_{b_{k+1}}^{b_k}$  可 1 阶近似为

$$\boldsymbol{\gamma}_{b_{k+1}}^{b_k} \approx \hat{\boldsymbol{\gamma}}_{b_{k+1}}^{b_k} \otimes \begin{bmatrix} 1 \\ \frac{1}{2} \mathbf{J}_{b_{\omega}}^{\gamma} \delta \mathbf{b}_{\omega} \end{bmatrix} \quad (13)$$

对式 (12) 进行优化得到一个单位四元数的目标值, 式 (13) 中  $\mathbf{J}_{b_{\omega}}^{\gamma}$  为旋转对陀螺仪偏置变化的雅可比矩阵. 通过求取雅可比矩阵构建正定方程后, 即可用最小二乘法求解. 求出陀螺仪的偏置之后, 需要根据求出的偏置重新进行预积分. 之后求解重力加速度和速度, 并对重力进行精细优化.

### 3.4 视觉惯性里程计

在初始化完成后, 进行紧耦合滑动窗口视觉惯性里程计状态估计. 滑动窗口内所有待估计的状态

量为

$$\begin{aligned} \mathbf{X} &= [\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_n, \mathbf{x}_c^b, \lambda_0, \lambda_1, \dots, \lambda_m] \\ \mathbf{x}_k &= [\mathbf{p}_{b_k}^w, \mathbf{v}_{b_k}^w, \mathbf{q}_{b_k}^w, \mathbf{b}_a, \mathbf{b}_g], k \in [0, n] \\ \mathbf{x}_c^b &= [\mathbf{p}_c^b, \mathbf{q}_c^b] \end{aligned} \quad (14)$$

其中  $\mathbf{x}_k$  是捕获第  $k$  帧图像时 IMU 的状态, 包含位置、速度、世界坐标系下 IMU 的朝向、加速度计和陀螺仪的偏置.  $\lambda_l$  是第  $l$  个特征在初次被观测到时的逆深度.

在 VINS-Mono 方法所支持的卷帘快门相机中, 除了估计上述的状态量以外, 还对 IMU 和相机的时间同步误差进行估计. 因相机和 IMU 的时间通过硬件同步采样技术可认为是准确可靠的, 故在 VINS-RSD 方法中省去了对时间同步误差的估计.

在同等视觉水平下, 当 IMU 噪声水平越高时, IMU 测量的置信度越低. 所以根据不同的 IMU 标定信息, 可以选择合适的损失核函数置信因子来调整视觉和 IMU 的相对置信度. 本文定义优化的目标函数为

$$\begin{aligned} \min_{\mathbf{X}} & \left( \|\mathbf{r}_p - \mathbf{H}_p \mathbf{X}\|^2 + \right. \\ & \left. \sum_{k \in \mathcal{B}} \rho_b \left( \left\| r_{\mathcal{B}}(\hat{\mathbf{z}}_{b_{k+1}}^{b_k}, \mathbf{X}) \right\|_{\mathcal{P}_{b_{k+1}}^{b_k}}^2 \right) + \right. \\ & \left. \sum_{(i,j) \in \mathcal{C}} \rho_c \left( \left\| r_{\mathcal{C}}(\hat{\mathbf{z}}_i^{c_j}, \mathbf{X}) \right\|_{\mathcal{P}_i^{c_j}}^2 \right) \right) \end{aligned} \quad (15)$$

其中  $\mathbf{r}_p - \mathbf{H}_p \mathbf{X}$  为边缘化先验信息残差,  $r_{\mathcal{B}}(\hat{\mathbf{z}}_{b_{k+1}}^{b_k}, \mathbf{X})$  为 IMU 残差,  $r_{\mathcal{C}}(\hat{\mathbf{z}}_i^{c_j}, \mathbf{X})$  为视觉残差.  $\rho_b(\cdot)$  和  $\rho_c(\cdot)$  为 IMU 和相机的复合损失核函数. 其中:

$$\begin{cases} \rho_b(s) = a_b s \\ \rho_c(s) = a_c \log(1 + s) \end{cases} \quad (16)$$

$\rho_b(s)$  为比例损失核函数与 2 范数平方和的复合函数, 即在平方和的基础上乘置信因子  $a_b$ . 而  $\rho_c(s)$  为比例损失核函数与 Cauchy 核函数的复合函数, 即在 Cauchy 核函数的基础上乘 IMU 置信因子  $a_c$ . 当 IMU 的噪声水平增大时, 其测量的置信度下降, 即相对来说视觉测量更加可靠, 此时可以将  $a_c$  增大,  $a_b$  减小; 反之可以将  $a_c$  减小,  $a_b$  增大.

本文方法与 VINS-Mono 方法在边缘化先验残差和 IMU 测量残差的处理上是一致的. 在视觉残差处理上, 假设第  $l$  个特征在第  $i$  个图像帧中首次被观测到, 则该特征在第  $j$  个图像帧中的测量残差为

$$r_{\mathcal{C}}(\hat{\mathbf{z}}_i^{c_j}, \mathbf{X}) = [\mathbf{b}_1 \ \mathbf{b}_2]^T \left( \bar{\mathbf{P}}_l^{c_j} - \frac{\mathbf{P}_l^{c_j}}{\|\mathbf{P}_l^{c_j}\|} \right) \quad (17)$$

其中  $\bar{\mathbf{P}}_l^{c_j}$  是第  $l$  个特征在第  $j$  帧中测量到的单位向量.  $\mathbf{P}_l^{c_j}$  是第  $i$  到第  $j$  图像帧的预测平移向量.  $\mathbf{b}_1$  和  $\mathbf{b}_2$  是正切平面上任意 2 个正交基. 而第  $j$  帧的测量值为

$$\bar{\mathbf{P}}_l^{c_j} = \boldsymbol{\pi}_c^{-1} \left( \begin{bmatrix} \hat{u}_l^{c_j} \\ \hat{v}_l^{c_j} \end{bmatrix} - \left( r_i - \frac{r_L}{2} \right) \frac{t_r}{r_L} \begin{bmatrix} v_{f_x} \\ v_{f_y} \end{bmatrix} \right) \quad (18)$$

其中  $[\hat{u}_l^{c_j}, \hat{v}_l^{c_j}]^T$  是图像中的特征观测量,  $\boldsymbol{\pi}_c$  和  $\boldsymbol{\pi}_c^{-1}$  为相机投影函数和反向投影函数.  $\mathbf{p}$  和  $\mathbf{R}$  是系统在坐标系中的平移和旋转.  $r_i$ 、 $r_L$ 、 $v_{f_x}$ 、 $v_{f_y}$ 、 $t_r$  的介绍见第 3.1 节. 将预测值  $\mathbf{P}_l^{c_j}$  加入卷帘快门校正, 则有:

$$\begin{aligned} \mathbf{P}_l^{c_j} &= \mathbf{R}_c^b \left( \mathbf{R}_w^{b_j} \left( \mathbf{R}_{b_i}^w \left( \mathbf{R}_c^b \frac{1}{\lambda_l} \left( \boldsymbol{\pi}_c^{-1} \left( \begin{bmatrix} \hat{u}_l^{c_j} \\ \hat{v}_l^{c_j} \end{bmatrix} - \left( r_i - \frac{r_L}{2} \right) \times \right. \right. \right. \right. \right. \right. \right. \right. \\ & \left. \left. \left. \left. \left. \left. \frac{t_r}{r_L} \begin{bmatrix} v_{f_x} \\ v_{f_y} \end{bmatrix} \right) \right) \right) + \mathbf{p}_c^b \right) + \mathbf{p}_{b_i}^w - \mathbf{p}_{b_j}^w \right) - \mathbf{p}_c^b \end{aligned} \quad (19)$$

其中,  $\lambda_l$  是特征在图像  $i$  中的逆深度. 已知预测值  $\mathbf{P}_l^{c_j}$  和测量值  $\bar{\mathbf{P}}_l^{c_j}$ , 即可通过式 (17) 计算卷帘快门相机的视觉残差. 联合边缘化先验信息残差、IMU 残差以及复合损失函数, 可以得到最终的优化目标函数. 对目标函数进行滑动窗口优化, 可以得到相机和 IMU 的位姿、速度和外参等信息.

### 3.5 稠密点云图

根据深度图像, 可以较容易地建立稠密点云图. 世界坐标系下点云为  $\mathbf{p}_w$ , IMU 载体坐标系相对于世界坐标系的旋转为  $\mathbf{R}_b^w$ , 平移为  $\mathbf{p}_b^w$ , 相机坐标系到 IMU 坐标系的外部旋转为  $\mathbf{R}_c^b$ , 外部平移为  $\mathbf{p}_c^b$ , 则世界坐标系下点云坐标  $\mathbf{p}_w$  为

$$\mathbf{p}_w = \mathbf{R}_b^w \times (\mathbf{R}_c^b \times \mathbf{p}_c + \mathbf{p}_c^b) + \mathbf{p}_b^w \quad (20)$$

采用统计滤波器对点云进行滤波, 稠密建图的效果如图 3 所示.



图 3 稠密点云图

Fig.3 Dense point cloud

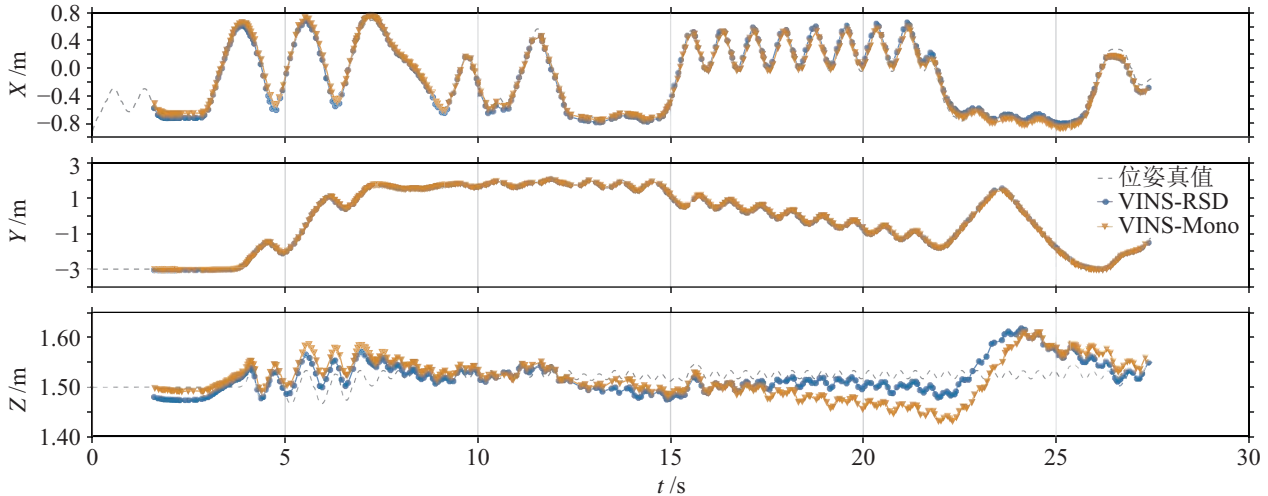


图4 轨迹1快速序列的绝对轨迹误差在 VINS-RSD 和 VINS-Mono 算法下的对比

Fig.4 The absolute trajectory error comparison of the fast sequence of trajectory 1 between VINS-RSD and VINS-Mono algorithms

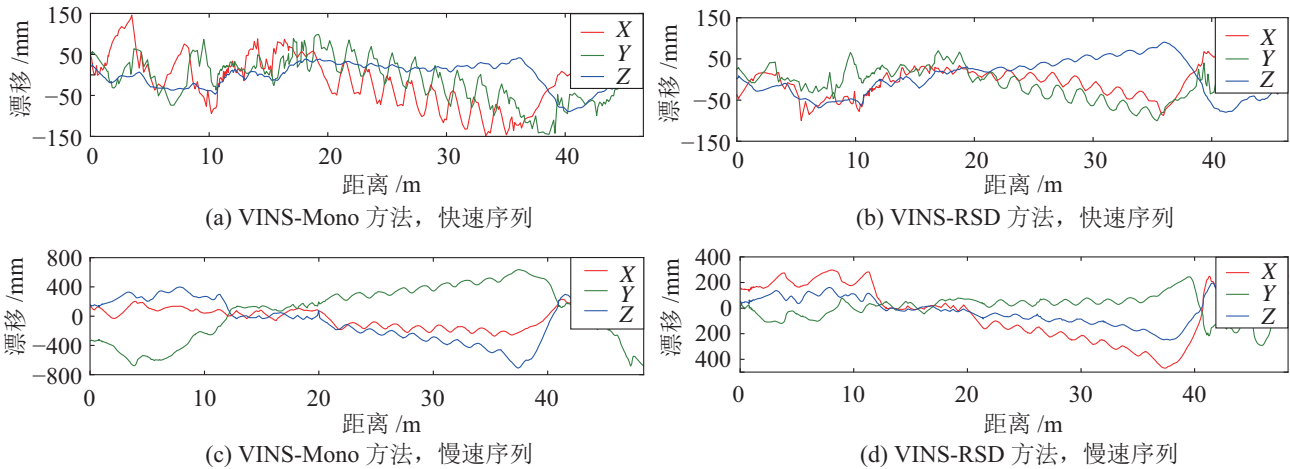


图5 轨迹1在 VINS-Mono 和 VINS-RSD 算法下在 X、Y、Z 轴上的误差

Fig.5 The errors of trajectory 1 in X, Y and Z axes under VINS-Mono and VINS-RSD algorithms

## 4 实验 (Experiment)

实验所用的计算机平台的 CPU 为 Intel Core i7-7700HQ (4 cores @ 2.80 GHz), 内存为 16 GB. 在参数设置中, 加速度噪声为  $\sigma_a = 1.0 \times 10^{-2}$ , 加速度偏置噪声为  $\sigma_{b_a} = 1.5 \times 10^{-2}$ , 陀螺仪噪声为  $\sigma_\omega = 1.0 \times 10^{-3}$ , 陀螺仪偏置噪声为  $\sigma_{b_\omega} = 1.0 \times 10^{-4}$ . 该 IMU 噪声强度是一种相对很高的噪声, 比通常的要高 1 个数量级. IMU 的帧率为 300 Hz, 图像帧率为 30 Hz. 相机损失核函数的置信因子  $a_c = 2.0$ , IMU 损失核函数的置信因子为  $a_b = 1.0$ .

最新的 VINS-Mono 方法提供了对卷帘快门相机的支持, 能得到比全局快门相机更好的估计效果. 因此本文将 VINS-RSD 方法与 VINS-Mono 方法的卷帘快门版本进行对比.

### 4.1 轨迹 1

本文使用 evo 工具<sup>[26]</sup> 和 rpg 轨迹评测工具<sup>[27]</sup>

进行绘图. 图 4 为轨迹 1 的快速序列在 10 次评测中均方根误差最接近评测均值的 2 组数据. 图中直观地展示了 VINS-RSD 方法与 VINS-Mono 方法卷帘快门相机模式的评测结果在 X、Y、Z 三个坐标轴上与轨迹真值的对比. 从图 4 中可以看出, VINS-Mono 方法和 VINS-RSD 方法的估计结果都与轨迹真值比较接近, 因为相机在 X 轴、Y 轴的运动范围比较大, 因此较小的误差在较大的坐标轴范围表现不明显. 但相机在 Z 轴坐标的运动范围比较小, 基本限定在 1.4 m~1.6 m 范围内, 从该轴看, VINS-RSD 方法与黑色虚线的轨迹真值更加贴近, 而 VINS-Mono 方法的漂移显然要更加明显, 从侧面反映出 VINS-RSD 方法比 VINS-Mono 方法具有更高的精度.

在闭环情况下利用轨迹 1 进行实验, VINS-Mono 方法和 VINS-RSD 方法在 X、Y、Z 轴上的

误差详情如图 5 所示. 由于中速序列的效果位于快速和慢速之间, 故本文中省去了对其绘图. 由图 5 可知, VINS-Mono 方法下和 VINS-RSD 方法下快速序列的漂移分别大概在  $-150\text{ mm}\sim 150\text{ mm}$  之间和  $-100\text{ mm}\sim 100\text{ mm}$  之间, VINS-Mono 方法下和 VINS-RSD 方法下慢速序列的漂移分别大概在  $-700\text{ mm}\sim 700\text{ mm}$  之间和  $-500\text{ mm}\sim 300\text{ mm}$  之间, VINS-RSD 方法在轨迹 1 上的漂移明显小于 VINS-Mono 方法. 在  $20\text{ m}\sim 40\text{ m}$  之间误差有比较明显的增大趋势, 因为此时的图像序列是无法检测到闭环的.

在闭环情况下对轨迹 1 的每个序列进行 10 次实验, 其结果见表 1, 任何速度下 VINS-RSD 方法

误差的最大值、均值、中值、最小值以及均方根都要小于 VINS-Mono 方法. 总体而言序列速度越快, 其误差越小, 因为此时 IMU 的漂移更小.

在系统闭环情况下利用轨迹 1 进行实验, VINS-RSD 方法下慢速序列的均方根误差约为  $23.9\text{ cm}$ , 中速序列的均方根误差约为  $12.1\text{ cm}$ , 快速序列的均方根误差约为  $7.1\text{ cm}$ . 3 种不同速度下, VINS-RSD 方法的误差最大值的平均值为  $33.7\text{ cm}$ , 相比 VINS-Mono 方法下降了  $51.7\%$ , 均值误差的平均值为  $12.3\text{ cm}$ , 相比 VINS-Mono 方法下降了  $46.2\%$ , 中值误差的平均值为  $11.4\text{ cm}$ , 相比 VINS-Mono 方法下降了  $46.3\%$ , 均方根误差的平均值为  $14.4\text{ cm}$ , 相比 VINS-Mono 方法下降了  $45.5\%$ .

表 1 有闭环时轨迹 1 的绝对轨迹误差 (单位: m)

Tab.1 Absolute trajectory errors of trajectory 1 with loop closure (unit: m)

序列	方法	最大值	均值	中值	最小值	均方根
轨迹 1 慢速	VINS-Mono	1.359 16	0.450 88	0.423 01	0.080 78	0.510 60
	VINS-RSD	0.545 43	0.208 80	0.199 59	0.019 80	0.239 22
轨迹 1 中速	VINS-Mono	0.515 42	0.164 60	0.139 40	0.019 65	0.195 35
	VINS-RSD	0.304 26	0.099 72	0.083 38	0.011 35	0.120 60
轨迹 1 快速	VINS-Mono	0.221 46	0.075 25	0.069 15	0.011 57	0.085 23
	VINS-RSD	0.162 24	0.062 66	0.057 88	0.008 56	0.071 01

表 2 无闭环时轨迹 1 的绝对轨迹误差 (单位: m)

Tab.2 Absolute trajectory errors of trajectory 1 without loop closure (unit: m)

序列	方法	最大值	均值	中值	最小值	均方根
轨迹 1 慢速	VINS-Mono	1.735 71	0.512 88	0.424 15	0.064 38	0.628 18
	VINS-RSD	1.038 54	0.259 07	0.160 13	0.043 88	0.345 44
轨迹 1 中速	VINS-Mono	0.859 36	0.194 82	0.126 00	0.013 68	0.262 59
	VINS-RSD	0.502 76	0.122 51	0.078 46	0.012 04	0.166 72
轨迹 1 快速	VINS-Mono	0.331 43	0.091 17	0.074 22	0.007 51	0.111 52
	VINS-RSD	0.239 16	0.075 48	0.063 34	0.005 15	0.091 83

对比表 1 和表 2 可知, 在任何速度下, VINS-RSD 方法的误差最大值、均值、中值、最小值以及均方根都要小于 VINS-Mono 方法. 速度越快的序列, 其误差越小, 因为此时 IMU 的漂移更小.

在系统无闭环情况下利用轨迹 1 进行实验, VINS-RSD 方法下慢速序列的均方根误差约为  $34.5\text{ cm}$ , 中速序列的均方根误差约为  $16.7\text{ cm}$ , 快速序列的均方根误差约为  $9.2\text{ cm}$ . VINS-RSD 方法的误差最大值的平均值为  $59.3\text{ cm}$ , 相比 VINS-Mono 方法下降了  $39.2\%$ ; 均值误差的平均值为  $15.2\text{ cm}$ , 相比 VINS-Mono 方法下降了  $42.8\%$ ; 中值误差的平均值为  $10.1\text{ cm}$ , 相比 VINS-Mono 方法

下降了  $51.6\%$ ; 均方根误差的平均值为  $20.1\text{ cm}$ , 相比 VINS-Mono 方法下降了  $39.7\%$ .

对比轨迹 1 有闭环的情况, 无闭环时误差更大, 表明闭环检测可以提升系统的定位精度. 无论有闭环还是无闭环, 快速序列的精度最高, 中速序列的精度次之, 慢速序列的精度最差.

#### 4.2 轨迹 2

图 6 为轨迹 2 的中速序列在 10 次评测中均方根误差最接近评测均值的 2 组数据. 图 6 直观地展示了 VINS-RSD 方法与 VINS-Mono 方法的评测结果在  $XOY$  平面上与轨迹真值的对比, 直观上看, 二者误差都较小, 从细节可以看出, VINS-RSD 方法

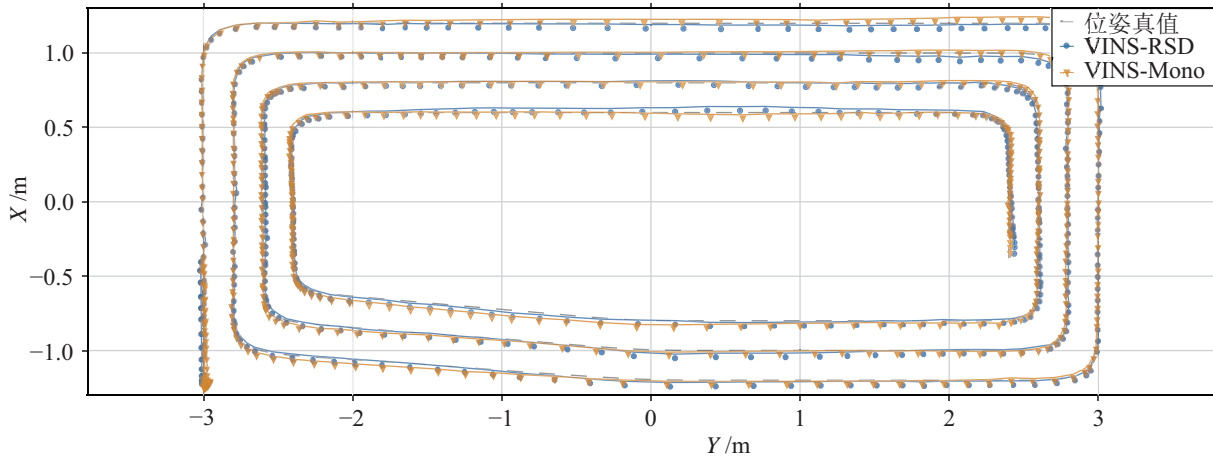


图6 轨迹2中速序列的绝对轨迹误差在VINS-RSD和VINS-Mono算法下的对比

Fig.6 The absolute trajectory error comparison of the medium sequence of trajectory 2 under VINS-RSD algorithm and VINS-Mono algorithm

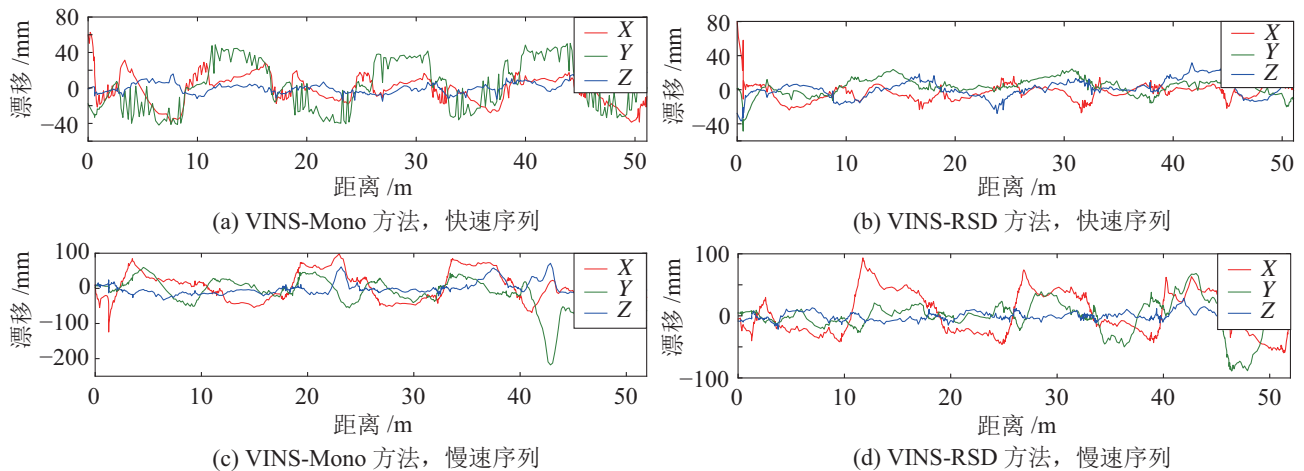


图7 轨迹2在VINS-Mono和VINS-RSD算法下在X、Y、Z轴上的误差

Fig.7 The errors of trajectory 2 in X, Y and Z axes under VINS-Mono and VINS-RSD algorithms

更加贴近轨迹的真值。

在闭环情况下利用轨迹2进行实验，VINS-Mono方法和VINS-RSD方法在X、Y、Z轴上的误差详情如图7所示。因为中速序列的效果位于快速和慢速之间，故本文中省去了对其绘图。由图7可知，VINS-Mono方法下快速序列的最大漂移大概在 $-60\text{ mm}\sim 80\text{ mm}$ 之间，VINS-RSD方法下序列的最大漂移同样在 $-60\text{ mm}\sim 80\text{ mm}$ 之间，但VINS-RSD方法在初始化完成之后的漂移迅速回落到约 $-30\text{ mm}\sim 30\text{ mm}$ 以内，相比于VINS-Mono方法下的 $-50\text{ mm}\sim 50\text{ mm}$ 明显要小一些。VINS-Mono方法下慢速序列的漂移大概在 $-200\text{ mm}\sim 100\text{ mm}$ 之间，VINS-RSD方法下慢速序列的漂移大概在 $-100\text{ mm}\sim 100\text{ mm}$ 之间，直观来看，VINS-RSD方法下的误差也要更小一些。

轨迹1和轨迹2的长度基本相同，同样模仿的

是手持设备在室内的运动，对比图5和图7可知，轨迹1相比轨迹2有更大的定位误差，且轨迹1在第 $20\text{ m}\sim 40\text{ m}$ 之间的误差有明显增大的趋势，这是因为轨迹1的运动随机性更强一些，而轨迹2的运动更加规律。轨迹2的运动轨迹类似于螺旋线，在所有位姿上基本都能检测到闭环，这极大地减小了定位误差。

对轨迹2的每个序列进行10次实验，对得到的结果取平均值。在有闭环的情况下，其绝对轨迹误差的统计信息如表3所示。从表3中可以看出，VINS-RSD方法除了在轨迹2快速序列上的最大误差大于VINS-Mono方法外，在其他情况下，其误差最大值、均值、中值、最小值以及均方根都要小于VINS-Mono方法。

在VINS-RSD方法带闭环情况下，轨迹2慢速序列的均方根误差约为 $3.9\text{ cm}$ ，中速序列的均方

根误差约为 2.3 cm, 快速序列的均方根误差约为 2.4 cm. 3 种不同速度下, VINS-RSD 方法的误差最大值的平均值为 8.4 cm, 相比 VINS-Mono 方法下降了 34.5%; 均值误差的平均值为 2.5 cm, 相比 VINS-Mono 方法下降了 27.0%; 中值误差的平均值为 2.3 cm, 相比 VINS-Mono 方法下降了 26.2%; 均方根误差的平均值为 2.9 cm, 相比 VINS-Mono 方法下降了 29.5%.

由表 4 可知, VINS-RSD 方法除了在轨迹 2 快速序列上的最大误差和均方根误差大于 VINS-Mono 方法外, 在其他情况下, 其误差最大值、均值、中

值、最小值以及均方根都要小于 VINS-Mono 方法. VINS-RSD 方法下轨迹 2 慢速序列、中速序列和快速序列的均方根误差分别约为 17.8 cm、9.0 cm 和 9.6 cm.

同工况下对比 VINS-RSD 方法和 VINS-Mono 方法可知, VINS-RSD 方法的误差最大值的平均值为 28.2 cm, 相比 VINS-Mono 方法下降了 24.2%; 均值误差平均值为 10.8 cm, 相比 VINS-Mono 方法下降了 23.3%; 中值误差平均值为 10.3 cm, 相比 VINS-Mono 方法下降了 19.0%; 均方根误差平均值为 12.1 cm, 相比 VINS-Mono 方法下降了 39.7%.

表 3 有闭环时轨迹 2 绝对轨迹误差 (单位: m)

Tab.3 Absolute trajectory errors of trajectory 2 with loop closure (unit: m)

序列	方法	最大值	均值	中值	最小值	均方根
轨迹 2 慢速	VINS-Mono	0.250 47	0.051 10	0.044 34	0.003 16	0.063 58
	VINS-RSD	0.112 85	0.034 51	0.031 94	0.003 16	0.038 89
轨迹 2 中速	VINS-Mono	0.068 68	0.023 77	0.021 87	0.002 91	0.026 67
	VINS-RSD	0.057 71	0.021 05	0.019 47	0.002 03	0.023 44
轨迹 2 快速	VINS-Mono	0.066 75	0.028 52	0.027 39	0.002 67	0.031 65
	VINS-RSD	0.082 34	0.019 95	0.017 69	0.002 61	0.023 58

表 4 无闭环时轨迹 2 绝对轨迹误差 (单位: m)

Tab.4 Absolute trajectory errors of trajectory 2 without loop closure (unit: m)

序列	方法	最大值	均值	中值	最小值	均方根
轨迹 2 慢速	VINS-Mono	0.661 52	0.231 45	0.203 52	0.044 45	0.270 42
	VINS-RSD	0.358 42	0.159 84	0.152 48	0.027 17	0.177 83
轨迹 2 中速	VINS-Mono	0.274 98	0.107 75	0.095 76	0.019 24	0.122 61
	VINS-RSD	0.168 81	0.081 32	0.078 94	0.008 08	0.089 52
轨迹 2 快速	VINS-Mono	0.178 30	0.084 55	0.081 36	0.014 66	0.091 71
	VINS-RSD	0.317 27	0.083 90	0.076 84	0.009 75	0.096 05

同轨迹 2 有闭环的情况相比, 无闭环时误差更大, 表明闭环检测可以提升系统的定位精度.

总体而言, 对于 6 个不同的轨迹序列, 在有闭环的情况下, VINS-RSD 方法的绝对轨迹误差的均方根误差皆要小于 VINS-Mono 方法, 平均下降 30.76%. 在无闭环的情况下, 除了轨迹 2 的快速序列, 其余 5 个轨迹序列在 VINS-RSD 方法下的绝对轨迹误差的均方根误差皆小于 VINS-Mono 方法, 6 个轨迹序列的均方根误差平均下降 25.94%.

## 5 结论 (Conclusion)

本文针对当前单目视觉惯性 SLAM 中初始化需要加速度激励以及高 IMU 噪声下系统精度下降的问题, 提出了使用卷帘快门 RGB-D 相机的 VINS-RSD 方法, 文章主要完成了以下工作:

(1) 扩展了 WHU-RSVI 数据集, 对其中的 RGB 图像都生成了对应的深度图像, 深度数据兼容常用的 TUM 格式数据, 可以作为 RGB-D 视觉 SLAM 方法或者 RGB-D 视觉惯性 SLAM 等相关方法的评价基准.

(2) 在 VINS-Mono 方法的基础上提出了一种基于卷帘快门 RGB-D 相机的视觉惯性 SLAM 方法——VINS-RSD, 通过利用深度图的深度信息, 解决了单目视觉惯性 SLAM 方法只能在特定运动条件下才能完成初始化的问题. 并能通过深度数据, 建立稠密的点云地图. 在制作的数据集上进行评测, 结果表明, 提出的 VINS-RSD 方法在有闭环的情况下, 均方根误差比 VINS-Mono 方法平均下降了 30.76%; 在无闭环的情况下, 均方根误差平均下降了 25.94%.

(3) 提出了一种复合损失核函数, 在工程应用中, 可根据相机与 IMU 的特性, 通过在损失核函数中选择合适的 IMU 与相机置信因子, 达到更好的定位和建图效果。

在未来的工作中, 将进一步研究不同强度 IMU 噪声对视觉惯性 SLAM 算法的影响, 并建立可以用于导航的稠密地图, 从而进一步完善该方法。

### 参考文献 (References)

- [1] Mourikis A I, Roumeliotis S I. A multi-state constraint Kalman filter for vision-aided inertial navigation[C]//IEEE International Conference on Robotics and Automation. Piscataway, USA: IEEE, 2007. DOI: 10.1109/ROBOT.2007.364024.
- [2] Heo S, Park C G. Consistent EKF-based visual-inertial odometry on matrix Lie group[J]. IEEE Sensors Journal, 2018, 18(9): 3780-3788.
- [3] Bloesch M, Omari S, Hutter M, et al. Robust visual inertial odometry using a direct EKF-based approach[C]//IEEE/RSJ International Conference on Intelligent Robots and Systems. Piscataway, USA: IEEE, 2015: 298-304.
- [4] Leutenegger S, Furgale P, Rabaud V, et al. Keyframe-based visual-inertial SLAM using nonlinear optimization[C]//Proceedings of Robotis: Science and Systems. 2013. DOI: 10.15607/RSS.2013.IX.037.
- [5] Leutenegger S, Lynen S, Bosse M, et al. Keyframe-based visual-inertial odometry using nonlinear optimization[J]. International Journal of Robotics Research, 2015, 34(3): 314-334.
- [6] Qin T, Li P L, Shen S J. VINS-Mono: A robust and versatile monocular visual-inertial state estimator[J]. IEEE Transactions on Robotics, 2018, 34(4): 1004-1020.
- [7] Qin T, Pan J, Cao S Z, et al. A general optimization-based framework for local odometry estimation with multiple sensors[DB/OL]. (2019-01-11) [2020-01-22]. <https://arxiv.org/abs/1901.03638>.
- [8] Qin T, Cao S Z, Pan J, et al. A general optimization-based framework for global pose estimation with multiple sensors [DB/OL]. (2019-01-11) [2020-01-22]. <https://arxiv.org/abs/1901.03642>.
- [9] Rosinol A, Abate M, Chang Y, et al. Kimera: An open-source library for real-time metric-semantic localization and mapping [DB/OL]. (2020-03-04) [2020-12-22]. <https://arxiv.org/abs/1910.02490v1>.
- [10] Kerl C, Stuckler J, Cremers D. Dense continuous-time tracking and mapping with rolling shutter RGB-D cameras[C]//IEEE International Conference on Computer Vision. Piscataway, USA: IEEE, 2015. DOI: 10.1109/ICCV.2015.261.
- [11] Lovegrove S, Patron-Perez A, Sibley G. Spline fusion: A continuous-time representation for visual-inertial fusion with application to rolling shutter cameras[C]//British Machine Vision Conference. Bristol, UK: BMVA Press, 2013. DOI: 10.5244/C.27.93.
- [12] Patron-Perez A, Lovegrove S, Sibley G. A spline-based trajectory representation for sensor fusion and rolling shutter cameras[J]. International Journal of Computer Vision, 2015, 113: 208-219.
- [13] Lee C R, Yoon K J. Inertial-aided rolling shutter relative pose estimation[DB/OL]. (2017-12-01) [2020-12-22]. <https://arxiv.org/abs/1712.00184>.
- [14] Li W L, Tang Y, Ding C, et al. Visual-inertial ego-motion estimation using rolling-shutter camera in autonomous driving [EB/OL]. (2020-02-01) [2020-05-01]. <http://copland.udel.edu/guang/iros19-vins-workshop/papers/03.pdf>.
- [15] Yang N, Wang R, Gao X, et al. Challenges in monocular visual odometry: Photometric calibration, motion bias, and rolling shutter effect[J]. IEEE Robotics and Automation Letters, 2018, 3(4): 2878-2885.
- [16] Cao L K, Ling J, Xiao X H. The WHU rolling shutter visual-inertial dataset[J]. IEEE Access, 2020, 8: 50771-50779.
- [17] Mackay D. Generating synthetic stereo pairs and a depth map with POV-Ray[R/OL]. (2007-06-01) [2018-03-01]. <https://cite.seerx.ist.psu.edu/viewdoc/download?doi=10.1.1.68.8768&rep=rep1&type=pdf>.
- [18] Sturm J, Engelhard N, Endres F, et al. A benchmark for the evaluation of RGB-D SLAM systems[C]//IEEE/RSJ International Conference on Intelligent Robots and Systems. Piscataway, USA: IEEE, 2012. DOI: 10.1109/IROS.2012.6385773.
- [19] Smisek J, Jancosek M, Pajdla T. 3D with Kinect[C]//IEEE International Conference on Computer Vision Workshops. Piscataway, USA: IEEE, 2011. DOI: 10.1109/ICCVW.2011.6130380.
- [20] Fossati A, Gall J, Grabner H, et al. Consumer depth cameras for computer vision: Research topics and applications[M]//Advances in Computer Vision and Pattern Recognition. Berlin, Germany: Springer, 2012.
- [21] Forster C, Carlone L, Dellaert F, et al. On-manifold preintegration for real-time visual-inertial odometry[J]. IEEE Transactions on Robotics, 2017, 33(1). DOI: 10.1109/TRO.2016.2597321.
- [22] Mur-Artal R, Tardós J D. Visual-inertial monocular SLAM with map reuse[J]. IEEE Robotics and Automation Letters, 2017, 2(2): 796-803.
- [23] Gao X S, Hou X R, Tang J L, et al. Complete solution classification for the perspective-three-point problem[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2003, 25(8): 930-943.
- [24] Lepetit V, Moreno-Noguer F, Fua P. EPnP: An accurate  $O(n)$  solution to the PnP problem[J]. International Journal of Computer Vision, 2009, 81: 155-166.
- [25] Penate-Sanchez A, Andrade-Cetto J, Moreno-Noguer F. Exhaustive linearization for robust camera pose and focal length estimation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(10): 2387-2400.
- [26] Grupp M. Evo: Python package for the evaluation of odometry and SLAM[DB/OL]. (2017-12-01) [2018-12-22]. <https://github.com/MichaelGrupp/evo>.
- [27] Zhang Z C, Scaramuzza D. A tutorial on quantitative trajectory evaluation for visual(-inertial) odometry[C]//IEEE/RSJ International Conference on Intelligent Robots and Systems. Piscataway, USA: IEEE, 2018. DOI: 10.1109/IROS.2018.8593941.

### 作者简介:

曹力科 (1989 -), 男, 博士生. 研究领域: 同步定位与地图创建.

肖晓晖 (1969 -), 女, 博士, 教授. 研究领域: 机器人学.