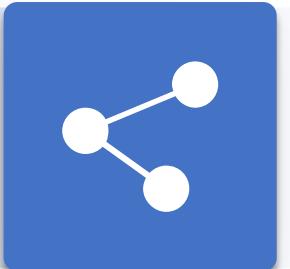




VIRAL SLAM: Tightly Coupled Camera-IMU-UWB-Lidar SLAM

Thien-Minh Nguyen, Shanghai Yuan, Muqing Cao
arXiv:2105.03296v3, 2021

智能网络与优化实验室





1

Introduction



介绍

将 UWB 集成到 SLAM 系统可以提供一层备份，防止 LiDAR 和 camera 失去跟踪，也允许用户获得全局定位信息。然而，在之前的一些集成 UWB 的方法中，要成功地将 UWB 与 SLAM 融合，尤其是在实时定位过程中，须估计 SLAM 坐标系 L 中锚点的坐标。

本文提出了一种新的方法。具体来说，利用锚点之间的距离测量，可以建立一个锚点的名义坐标，有效地定义一个与环境中要执行的任务相一致的首选帧 W ，然后在 BA 过程中进一步细化 L 与 W 之间的转换。随后，可以将锚点在 W 中的坐标转换为 L ，用于构建局部滑动窗口优化过程中的基于距离的因子。将锚点坐标估计和机器人状态估计分离以确保收敛，特别是在滑动窗口内运动太短的情况下，缺乏对锚点位置估计收敛的激励。还注意到，专注于一个简单而有效的 2 或 3 个锚点的 UWB 网络，在 UAV 中有多个体偏移测距节点。这种简单的网络允许相对准确的机器人初始化和锚定位置在 W ，这有利于准确无缝地将 UWB 集成到 SLAM 系统中。



贡献

- 提出了一个全面的 SLAM 框架，将不同传感方式的多个传感器紧密集成，即 LiDAR、camera、IMU 和 UWB 测距传感器。
- 使用本地地图构建的 LiDAR 点云提出了一种地图匹配边缘（ MMM ）方案的视觉特征。
- 设计了一个回环检测方案，由视觉位置识别触发，并通过两阶段点云对齐进一步完善。
- 提出了一种融合 UWB 的新方案，其中锚点位置和距离偏差的估计被授权给 Boundle Adjustment (BA) 线程，它们的值在 UWB 测距融合的局部滑动窗口优化中是固定的。



2

Methodology



坐标系 (Coordinate Frames)

定义局部坐标系 L ，其原点与初始时刻机器人坐标系的原点重合， z 轴指向重力反方向，机器人初始偏航角为 0。还需要固定锚点坐标，即世界坐标 W 。由于三个锚点位于同一平面上，这些锚点在 W 中的 3D 坐标可以通过锚点之间的距离加上名义高度 (nominal height) z^* 得到。如图更加详细地描述了此坐标系。

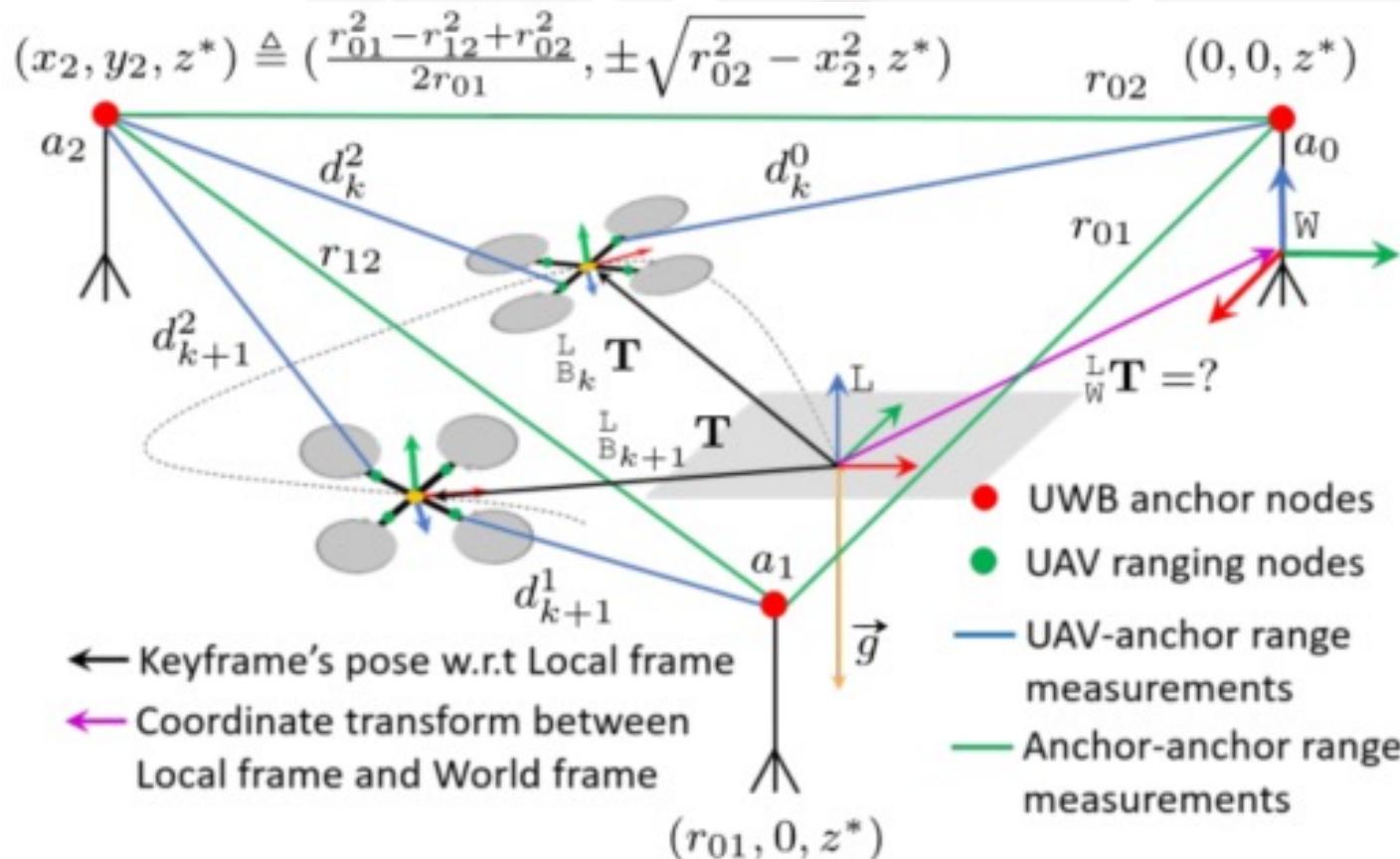


Fig. 2: A so-called world frame W can be defined when fixing the coordinates of the anchor nodes using the anchor-to-anchor distances. On the other hand, the SLAM system takes reference to a local coordinate frame L that coincides with the initial key frame's pose. To successfully combine UWB with SLAM, the transform L^T_W needs to be resolved.



状态估计 (State Estimates)

在每一个时间步 t_k ，定义滑动窗口 $\hat{\mathcal{T}}_k$ 由过去的 M 个时间步的机器人状态估计组成：

$$\hat{\mathcal{T}}_k = (\hat{\mathcal{X}}_w, \hat{\mathcal{X}}_{w+1}, \dots, \hat{\mathcal{X}}_k), w = k - M + 1$$

$$\hat{\mathcal{X}}_k = (\hat{\mathbf{q}}_k, \hat{\mathbf{p}}_k, \hat{\mathbf{v}}_k, \hat{\mathbf{b}}_k^\omega, \hat{\mathbf{b}}_k^a) \in SO(3) \times \mathbb{R}^{12}$$

$\hat{\mathbf{q}}_k, \hat{\mathbf{p}}_k, \hat{\mathbf{v}}_k$ 分别为局部帧 L 在时间步 t_k 的旋转四元数，位置和速度； $\hat{\mathbf{b}}_k^\omega, \hat{\mathbf{b}}_k^a$ 则为 IMU 加速度计和陀螺仪的偏差。

同时，定义在滑动窗口上被跟踪的 N_v^k 个视觉特征的逆深度的状态为

$$\hat{\lambda}^1, \hat{\lambda}^2, \dots, \hat{\lambda}^{N_v^k}, \hat{\Lambda}_k = (\hat{\lambda}^1, \hat{\lambda}^2, \dots, \hat{\lambda}^{N_v^k}) \in \mathbb{R}^{N_v^k}$$



状态估计 (State Estimates)

利用滑动窗估计方法，将关键帧进行边缘处理，得到全局位姿图。对存储的每一个关键帧 i ，定义它的位姿估计为 \hat{T}_i^L 。在 BA 过程中每当一定数量的新关键帧被接受或获得一个回环因子，位姿估计将被更新。

除了关键帧的位姿，还寻找估计以下与 UWB 相关的参数

$${}_{\mathcal{W}}^L \mathbf{T} = ({}_{\mathcal{W}}^L \mathbf{R}, {}_{\mathcal{W}}^L \mathbf{p}), {}_{\mathcal{W}}^L \mathbf{R} \in SO(3), {}_{\mathcal{W}}^L \mathbf{p} \in \mathbb{R}^3; \mathbf{b}^r \in \mathbb{R}$$

其中 ${}_{\mathcal{W}}^L \mathbf{T}$ 是局部坐标系和世界坐标系之间的坐标转换， \mathbf{b}^r 为测距偏差。



同步 (Synchronization)

VIRAL SLAM 中的同步方法为结合了之前的融合 IMU 的多 LiDAR、融合 IMU 和 UWB 数据的单 LiDAR 方案，并加入了立体图像作为新的输入。如图是同步方案的说明。在优化过程结束后，将会获得优化后的状态 $\hat{x}_w, \hat{x}_{w+1}, \dots, \hat{x}_k$ ，然后可以从其中选取一个作为全局地图的关键帧 \mathcal{K} 。

(CFC 为融合特征点云，Combined Feature Cloud)

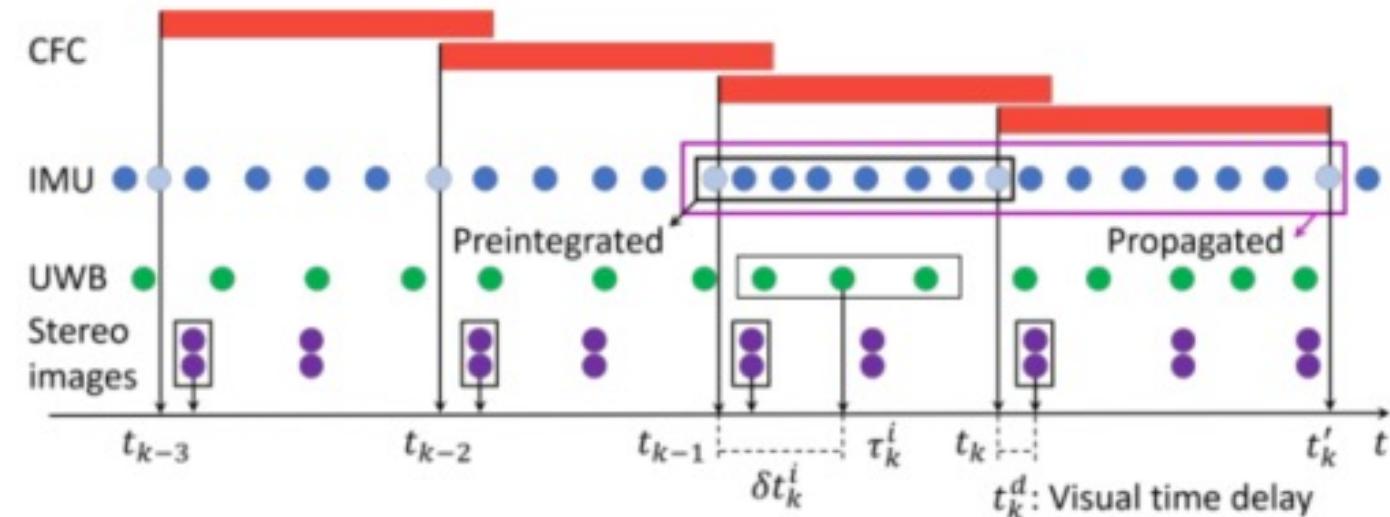


Fig. 3: Synchronization among the sensors. The light blue circles represent the interpolated IMU samples.



同步 (Synchronization)

简单来说，随机选中一个 LiDAR 作为主 LiDAR (the primary)，其时间戳被用于确定滑动窗口的时间步；其他 LiDAR 的输入被融合到 primary 当中，获得融合特征点云 (Combined Feature Cloud , CFC) 作为单传感器输入。

IMU 数据融合时间步用于传播 (propagation) 和预融合 (preintegration)。

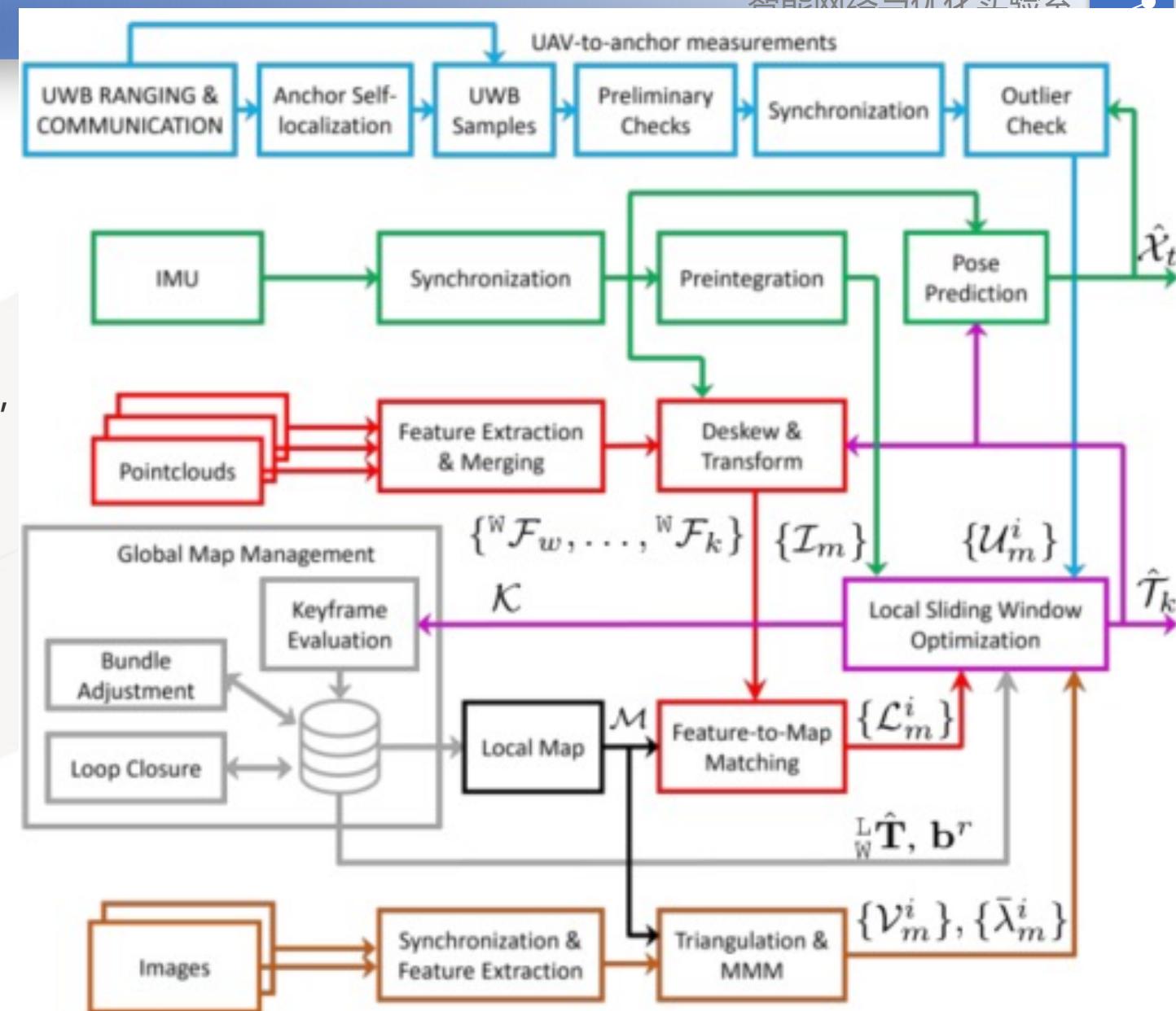
UWB 根据时间步的间隔被分组为 boundles。此处，记 UWB 的时间戳为 $\tau_k^i \in (t_{k-1}, t_k]$ 。

Camera 是由外部硬件设备触发的，因此它们的图像可以在与 LiDAR CFC 进一步同步之前成对同步。对于每一个时间步，记图像的测量延迟为 t_k^d 。当视觉特征在图像平面内被跟踪时，该时间延迟将用于补偿视觉特征的像素坐标。

系统概览 (System Overview)

如图为 VIRAL SLAM 的主要功能块，最具扩展性的是实时定位，其中所有传感器数据都被同步和处理，最终在损失函数中创建因子，并使用 ceres 求解器进行优化。除了这个对时间要求严格的模块外另一个模块在后台运行来管理关键帧、检测回环和 BA 优化。

这些块将在接下来讲述。





实时定位功能块 (Real-time Localization Function Blocks)

1. 局部滑动窗口优化 (Local Sliding Window Optimization)

在局部滑动窗口上进行状态估计，建立并优化求解了如下的损失函数：

$$f(\hat{\mathcal{T}}_k, \hat{\Lambda}_k)$$

$$\begin{aligned}
 &= \left\{ \sum_{m=w+1}^k \left\| \mathbf{r}_{\mathcal{I}}(\hat{\mathcal{X}}_{m-1}, \hat{\mathcal{X}}_m, \mathcal{I}_m) \right\|_{P_{\mathcal{I}_m}^{-1}}^2 + \sum_{m=w}^k \sum_{i=1}^{N_{\mathcal{L}}^m} \rho_H \left(\left\| \mathbf{r}_{\mathcal{L}}(\hat{\mathcal{X}}_m, \mathcal{L}_m^i) \right\|_{P_{\mathcal{L}_m^i}^{-1}}^2 \right) + \sum_{m=w}^k \sum_{i=1}^{N_{\mathcal{U}}^m} \left\| \mathbf{r}_{\mathcal{U}}(\hat{\mathcal{X}}_{m-1}, \hat{\mathcal{X}}_m, {}^L\hat{\mathbf{T}}, \hat{\mathbf{b}}^r, \mathcal{U}_m^i) \right\|_{P_{\mathcal{U}_m^i}^{-1}}^2 \right. \\
 &\quad \left. + \sum_{i=1}^{N_{\mathcal{V}}^k} \sum_{b \in \mathcal{C}^i} \rho_A \left(\left\| \mathbf{r}_{\mathcal{V}}(\hat{\mathcal{X}}_{m_a}, \hat{\mathcal{X}}_{m_b}, \hat{\lambda}^i, \mathcal{V}_{ab}^i) \right\|_{P_{\mathcal{V}_{ab}^i}^{-1}}^2 \right) \right\}
 \end{aligned}$$



实时定位功能块 (Real-time Localization Function Blocks)

1. 局部滑动窗口优化 (Local Sliding Window Optimization)

其中：

- $\rho_H(\cdot)$ 和 $\rho_A(\cdot)$ 分别表示 Huber 和 arctan 损失；
- $\mathcal{I}_m, \mathcal{L}_m^i, \mathcal{U}_m^i, \mathcal{V}_{ab}^i$ 分别表示 IMU、LiDAR、UWB 和视觉特征的观测数据；
- $N_{\mathcal{L}}^m \in \mathbb{N}$ 为从 CFC 中提取的特征图匹配 (FFM) 系数的数量， $N_{\mathcal{U}}^m \in \mathbb{N}$ 为在时间间隔 $(t_{m-1}, t_m]$ 内获取的 UWB 采样数据的数量， $N_{\mathcal{V}}^k \in \mathbb{N}$ 为从 t_w 到 t_k 滑动窗口内视觉特征的数量；
- \mathcal{C}^i 除了相机 C_a 外观察获取视觉特征 f^i 的相机集合；
- $\tilde{\lambda}^i$ 表示状态估计 $\hat{\lambda}^i$ 或地图边缘匹配 (MMM) 特征的边缘逆深度 $\bar{\lambda}^i$ 。



实时定位功能块 (Real-time Localization Function Blocks)

2. 传感器数据处理 (Sencor Data Processing)

2.1 LiDAR & IMU

参考：

T.-M. Nguyen, S. Yuan, M. Cao, Y. Lyu, T. H. Nguyen, and L. Xie, "*Miliom: Tightly coupled multi-input lidar-inertial odometry and mapping*," IEEE Robotics and Automation Letters, vol. 6, no. 3, pp. 5573–5580, May 2021.



实时定位功能块 (Real-time Localization Function Blocks)

2. 传感器数据处理 (Sencor Data Processing)

2.2 UWB

定义每一个 UWB 采样为 $u_m^i = (\check{d}^i, {}^W\boldsymbol{x}^i, \boldsymbol{y}^i, \tau_m^i, t_{m-1}, t_m)$ ，其中 \check{d}^i 为距离测量， ${}^W\boldsymbol{x}^i$ 为锚点相对于坐标系 W 的坐标， \boldsymbol{y}^i 为 UAV 测距节点在局部坐标系 $B_{\tau_k^i}$ 中的坐标， τ_m^i 为测距时间戳， t_{m-1} 和 t_m 为 τ_m^i 的前后时间步。然而，当前不同的是距离测量 \check{d}^i 在 $t_k + \delta t^i$ 的值定义为向量 ${}^L\boldsymbol{d}^i$ ，收到高斯噪声和偏差的影响：

$$\check{d}^i = \| {}^L\boldsymbol{d}^i \| + \boldsymbol{\eta}_u^i + \boldsymbol{b}^r, \boldsymbol{\eta}_{u_i} \sim \mathcal{N}(0, \sigma_u^2)$$

$${}^L\boldsymbol{d}^i = \boldsymbol{p}_m + \boldsymbol{R}_{m-1} \exp \left(s^i \log(\boldsymbol{R}_{m-1}^{-1} \boldsymbol{R}_m) \right) \boldsymbol{y}^i - a^i \boldsymbol{v}_{m-1} - b^i \boldsymbol{v}_m - {}_W^L \boldsymbol{R} \boldsymbol{x}^i - {}_W^L \boldsymbol{t}$$

其中， $s^i = \frac{\delta t_i}{\Delta t_m}$, $a^i = \frac{\Delta t_m^2 - \delta t_i^2}{2\Delta t_m}$, $b^i = \frac{(\Delta t_m - \delta t_i)^2}{2\Delta t_m}$, $\delta t_i = \tau_k^i - t_{m-1}$, $\Delta t_m = t_m - t_{m-1}$ 。因此，UWB 测距残差为

$$\boldsymbol{r}_u = \| {}^L\boldsymbol{d}(\hat{\boldsymbol{x}}_{m-1}, \hat{\boldsymbol{x}}_m, {}_W^L \tilde{\boldsymbol{T}}, u_m^i) \| + \bar{b}^r - \check{d}^i$$

在滑动窗口优化过程中 ${}_W^L \tilde{\boldsymbol{T}}$ 和 \bar{b}^r 保持不变。并且，UWB 不被融合直到 BA 过程更新了 ${}_W^L \tilde{\boldsymbol{T}}$ 和 \bar{b}^r 。



实时定位功能块 (Real-time Localization Function Blocks)

2. 传感器数据处理 (Sencor Data Processing)

2.3 Camera

在滑动窗口上，BRIEF 特征被跟踪并与时间步相关联。对于视觉特征 f^i 及其投影在相机 C_a 和 C_b 上的投影坐标对 $\mathcal{V}_{ab}^i = ({}^{C_a}\mathcal{Z}^i, {}^{C_b}\mathcal{Z}^i)$ ，其与观测值的残差定义为

$$\mathbf{r}_v(\hat{\mathcal{X}}_a, \hat{\mathcal{X}}_b, \mathcal{V}_{ab}^i) = \pi({}^{C_b}\hat{\mathbf{f}}^i) - {}^{C_b}\mathcal{Z}^i$$

$${}^{C_b}\hat{\mathbf{f}}^i = {}^{C_b}\mathbf{R}^{-1}(\bar{\mathbf{R}}_{m_b}^{-1}({}^L\hat{\mathbf{f}}^i - \hat{\mathbf{p}}_{m_b}) - {}^{C_b}\mathbf{t})$$

$${}^L\hat{\mathbf{f}}^i = \bar{\mathbf{R}}_{m_a} \left({}^{C_a}\mathbf{R} \left[(\hat{\lambda}^i)^{-1} {}^{C_a}\mathcal{Z}^i \right] + {}^{C_a}\mathbf{t} \right) + \hat{\mathbf{p}}_{m_a}$$

${}^{C_a}\mathbf{R}$, ${}^{C_b}\mathbf{R}$, ${}^{C_a}\mathbf{t}$, ${}^{C_b}\mathbf{t}$ 为相机的常量外参，坐标系 C_a 和 C_b 可以与相同的状态但不同的 camera 耦合，或相同 camera 的不同状态，或状态和 camera 都不同。

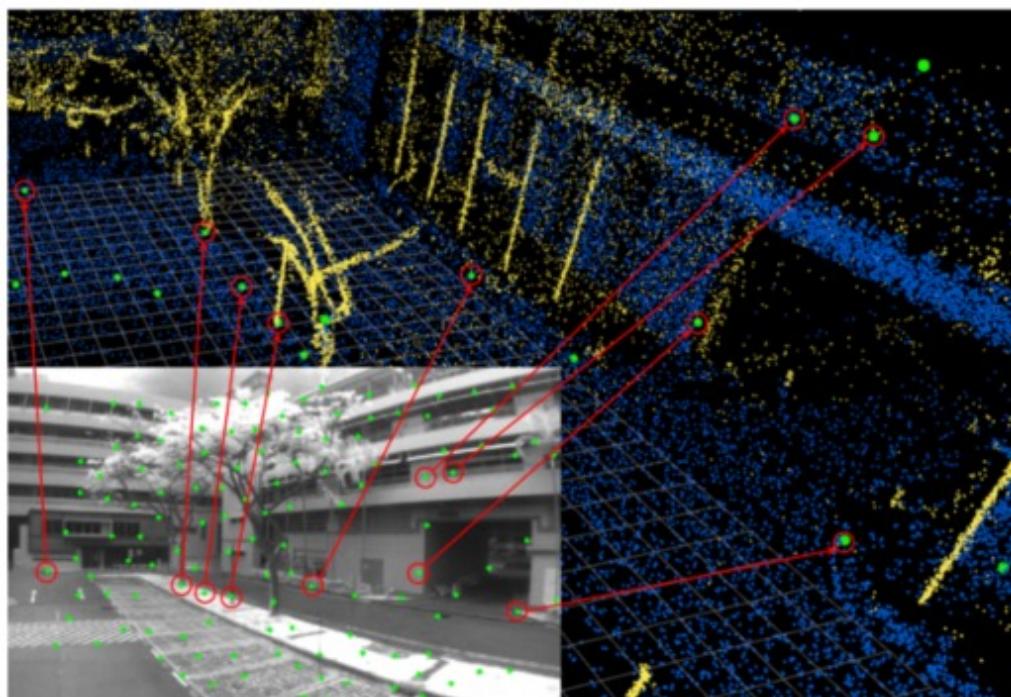


实时定位功能块 (Real-time Localization Function Blocks)

2. 传感器数据处理 (Sencor Data Processing)

2.3 Camera

通过检查是否视觉特征的 3D 坐标估计很好地匹配到 3D 局部地图 \mathcal{M} 上，可以边缘化这个特征为一个滑动窗口的固定先验。图中算法提供了 MMM 的处理过程，另一个图则说明了 MMM 对某些特征的处理结果。



Algorithm 1: MMM process on a visual feature ${}^{C_a}f^i$

```

Input:  ${}^{C_a}\mathcal{Z}^i$ ,  $\hat{\lambda}^i$ ,  $\hat{T}_{m_a}$ ,  $\mathcal{M}$ .
Output:  ${}^L\bar{f}^i$ .
1 Compute:  ${}^{C_a}\hat{f}^i = {}_{C_a}^B \mathbf{R} \left[ (\hat{\lambda}^i)^{-1} {}^{C_a}\mathcal{Z}^i \right] + {}_{C_a}^B \mathbf{t}$ ;
2 Compute:  ${}^L\hat{f}^i = \hat{\mathbf{R}}_{m_a} {}^{C_a}\hat{f}^i + \hat{\mathbf{p}}_{m_a}$ ;
3 Find  $\mathcal{N} = \text{KNN}({}^L\hat{f}^i, \mathcal{M})$ ;
4 Find  $\mathbf{n}^* = \underset{\mathbf{n} \in \mathbb{R}^3}{\operatorname{argmin}} \sum_{\mathbf{x} \in \mathcal{N}} \|\mathbf{n}^\top \mathbf{x} + 1\|^2$ ;
5 Compute:  $\bar{\mathbf{n}} = \frac{\mathbf{n}^*}{\|\mathbf{n}^*\|}$  and  ${}^L\hat{\mathbf{p}} = \hat{\mathbf{R}}_{m_a} {}_{C_a}^B \mathbf{t} + \hat{\mathbf{p}}_{m_a}$ ;
6 Compute:  $\mathbf{u} = \mathbf{R}_{m_a} {}_{C_a}^B \mathbf{R} {}^{C_a} \mathcal{Z}^i$ ;
7 Compute:  $\mathbf{c} = {}_{C_a}^L \hat{\mathbf{p}} - \mathbf{u}(1 + \bar{\mathbf{n}}^\top {}_{C_a}^L \hat{\mathbf{p}}) / (\bar{\mathbf{n}}^\top \mathbf{u})$ ;
8 if  $\min_{\mathbf{x} \in \mathcal{N}} \|\mathbf{x} - {}^L\hat{f}^i\| \leq 0.25$ 
9   and  $\|\mathbf{c} - \mathbf{x}\| < 1.0, \forall \mathbf{x} \in \{{}^L\hat{f}^i\} \cup \mathcal{N}$ 
10  and  $|\bar{\mathbf{n}}^\top \mathbf{x} + 1| < 0.1, \forall \mathbf{x} \in \mathcal{N}$ 
11 then
12   Set:  ${}^L\bar{f}^i = \mathbf{c}$ ;
13 end

```



全局优化块 (Global Optimization Blocks)

1. 关键帧管理 (Key Frame Management)

1.1 关键帧接收 (Key Frame Admission)

在每一次的滑动窗口上的优化后，寻找一些当前时间 $t_{k-M/2}$ 状态的最近邻居，并且如果其相对所有这些邻居的距离或旋转超过一个阈值，那么与这个时间步相关联的数据将被边缘化为先验。

参考：

T.-M. Nguyen, S. Yuan, M. Cao, Y. Lyu, T. H. Nguyen, and L. Xie, "*Miliom: Tightly coupled multi-input lidar-inertial odometry and mapping*," IEEE Robotics and Automation Letters, vol. 6, no. 3, pp. 5573–5580, May 2021.

1.2 关键帧选择 (Key Frame Selection)

关键帧的选取用于在 FMM (特征图匹配) 过程中建立局部点云地图。选择过程发生在优化过程之前，并且基于 IMU 的传播位姿 \tilde{T}_k 。因此，关键帧的集合为 $\{\mathcal{K}_a\} \cup \{\mathcal{K}_b\} \cup \{\mathcal{K}_c\}$ ，其中 $\{\mathcal{K}_a\}$ 为最后 M 帧的集合， $\{\mathcal{K}_b\}$ 为与 \tilde{T}_k 最近的 M 个邻居的集合， $\{\mathcal{K}_c\}$ 为以 \tilde{T}_k 为中心 $2m \times 2m \times 2m$ 体素范围内的关键帧。



全局优化块 (Global Optimization Blocks)

2. 束集调整 (Bundle Adjustment)

BA 过程是构建和优化如下的损失函数：

$$f(\hat{y}, {}^L_W T, \hat{b}^r) = \sum_{n=1}^N \|r_1(\hat{T}_{n-1}, \hat{T}_n, {}^{n-1}{}_n \hat{T})\|_{P_1^{-1}}^2 + \sum_{(p,c) \in \mathcal{H}} \|r_2(\hat{T}_p, \hat{T}_c, {}^p{}_c \hat{T})\|_{P_2^{-1}}^2 + \sum_{n=1}^N \sum_{i=1}^{N_{\mathcal{U}}^n} \|r_3(\hat{T}_n, {}^L_W \hat{T}, \hat{b}^r, \mathcal{U}_n^i)\|_{P_3^{-1}}$$

其中 $\hat{y} = (\hat{T}_0, \hat{T}_1, \dots, \hat{T}_N)$ 为关键帧位姿， ${}^{n-1}{}_n \hat{T}$ 和 ${}^p{}_c \hat{T}$ 为相对位姿和回环先验， \mathcal{H} 为回环对集合， \mathcal{U}_n^i 为一个边缘化的 UWB 测量，其时间戳与 t_n 时刻的关键帧相差 0.2s，即：

$$\mathcal{U}_n^i = (\check{d}^i, {}^W x^i, y^i, {}^{t_n} \bar{R}, {}^{t_n} \bar{t})$$

$\check{d}^i, {}^W x^i, y^i$ 在传感器数据处理部分已经提到， $({}^{t_n} \bar{R}, {}^{t_n} \bar{t})$ 则为 B_{t_n} 和 $B_{\tau_n^i}$ 之间的相对位姿，可由 IMU 传播得到。



全局优化块 (Global Optimization Blocks)

2. 束集调整 (Bundle Adjustment)

BA 过程是构建和优化如下的损失函数 :

$$f(\hat{y}, {}_W^L \bar{T}, \hat{b}^r) = \sum_{n=1}^N \| \mathbf{r}_1(\bar{T}_{n-1}, \bar{T}_n, {}^{n-1}{}_n \bar{T}) \|_{P_1^{-1}}^2 + \sum_{(p,c) \in \mathcal{H}} \| \mathbf{r}_2(\bar{T}_p, \bar{T}_c, {}^p{}_c \bar{T}) \|_{P_2^{-1}}^2 + \sum_{n=1}^N \sum_{i=1}^{N_{\mathcal{U}}^n} \| \mathbf{r}_3(\bar{T}_n, {}_W^L \bar{T}, \hat{b}^r, \mathcal{U}_n^i) \|_{P_3^{-1}}$$

残差 $\mathbf{r}_1(\cdot)$ 和 $\mathbf{r}_2(\cdot)$ 则是直接的 , 而 $\mathbf{r}_3(\cdot)$ 则为 :

$$\mathbf{r}_3(\bar{T}_n, {}_W^L \bar{T}, \hat{b}^r, \mathcal{U}_n^i) = \| \hat{d}^i \| + \hat{b}^r - \check{d}^i$$

$$\hat{d}^i = \hat{\mathbf{p}}_n + \hat{\mathbf{R}}_n \frac{t_n}{\tau_n^i} \hat{\mathbf{R}} \left(\mathbf{y}^i + \frac{t_n}{\tau_n^i} \hat{\mathbf{t}} \right) - {}_W^L \hat{\mathbf{R}} \mathbf{x}^i - {}_W^L \hat{\mathbf{t}}$$



全局优化块 (Global Optimization Blocks)

2. 束集调整 (Bundle Adjustment)

为保证位姿图由足够的激励使得 anchor-related 状态收敛，一开始不在 BA 损失函数中加入 r_3 ，而是在关键帧位姿空间分布满足一定条件后再添加。具体地，可以通过计算关键帧的集合稀释 (geometric dilution) 来衡量：

$$\Gamma = \left(\sum_{n=1}^N (\bar{\mathbf{p}}_n - \mu)(\bar{\mathbf{p}}_n - \mu)^T \right)^{-1}, \mu = \frac{1}{N} \sum_{n=1}^N \bar{\mathbf{p}}_n$$

在 Γ 上进行特征值分解得到特征值 $\sigma_1 \geq \sigma_2 \geq \sigma_3 > 0$ 。如果 $\sigma_1 < c_1$ 且 $\frac{\sigma_1}{\sigma_3} < c_2$ (c_1 和 $c_2 > 1$ 为预定义的参数)，那么可以开始往损失函数中添加残差 r_3 。最后可以得到 ${}^L\mathbf{T}$ 和 \mathbf{b}^r 的估计值。



全局优化块 (Global Optimization Blocks)

3. 回环检测 (Loop Closure)

为构建回环先验 p_cT ，使用一个三阶段的方法：

- 阶段一：首先，当一个新关键帧被接收后，将它的视觉特征与数据库用词袋模型（DBoW 库）进行比较。如果一个匹配被标记，则可以提取变换 T_c 和 T_p ，即相对于当前和之前的关键帧的位姿的变换。然后，在 T_p 之前和之后被接受的关键帧中寻找出一部分，利用其边缘化的 CFC 构建局部地图 ${}^{B_p}M_p$ (B_p 表示机器人坐标系)。
- 阶段二：使用 ICP 对齐 ${}^{B_c}F_c$ 和 ${}^{B_p}M_p$ 来获得一个匹配分数，同时获得 ${}^{B_p}{}^cT$ 的初始估计。如果匹配分数低于阈值，则执行阶段三。



全局优化块 (Global Optimization Blocks)

3. 回环检测 (Loop Closure)

为构建回环先验 ${}^p_c\bar{T}$ ，使用一个三阶段的方法：

- 阶段三：在 ${}^{B_c}\mathcal{F}_c$ 和 ${}^{B_p}\mathcal{M}_p$ 间执行 FMM 来计算 FMM 系数，然后构建如下损失函数并优化：

$$f\left({}^{B_p}_{B_c}\bar{\mathbf{T}}\right) = \sum_{i=1}^{N_{\mathcal{L}}^C} \rho\left(\left\| \mathbf{r}_{\mathcal{L}}\left({}^{B_p}_{B_c}\bar{\mathbf{T}}, \mathcal{L}_c^i\right) \right\|_{{P}_{\mathcal{L}_c^i}^{-1}}^2\right)$$

优化上式获得优化相对位姿 ${}^{B_p}_{B_c}\bar{\mathbf{T}}^*$ 后，如果比率 $f\left({}^{B_p}_{B_c}\bar{\mathbf{T}}\right)/N_{\mathcal{L}}^C$ 低于阈值，则 ${}^{B_p}_{B_c}\bar{\mathbf{T}}^*$ 被标记为回环先验 ${}^{B_p}_{B_c}\bar{T}$ 。



3

Experiment



实验设置

- 数据集 : NTU VIRAL
- 实验环境 : AirSim
- 评估 : ATE (Absolute Trajectory Error)



实验结果

TABLE II: ATE of VIRAL SLAM's key frame positions with different sensor combinations over the employed datasets (IMU is always used). All values are in m. The *average* ATE is calculated for each class of datasets.

Dataset	Lidars	Lidars +Cameras	Lidars +UWB	Lidars +UWB +Cameras
eee_01	0.0380	<u>0.0390</u>	0.0822	0.0861
eee_02	<u>0.0451</u>	0.0347	0.0647	0.0505
eee_03	0.0385	<u>0.0438</u>	0.0608	0.0494
nya_01	0.0429	<u>0.0436</u>	0.0545	0.0584
nya_02	<u>0.0463</u>	0.0416	0.0635	0.0551
nya_03	0.0383	<u>0.0392</u>	0.0696	0.0621
sbs_01	0.0441	<u>0.0483</u>	0.0585	0.0587
sbs_02	0.0512	0.0476	0.0547	0.0518
sbs_03	0.0514	<u>0.0524</u>	0.0696	0.0716
Average	<u>0.0440</u>	0.0434	0.0642	0.0604
bid_01	0.2104	0.2039	<u>0.1585</u>	0.1583
bid_02	0.6046	<u>0.6033</u>	0.6111	0.6029
bid_03	0.1904	0.1865	<u>0.1822</u>	0.1765
Average	0.3351	0.3312	0.3173	0.3126
nbh_01	0.0393	<u>0.0508</u>	0.1910	0.1938
nbh_02	<u>0.0374</u>	0.0364	0.1489	0.1618
nbh_03	<u>0.0453</u>	0.0342	0.1653	0.1413
nbh_04	0.0178	<u>0.0201</u>	0.1811	0.1963
nbh_05	0.0225	<u>0.0257</u>	0.3224	0.2763
Average	0.0325	<u>0.0335</u>	0.2017	0.1939

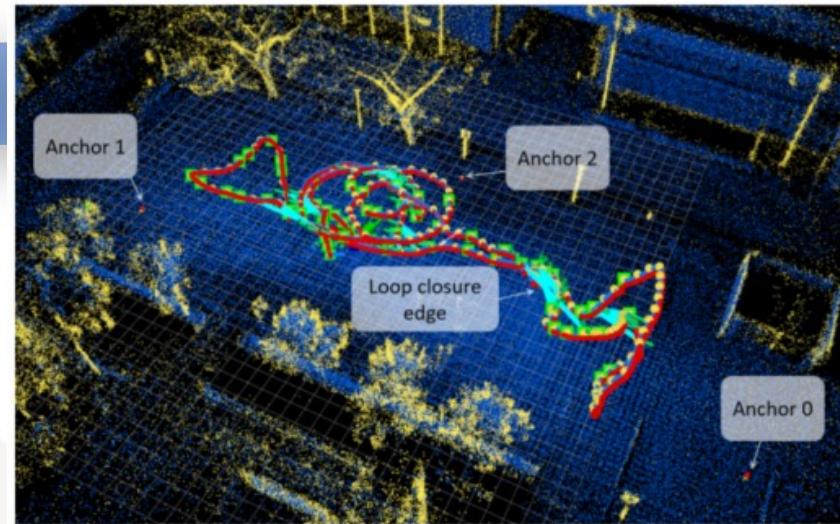


Fig. 6: VIRAL SLAM result on eee_02 dataset. The estimated trajectory is in blue, and ground truth is in red. The key frame poses are marked by the yellow circles. The activated key frames for local map building are highlighted by the green squares. The UWB anchors are also marked with red dots. The loop edges are marked with light cyan lines. Some visual features can be marked with small green circles.

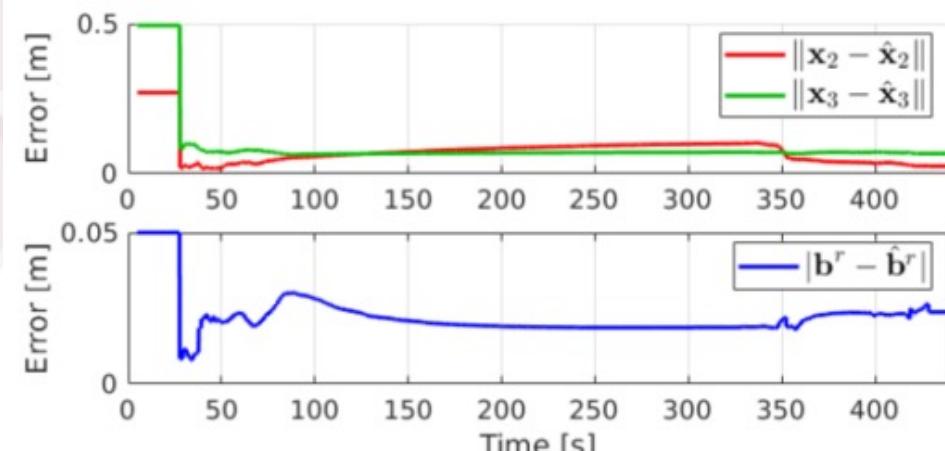


Fig. 7: Error of the estimates on anchor position and ranging bias by the BA process over time.



4

References



参考

- 无





謝謝

Thank You

THANKS

