2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

DAMS-LIO: A Degeneration-Aware and Modular Sensor-Fusion LiDAR-inertial Odometry

Fuzhang Han[1*], Han Zheng[2*], Wenjun Huang[1], Rong Xiong[1], Yue Wang[1], and Yanmei Jiao[3]

***Abstract*— With robots being deployed in increasingly complex environments like underground mines and planetary surfaces, the multi-sensor fusion method has gained more and more attention which is a promising solution to state estimation in the such scene. The fusion scheme is a central component of these methods. In this paper, a light-weight iEKF-based LiDAR-inertial odometry system is presented, which utilizes a degeneration-aware and modular sensor-fusion pipeline that takes both LiDAR points and relative pose from another odometry as the measurement in the update process only when degeneration is detected. Both the Cramer-Rao Lower Bound (CRLB) theory and simulation test are used to demonstrate the higher accuracy of our method compared to methods using a single observation. Furthermore, the proposed system is evaluated in perceptually challenging datasets against various state-of-the-art sensor-fusion methods. The results show that the proposed system achieves real-time and high estimation accuracy performance despite the challenging environment and poor observations.**

I. Introduction

Over the last decade, robotic systems have witnessed a significant increase in popularity and are used in increasingly complex applications like search and rescue [1], exploration, and mapping [2]. However, accurate and robust state estimation in these scenarios is not trivial due to the poorly lit (e.g., fog, dust) and self-similar structures (e.g., tunnels, long corridors), which leads to the lack of reliable visual and geometry features. These problems pose unique challenges to methods based on single sensor observations (e.g., incorrect feature matching or degradation problems) [3] [4] [5]. Thus multi-modal sensor fusion has been widely deployed in such tasks [6] [7]. Based on their design scheme, these approaches can be grouped into two main categories: tightly-coupled methods and loosely-coupled methods.

The former considers the error of different observations simultaneously, such as reprojection errors of visual features and point-to-plane error of Light Detection and Ranging (LiDAR) points [8] [9] [10]. They show marked improvement in accuracy and robustness in most scenarios. But as for above mentioned extreme conditions, tightly-coupled methods may be susceptible to sensor failures since most of them only use a single estimation engine [6]. Moreover, although these methods have an analysis of the quality of different measurements (e.g., number of feature points), in some conditions, like dark but geometrically feature-rich places, their localization accuracy may be lower than those using single high-fidelity measurements because performing multi-sensor fusion all the time injects more noise to the estimator.

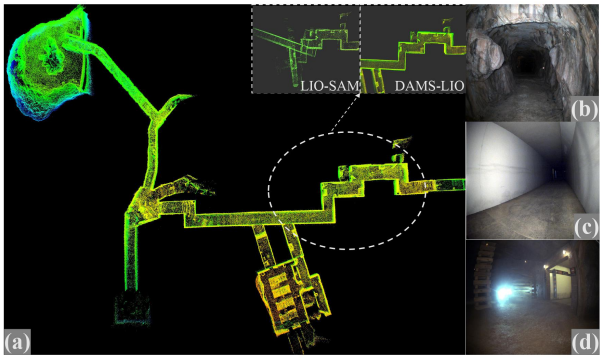


Fig. 1: (a) is the mapping result of DAMS-LIO in the CERBERUS DARPA Subterranean Challenge Datasets, which contains lots of challengeing environments as shown in (b)-(c) including darkness, long tunnel and textureless area. We compare our mapping results with the state-of-the-art LiDAR-inertial odometry, LIO-SAM, in the upper right corner of (a).

In comparison, loosely coupled methods are more robust in such perceptually-challenging conditions. Usually, they have a primary estimation engine (typically LiDAR-based methods) and regard the estimation result of different odometry as the initial value of scan-to-scan or scan-to-map [11] [12] when they detect degeneration of the primary engine. These approaches provide a decent trade-off between accuracy and robustness and have witnessed great success in the recent DARPA Robotics Challenge [13]. However, the other odometry in these methods have no influence on the hessian matrix of the system, so they do not fully utilize the information of other odometry and the performance may be severely compromised in the case of a poor prior.

Motivated by the discussion above, we propose a degeneration-aware and modular sensor-fusion pipeline in the iterated extended Kalman filter framework (iEKF), drawing on the state-of-the-art LiDAR-inertial odometry FAST-LIO2 [14]. Following the loosely-coupled model, it works as LiDAR-inertial odometry in a well-condition and performs sensor-fusion with other odometry when degeneration is detected. The distinctive insight of our approach is that we take both LiDAR feature points and relative pose provided by other odometry as measurements to participate in the

*Both authors contributed equally to this work

This work is supported by the National Key R&D Program of China(2020YFB1313300), "Ling-Yan" Research and Development Project of Zhejiang Province of China(2023C03185) and Natural Science Foundation of Zhejiang Province(LGG21F030012). [1]Fuzhang Han, Yue Wang, Wenjun Huang and Rong Xiong are with the State Key Laboratory of Industrial Control and Technology, Zhejiang University, Hangzhou, P.R. China. [2]Han Zheng is with the College of Electrical Engineering, Zhejiang University, China. [3]Yanmei Jiao is with the School of Information Science and Engineering, Hangzhou Normal University, Hangzhou 311121, China.

Corresponding author, ymjiao@hznu.edu.cn, Co-corresponding author, wangyue@iipc.zju.edu.cn.

subsequent update in iEKF so that the other odometery information can be fully utilized while relying less on them. Theoretical analysis based on Cramer-Rao Lower Bound (CRLB) [15] theorem demonstrates that integrating relative pose as measurements into the update process can achieve higher accuracy than as initial values in registration.

In summary, the contributions of this paper are listed as follows:

- A lightweight degeneration-aware and modular sensor-fusion LiDAR-inertial odometry system (DAMS-LIO) is proposed, which performs robust and accurate state estimation in extreme environments and offers a marked advantage for the complex exploration tasks by robots with limited computing resources.
- A novel sensor-fusion method to fully fuse the information of LiDAR and the other odometry is proposed, which takes both LiDAR points and relative pose from the other odometry as measurements in the update process only when degeneration is detected.
- Theoretical analysis based on CRLB theorem is performed to quantify the performance and demonstrate the high accuracy of the proposed sensor-fusion method.
- Extensive experiments on simulation and real-world datasets validate the robustness and accuracy of our method.

II. Related Works

Since autonomous robots in extreme and unknown environments (e.g., underground or planetary exploration) is subject to many challenges and limitations, individual sensor modalities might fail (e.g., due to camera blackouts or degenerate geometries for LiDAR). Hence in recent years, there are several efforts have been made for the multi-sensor fusion method, which can be classified as either loosely coupled methods or tightly coupled methods.

A. Tightly Coupled Method

The tightly-coupled method typically incorporates the measurement of different sensors into the state optimization process. In the work of [8], LiDAR-inertial and visual-inertial systems are fused based on a tightly-coupled smooth and mapping framework, which can work independently when failure is detected in one of them or jointly in a well-condition. [9] proposes an MSCKF-based LIC-fusion framework, which performs state estimation by IMU measurements, sparse visual and LiDAR features, and simultaneous online spatial and temporal calibration. Although these methods show marked improvement in the system's robustness and accuracy, they are susceptible to failure when the sensors are damaged and difficult to extend to other sensors.

B. Loosely Coupled Method

Zhang and Singh [16] propose V-LOAM, which utilizes the result of loosely-coupled Visual-Inertial odometry (VIO) prior to the initialization of the LiDAR mapping system. [11] comes up with a multi-sensor LiDAR-centric solution, LOCUS, which adds a health monitoring module to select a near-optimal prior to the LiDAR scan-matching optimization. These loosely-coupled approaches show more robust performance and higher resilience compared to tightly-coupled ones. However, their final performance still relies on laser scan alignment. Thereby these methods are still sensitive to the quality of LiDAR data. Hence, to fully utilize the information of the prior pose while retaining the advantages of the loosely-coupled method, we come up with a degeneration-aware and LiDAR-centric sensor fusion pipeline. It only receives pose measurements when LiDAR inertial odometry fails to make a trade-off between robustness and accuracy.

III. System Overview

The definition of each frame in our system is follows the typical LIO frame definitions. $\{L\}$ and $\{I\}$ are corresponding to the LiDAR and IMU frame respectively, while $\{G\}$ is a local-vertical reference frame whose origin coincides with the initial IMU position. $\{O\}$ represents the other odometry measurement, whose origin is denoted as $\{M\}$. ${}^{L}\boldsymbol{p}$ denotes the position of each point to the LiDAR frame.

The definition of some important and frequently used symbols are shown in Table I.

TABLE I: Important Symbols Definition

Symbols	Definition
t_k, t_{k+1}	The sample time of k-th and its next IMU measurement
τ_i, τ_{i+1}	The scan time of i-th and its next scan
I_k, I_i	The IMU frame at time t_k and τ_i
L_k, L_{k+1}	The LiDAR frame at time t_k and t_{k+1}
O_k, O_{k+1}	The other odometry frame at time t_k and t_{k+1}
$\mathbf{x}, \hat{\mathbf{x}}, \bar{\mathbf{x}}$	The true, predicted and updated value of x
$\hat{\mathbf{x}}^{\kappa}$	The κ-th update of $\boldsymbol{x}$ in the iterated Kalman filter
$\tilde{\mathbf{x}}$	The error between true x and its estimation value $\bar{\mathbf{x}}$
${}^{A}_{B}q, {}^{A}p_B$	The rotation (represented by quaternion and its rotation matrix is ${}^{A}_{B}R$) and translation from frame A to B

A. State Vector

The states estimated in our system containing current IMU states $\mathbf{x}_I$, the extrinsic parameters between LiDAR and IMU ${}^{I}\mathbf{x}_L$, and the transformation from other odometry frame to IMU frame ${}^{I}\mathbf{x}_O$. At time t_k, the state is written as:

$$\mathbf{x}_k = [\boldsymbol{x}_{I,k}^{\top} \quad \boldsymbol{x}_L^{\top} \quad \boldsymbol{x}_O^{\top}]^{\top} \tag{1}$$

$$\mathbf{x}_{I_k} = [{}^{G}_{I_k}\bar{\boldsymbol{q}}^{\top} \quad {}^{G}\boldsymbol{p}_{I_k}^{\top} \quad {}^{G}\boldsymbol{v}_{I_k}^{\top} \quad \boldsymbol{b}_{g_k}^{\top} \quad \boldsymbol{b}_{a_k}^{\top} \quad \boldsymbol{g}_k\ ]^{\top} \tag{2}$$

$$\mathbf{x}_L = [{}^{I}_{L}\bar{\boldsymbol{q}}^{\top} \quad {}^{I}\boldsymbol{p}_L^{\top}]^{\top}, \quad \mathbf{x}_O = [{}^{I}_{O}\bar{\boldsymbol{q}}^{\top} \quad {}^{I}\boldsymbol{p}_O^{\top}]^{\top} \tag{3}$$

${}^{G}\boldsymbol{v}_I$ is the velocity of IMU in global frame, $\boldsymbol{b}_{g,k}$ and $\boldsymbol{b}_{a,k}$ represent the gyroscope and accelerometer biases respectively, $\boldsymbol{g}$ is the gravity vector in frame $\{G\}$, $\mathbf{x}_L = \{{}^{I}_{L}\bar{\boldsymbol{q}}, {}^{I}\boldsymbol{p}_L\}$ and $\mathbf{x}_O = \{{}^{I}_{O}\bar{\boldsymbol{q}}, {}^{I}\boldsymbol{p}_O\}$ denotes the transformation from other odometry frame $\{O\}$ and LiDAR frame $\{L\}$ to IMU frame $\{I\}$.

B. IMU Propagation

Since measurements of IMU are affected by bias $\boldsymbol{b}$ and zero-mean Gaussian noise $\boldsymbol{n}$ [17], they can be modeled as:

$$\boldsymbol{\omega}_m(t) = {}^{I}\boldsymbol{\omega}(t) + \boldsymbol{b}_g(t) + \boldsymbol{n}_g(t) \tag{4}$$

$$\boldsymbol{a}_m(t) = {}^{I(t)}_{G}\boldsymbol{R}({}^{G}\boldsymbol{a}_I(t) + {}^{G}\boldsymbol{g}) + \boldsymbol{b}_a(t) + \boldsymbol{n}_a(t) \tag{5}$$

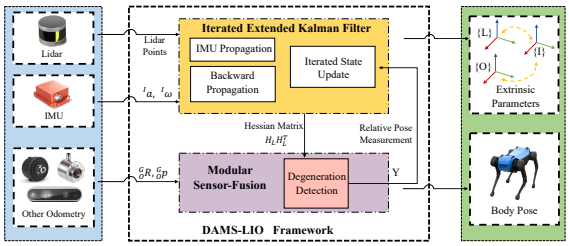


Fig. 2: Framework of the proposed DAMS-LIO.

where $\boldsymbol{\omega}_m(t)$ and $\boldsymbol{a}_m(t)$ are the raw measurement data. ${}^I\boldsymbol{\omega}(t)$ is the angular velocity of IMU in local frame $\{I\}$. ${}^G\boldsymbol{g}$ and ${}^G\boldsymbol{a}_I(t)$ are the acceleration of gravity and IMU expressed in the global frame. The IMU kinematics is the same as [18], to keep our presentation concise, we do not repeat the description here. To propagate the covariance matrix from time t_k to t_{k+1}, we have the generation format of the linearized discrete-time model, following [19] as:

$$\tilde{\boldsymbol{x}}_{k+1} = \boldsymbol{\Phi}_k \tilde{\boldsymbol{x}}_k + \boldsymbol{G}\boldsymbol{n}_k \tag{6}$$

where $\boldsymbol{\Phi}_k$ is the linearized system state transition matrix, $\boldsymbol{n}_k = [\boldsymbol{n}_g \quad \boldsymbol{n}_{wg} \quad \boldsymbol{n}_a \quad \boldsymbol{n}_{wa}]$ is the system noise. The error state is defined as $\tilde{\boldsymbol{x}} = \boldsymbol{x} - \hat{\boldsymbol{x}}$ for all variables except for quaternion, which is defined by the relation $q = \hat{q} \otimes \delta q$. Same as [18], the symbol $\otimes$ means quaternion multiplication, and the error quaternion is defined as $\delta q \simeq [\frac{1}{2}\boldsymbol{\delta\theta}^\top \quad 1]^\top$.

Denoting the covariance of $\boldsymbol{n}_k$ as $\boldsymbol{Q}_k$, then we can propagate the state covariance from t_k to t_{k+1} as:

$$\boldsymbol{P}_{k+1} = \boldsymbol{\Phi}\boldsymbol{P}_k\boldsymbol{\Phi}^\top + \boldsymbol{G}\boldsymbol{Q}_k\boldsymbol{G}^\top \tag{7}$$

where $\boldsymbol{\Phi} = \mathrm{diag}(\boldsymbol{\Phi}_I, \boldsymbol{\Phi}_O)$ and $\boldsymbol{G} = [\boldsymbol{G}_I^\top, \boldsymbol{G}_O^\top]^\top$. $\boldsymbol{\Phi}_I$ and $\boldsymbol{G}_I$ indicate the part related to the variable other than the extrinsic between other odometry and IMU, which are the same as the definition in [14]. Moreover $\boldsymbol{\Phi}_O = \boldsymbol{I}_{6\times6}$ and $\boldsymbol{G}_O = \boldsymbol{0}_{6\times12}$.

C. Measurement Model

1) **LiDAR measurement**: As for the measurements from LiDAR, we model them as in [14]. After motion compensation for the scan sampled at time τ_i, we define the j-th point at the local LiDAR frame as ${}^L\boldsymbol{p}_j$. Through backward propagation, the point can be changed to a scan-end measurement corresponding to the IMU measurement at t_k. Meanwhile, since each point should lie on a small plane patch in the map, we can get

$$\boldsymbol{0} = {}^G\boldsymbol{u}_j^\top ({}^G_{I_k}\boldsymbol{R}({}^I_L\boldsymbol{R}({}^L\boldsymbol{p}_j + {}^L\boldsymbol{n}_j) + {}^I\boldsymbol{p}_L) + {}^G\boldsymbol{p}_{I_k} - {}^G\boldsymbol{q}_j) \tag{8}$$

where ${}^G\boldsymbol{q}_j$ is the point on the small plane and ${}^G\boldsymbol{u}_j$ is the normal of the plane. ${}^I\boldsymbol{R}_L$ and ${}^G\boldsymbol{R}_{I_k}$ are the rotation matrix corresponding to the ${}^I_L\boldsymbol{q}$ and ${}^G_{I_K}\boldsymbol{q}$. ${}^L\boldsymbol{n}_j$ is the ranging and beam-directing noise of the point ${}^L\boldsymbol{p}_j$. (8) can also be summarized in a more compact from as

$$\boldsymbol{z}_l = \boldsymbol{0} = \boldsymbol{h}_l(\boldsymbol{x}_k, {}^L\boldsymbol{p}_j + {}^L\boldsymbol{n}_j) \tag{9}$$

To linearize the measurement model for the update, we approximate the measurement model by its first order approximation at $\hat{\boldsymbol{x}_k}$ as

$$\boldsymbol{r}_l = \boldsymbol{0} - \boldsymbol{h}_l(\boldsymbol{x}_k, {}^L\boldsymbol{p}_j) \simeq H_L\tilde{\boldsymbol{x}}_{\boldsymbol{k}} + \boldsymbol{v}_l \tag{10}$$

where $\boldsymbol{v}_l \in N(\boldsymbol{0}, \boldsymbol{R}_l)$ is the noise of Gaussian distribution due to the raw measurement noise ${}^L\boldsymbol{n}_j$. H_L is the jacobian matrix of residual $\boldsymbol{r}_l$ for error state as $\tilde{\boldsymbol{x}}_{\boldsymbol{k}}$, which is given as

$$H_L = {}^G\boldsymbol{u}_j^\top \begin{bmatrix} H_{L1} & \boldsymbol{I}_3 & \boldsymbol{0}_{3\times12} & H_{L2} & {}^G\hat{\boldsymbol{R}}_{I_k} & \boldsymbol{0}_{3\times6} \end{bmatrix} \tag{11}$$

$$H_{L1} = {}^G\hat{\boldsymbol{R}}_{I_k}\lfloor {}^L\boldsymbol{p}_j + {}^I\hat{\boldsymbol{p}}_L\times \rfloor, H_{L2} = -{}^G\hat{\boldsymbol{R}}_{I_k}{}^I\hat{\boldsymbol{R}}_L\lfloor {}^L\boldsymbol{p}_j\times\rfloor \tag{12}$$

2) **The other odometry measurement**: Due to the different publish frequency of each odometry, we perform the linear interpolation to obtain the other odometry poses with the estimated state at time t_k and t_{k+1} and denote their frame as O_{k-1} and O_k. We define the transformation matrix, ${}^AT_B$, from A to B as ${}^AT_B = [{}^A\boldsymbol{R}_B, {}^A\boldsymbol{p}_B;\ \boldsymbol{0}_{3\times3}, 1]$. Then, we have the transformation relationship as

$${}^{O_{k-1}}T_{O_k} = ({}^GT_{I_{k-1}}{}^IT_O)^{-1}({}^GT_{I_k}{}^IT_O) \tag{13}$$

where ${}^{O_{k-1}}T_{O_k} = {}^GT_{O_{k-1}}^{-1}{}^GT_{O_k}$ is the relative pose measurement calculated by other odometry. Hence based on (13), we can easily get the measurement model $\boldsymbol{z}_O = \begin{bmatrix}\boldsymbol{z}_r \\ \boldsymbol{z}_p\end{bmatrix}$ as

$$\boldsymbol{z}_r = {}^{O_{k-1}}\boldsymbol{R}_G{}^G\boldsymbol{R}_{I_k}{}^I\boldsymbol{R}_O \tag{14}$$

$$\boldsymbol{z}_p = {}^{O_{k-1}}\boldsymbol{R}_G(({}^G\boldsymbol{R}_{I_k}{}^I\boldsymbol{p}_O - {}^G\boldsymbol{R}_{I_{k-1}}{}^I\boldsymbol{p}_O) + {}^G\boldsymbol{p}_{I_k} - {}^G\boldsymbol{p}_{I_{k-1}}) \tag{15}$$

where ${}^{O_{k-1}}\boldsymbol{R}_G = ({}^G\boldsymbol{R}_{I_{k-1}}{}^I\boldsymbol{R}_O)^\top$. We assume the state at t_{k-1} is known, thus similar to (9) and (10), we have the residual of odometry measurement as $\boldsymbol{r}_O = \begin{bmatrix}\boldsymbol{r}_r \\ \boldsymbol{r}_p\end{bmatrix} \simeq \begin{bmatrix}H_{O_r}\tilde{\boldsymbol{x}}_{\boldsymbol{k}} + \boldsymbol{v}_r \\ H_{O_p}\tilde{\boldsymbol{x}}_{\boldsymbol{k}} + \boldsymbol{v}_p\end{bmatrix}$, where $\boldsymbol{r}_r$[1] and $\boldsymbol{r}_p$ are the rotation and translation errors calculated like (10), $\boldsymbol{v}_r \in N(\boldsymbol{0}, \boldsymbol{R}_r)$ and $\boldsymbol{v}_p \in N(\boldsymbol{0}, \boldsymbol{R}_p)$ are the corresponding Gaussian noise. Thus, based on (14) and (15), we have the jacobian matrix of rotation and translation residual for error state as

$$H_{O_r} = \begin{bmatrix}{}^I\hat{\boldsymbol{R}}_O^\top & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times18} & H_{O_{r1}} & \boldsymbol{0}_{3\times3}\end{bmatrix} \tag{16}$$

$$H_{O_p} = \begin{bmatrix}H_{O_{p1}} & H_{O_{p2}} & \boldsymbol{0}_{3\times18} & H_{O_{p3}} & H_{O_{p4}}\end{bmatrix} \tag{17}$$

where

$$H_{O_{r1}} = \boldsymbol{I}_3 - {}^I\hat{\boldsymbol{R}}^\top{}_O{}^G\hat{\boldsymbol{R}}^\top{}_{I_k}{}^G\hat{\boldsymbol{R}}_{I_{k-1}}{}^I\hat{\boldsymbol{R}}_O \tag{18}$$

$$H_{O_{p1}} = -{}^I\hat{\boldsymbol{R}}^\top{}_O{}^G\hat{\boldsymbol{R}}^\top{}_{I_{k-1}}{}^G\hat{\boldsymbol{R}}_{I_k}\lfloor {}^I\hat{\boldsymbol{p}}_O\times\rfloor \tag{19}$$

$$H_{O_{p2}} = ({}^G\hat{\boldsymbol{R}}_{I_{k-1}}{}^I\hat{\boldsymbol{R}}_O)^\top \tag{20}$$

$$H_{O_{p3}} = \lfloor {}^I\hat{\boldsymbol{R}}_O^\top{}^G\hat{\boldsymbol{R}}^\top{}_{I_{k-1}}\hat{\boldsymbol{P}}_1\times\rfloor \tag{21}$$

$$\hat{\boldsymbol{P}}_1 = ({}^G\hat{\boldsymbol{R}}_{I_k} - {}^G\hat{\boldsymbol{R}}_{I_{k-1}})^I\hat{\boldsymbol{p}}_O + {}^G\hat{\boldsymbol{p}}_{I_k} - {}^G\hat{\boldsymbol{p}}_{I_{k-1}} \tag{22}$$

$$H_{O_{p4}} = ({}^G\hat{\boldsymbol{R}}_{I_{k-1}}{}^I\hat{\boldsymbol{R}}_O)^\top({}^G\hat{\boldsymbol{R}}_{I_k} - {}^G\hat{\boldsymbol{R}}_{I_{k-1}}) \tag{23}$$

D. Degeneration-Aware Update

Following [14], we have the following error state:

$$\boldsymbol{x}_k \boxminus \hat{\boldsymbol{x}}_k = (\hat{\boldsymbol{x}}_k^\kappa \boxplus \tilde{\boldsymbol{x}}_k^\kappa) \boxminus \hat{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}^\kappa \boxminus \hat{\boldsymbol{x}}_k + \boldsymbol{M}^\kappa\tilde{\boldsymbol{x}}_k^\kappa \tag{24}$$

[1]For rotation the minus operation is defined in Lie group

where $\boxplus/\boxminus$ means plus and minus operators in Lie group [20]. $\boldsymbol{M}^{\kappa}$ is partial differentiation of $(\hat{\boldsymbol{x}}_k^{\kappa} \boxplus \tilde{\boldsymbol{x}}_k^{\kappa}) \boxminus \hat{\boldsymbol{x}}_k$ with respect to $\tilde{\boldsymbol{x}}_k^{\kappa}$ evaluated at zeros:

$$\begin{aligned}\boldsymbol{M}^{\kappa} = \operatorname{diag}(\boldsymbol{A}(\delta^{G}\boldsymbol{\theta}_{I_k})^{-\top}, \boldsymbol{I}_{15\times 15}, \boldsymbol{A}(\delta^{I}\boldsymbol{\theta}_{L_k})^{-\top}, \\ \boldsymbol{I}_{3\times 3}, \boldsymbol{A}(\delta^{I}\boldsymbol{\theta}_{O_k})^{-\top}, \boldsymbol{I}_{3\times 3})\end{aligned} \tag{25}$$

where $\boldsymbol{A}(\cdot)$ is defined in [21] and $\delta^{X}\boldsymbol{\theta}_{Y} = {}^{X}\hat{\boldsymbol{R}}_{Y}^{\kappa} \boxminus {}^{X}\hat{\boldsymbol{R}}_{Y}$. Based on (24) and taking it into the first-order approximation measurement model mentioned above, the problem can be summarized as:

$$\min_{\tilde{\boldsymbol{x}}_k^{\kappa}}(\| \hat{\boldsymbol{x}}^{\kappa} \boxminus \hat{\boldsymbol{x}}_k \|^2_{\hat{\boldsymbol{P}}_k^{-1}} + \sum \| \boldsymbol{z}^{\kappa} + H^{\kappa}\tilde{\boldsymbol{x}}_k^{\kappa} \|^2_{\boldsymbol{R}^{-1}}) \tag{26}$$

We exploit the agile update scheme to avoid the noise of pose measurement degrading the LiDAR odometry in well-condition and robustifying the odometry in degeneration conditions. Theoretically, the eigenvalues corresponding to the degeneration dimensions are precisely zero. However, in practice, they are typically a small value in the actual calculation due to the noise of data and limited computational accuracy. Hence, we first calculate the eigenvalues $\{\lambda_i\}$ of the Hessian matrix $\boldsymbol{H}_L^{\top}\boldsymbol{H}_L$ and refer to the heuristic method in [22] to determine how well the geometry feature of the scene. If the eigenvalue is smaller than the thread, we can infer the existence of degeneration. Usually, the setting of this value depends on the user's experience, so it may need to be adjusted when encountering different scenarios.

If LIO is in well condition and assume we have m LiDAR measurements, then $\boldsymbol{z}^{\kappa} = \boldsymbol{z}_l^{\kappa} = [\boldsymbol{z}_{l1}^{\kappa}, \ldots, \boldsymbol{z}_{lm}^{\kappa}]^{\top}$, $H = H_L^{\kappa} = [H_{l1}^{\kappa}, \ldots, H_{lm}^{\kappa}]^{\top}$ and $\boldsymbol{R} = \boldsymbol{R}_L = \operatorname{diag}(\boldsymbol{R}_{l1}, \ldots, \boldsymbol{R}_{lm})$. Otherwise, if degeneration is detected, $\boldsymbol{z}^{\kappa} = [\boldsymbol{z}_l^{\kappa}, \boldsymbol{z}_r^{\kappa}, \boldsymbol{z}_p^{\kappa}]^{\top}$, $H = [H_L^{\kappa}, H_{O_r}^{\kappa}, H_{O_p}^{\kappa}]^{\top}$ and $\boldsymbol{R} = \operatorname{diag}(\boldsymbol{R}_L, \boldsymbol{R}_{\boldsymbol{O_r}}, \boldsymbol{R}_{\boldsymbol{O_p}})$. Then we can perform iterated Kalman filter update the same in [14].

IV. Cramér–Rao Lower Bound Theorem

In this section, we furnish the reader with further insight into the proposed fusion method and compare the accuracy with other purely LiDAR-based methods using the Cramér-Rao Lower Bound (CRLB). The CRLB is a lower bound on the variance of an estimator, which is often used to evaluate the performance of the data fusion method [23] [24]. Following [25], the CRLB is calculated by taking the inverse of the Fisher information matrix as:

$$CRLB = J^{-1} = \boldsymbol{H}^{\top}\boldsymbol{R}^{-1}\boldsymbol{H}^{-1} \tag{27}$$

where $\boldsymbol{H}$ and $\boldsymbol{R}$ are corresponding to the Jacobian and covariance matrix of measurement in section III. The smaller CRLB, the higher accuracy the system can achieve in theory. For the needs of analysis, the following lemmas [26] are needed.

Lemma 1: The inverse of a positive definite matrix is also positive definite

Lemma 2: A real symmetric matrix A is positive definite if there exists a real nonsingular matrix B such that $A = BB^{\top}$

Lemma 3: Let A be a positive definite matrix and B be a $m \times n$ real matrix. $B^{\top}AB$ is positive definite if the rank for B:$r(B) = n$

Based on the theory mentioned above, we then calculate and compare the CRLB of different methods separately.

1) **Purely LiDAR-based method**: Since the purely LiDAR-based method does not use odometry pose measurements, the extrinsic parameters related to odometry are removed from the state vector. Moreover, for simplicity of calculation, we ignore the variable whose corresponding part in H is a zero vector. Thus based on (11) and (12), the measurement Jacobian matrix of purely LiDAR-based method is rewritten as

$$\boldsymbol{H}_{li} = {}^{G}\boldsymbol{u}_j^{\top}\begin{bmatrix}\boldsymbol{H}_{L1} & \boldsymbol{I}_3 & \boldsymbol{H}_{L2} & {}^{G}_{I_k}\hat{\boldsymbol{R}}\end{bmatrix} = \begin{bmatrix}\boldsymbol{H}_{pose} & \boldsymbol{H}_{extrinsic}\end{bmatrix} \tag{28}$$

We substitute (28) into (27) and represent it in the form of a chunking matrix as follows:

$$\mathbf{J}_{li} = \begin{bmatrix}\boldsymbol{U} & \boldsymbol{B} \\ \boldsymbol{B}^{\top} & \boldsymbol{C}\end{bmatrix} \tag{29}$$

Since covariance matrix of LiDAR noise is a real symmetric matrix and the rank of H_{li} is equal to its row, we can reason that $\mathbf{J}_{li}{}^{-1}$ is positive define matrix, which is denoted as $\mathbf{J}_{li}{}^{-1} > 0$. Then, based on the inverse formula for chunking matrix, we can easily obtain the part corresponding to the estimated pose as

$$\mathrm{CRLB}_{li} = (\boldsymbol{U} - \boldsymbol{B}\boldsymbol{C}^{-1}\boldsymbol{B}^{\top})^{-1} \tag{30}$$

2) **Pose-fusion method**: Following the process mentioned above, we have the measurement Jacobian matrix as

$$\boldsymbol{H}_{pf} = \begin{bmatrix}\boldsymbol{H}_{pose} & \boldsymbol{H}_{extrinsic} & \mathbf{0}_{3\times 6} \\ \boldsymbol{H}_{pose1} & \mathbf{0}_{3\times 6} & \boldsymbol{H}_{extrinsic1} \\ \boldsymbol{H}_{pose2} & \mathbf{0}_{3\times 6} & \boldsymbol{H}_{extrinsic2}\end{bmatrix} \tag{31}$$

Then we substitute (31) into (27) and represent it same as (29):

$$\mathbf{J}_{pf} = \begin{bmatrix}\boldsymbol{U}+\boldsymbol{F} & \boldsymbol{B} & \boldsymbol{D} \\ \boldsymbol{B}^{\top} & \boldsymbol{C} & \mathbf{0} \\ \boldsymbol{D}^{\top} & \mathbf{0} & \boldsymbol{E}\end{bmatrix} \tag{32}$$

Then, we marginalize the odometry-extrinsic-related variable to get the marginalized Fisher Information Matrix of the variables $\mathbf{J}_{pfmar}$ as

$$\mathbf{J}_{pfmar} = \begin{bmatrix}\boldsymbol{U}+\boldsymbol{F}-\boldsymbol{D}\boldsymbol{E}^{-1}\boldsymbol{D}^{\top} & \boldsymbol{B} \\ \boldsymbol{B}^{\top} & \boldsymbol{C}\end{bmatrix} \tag{33}$$

Then following (30), the corresponding CRLB for estimated pose is

$$\mathrm{CRLB}_{pf} = (\boldsymbol{U} - \boldsymbol{B}\boldsymbol{C}^{-1}\boldsymbol{B}^{\top} + \boldsymbol{F} - \boldsymbol{D}\boldsymbol{E}^{-1}\boldsymbol{D}^{\top})^{-1} \tag{34}$$

3) **Comparison**: Considering a case in which only the odometry pose measurement is used (i.e., no extrinsic variables between LiDAR and IMU in the state vector), the Fisher information matrix is

$$\mathbf{J}_{op} = \begin{bmatrix}\boldsymbol{F} & \boldsymbol{D} \\ \boldsymbol{D}^{\top} & \boldsymbol{E}\end{bmatrix} \tag{35}$$

Under this condition, $(\boldsymbol{F} - \boldsymbol{D}\boldsymbol{E}^{-1}\boldsymbol{D}^{\top})^{-1}$ is the information matrix corresponding to estimated pose and thus obviously is positive definite. Based on Lemma 1, we can refer that

TABLE II: Accuracy Comparison Results of Each Method Operating on Various Challenge Environments

Dataset	CERBERUS								M2DGR					
	anymal1		anymal2		anymal3		anymal4		gate01		gate03		street03	
	max	mean	max	mean	max	mean	max	mean	max	mean	max	mean	max	mean
LOCUS	1.61	0.49	-	-	1.22	0.26	2.69	0.67	13.09	6.28	20.31	6.56	-	-
LVI-SAM	8.18	1.11	1.86	0.44	2.87	0.45	1.20	0.36	10.01	4.28	22.93	9.06	19.14	9.67
LIO-SAM	23.9	10.3	-	-	-	-	23.1	8.61	13.27	6.68	12.27	3.30	12.86	7.52
VINS-MONO	5.93	2.34	11.20	1.55	4.40	1.48	2.00	0.72	6.92	3.75	12.75	8.21	14.46	4.61
Fast-LIO2	**0.48**	**0.14**	0.41	**0.14**	-	-	22.75	0.55	4.55	2.29	8.37	2.93	5.22	0.72
DAMS-LIO	0.67	0.20	**0.35**	**0.14**	**0.73**	**0.17**	**1.09**	**0.27**	**2.73**	**0.94**	**3.38**	**2.24**	**1.69**	**0.47**

"-" means that the method fails. The units for all values are meters.

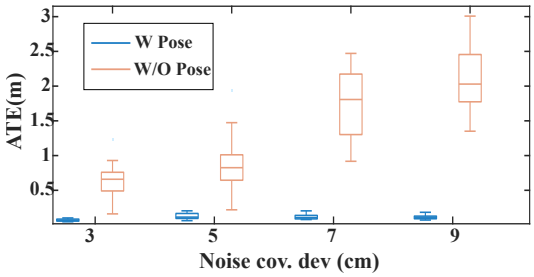


Fig. 3: Simulation validation result. The value of the horizontal coordinate represents the standard deviation of the noise. W and W/O pose mean with and without pose measurement.

$(\boldsymbol{F}-\boldsymbol{D}\boldsymbol{E}^{-1}\boldsymbol{D}^{\top})>0$ According to [27], we have

$$(\boldsymbol{U}-\boldsymbol{B}\boldsymbol{C}^{-1}\boldsymbol{B}^{\top}+\boldsymbol{F}-\boldsymbol{D}\boldsymbol{E}^{-1}\boldsymbol{D}^{\top})^{-1}<(\boldsymbol{U}-\boldsymbol{B}\boldsymbol{C}^{-1}\boldsymbol{B}^{\top})^{-1} \tag{36}$$

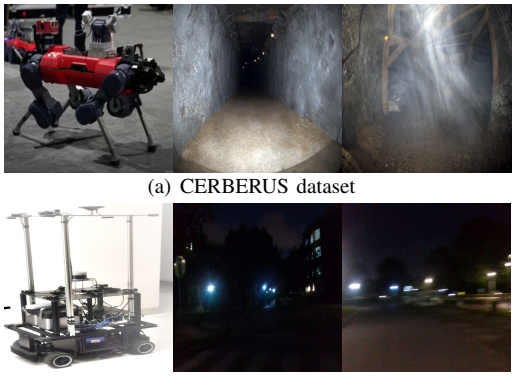
Then, it can be verified that $\mathrm{CRLB}_{pf}<\mathrm{CRLB}_{li}$. Namely, the method that fuses pose measurement and LiDAR points has a smaller lower bound on the covariance of pose estimation than the method that uses LiDAR points alone. Substituting the above analysis into the application scenario, it shows that our sensor-fusion method makes fuller use of the other odometry measurement than those using odometry information as an initial value, and therefore achieves better accuracy in pose estimation.

V. Experiments

In this section, we fully evaluate the proposed DAMS-LIO method with both simulation and real-world experiments. The simulation experiments are mainly used to verify the advantages of the proposed fusion method. Then we compare the accuracy and robustness of our method against other state-of-the-art methods in the public dataset and demonstrate the computing efficiency.

A. Simulation Validations

The primary purpose of the simulation experiment, which is built with the Gazebo simulator, is to validate the theoretical analysis in section IV. We construct a long corridor to simulate the degeneration environment and sample data through turtlebot3 mobile robot equipped with the velodyne VLP16 LiDAR sensor and wheel encoder. Similar to the simulator in Openvins [28], we obtain the simulated IMU data by sample interpolation of the true values of the robot trajectory.



(a) CERBERUS dataset

(b) M2DGR dataset

Fig. 4: Illustration of the real-world datasets used in experiment.

The gaussian noise $N(0,\sigma^2)$ is added to the LiDAR point data. Here the covariance σ^2 varies from 3 to 9 cm with an interval of 2 cm. We run our method repeatedly, with and without pose measurement, 20 times to obtain the absolute trajectory error (ATE) and evaluate the mean and dispersion of the errors in the form of box plots. As shown in Fig. 3, with an increase in the noise variance, the mean and covariance of the APE obtained by the method with pose observation are smaller than those without pose measurements. Therefore, this validates that fusing the odometry pose and LiDAR points observation simultaneously can achieve higher accuracy performance and lower covariance bound in pose estimation.

B. Real-world Experiments

To further validate the practical performance of our method, we compare it with current state-of-the-art state estimation systems on publicly available and challenging datasets. To maintain the viability of various methods operations, we choose the CERBERUS DARPA Subterranean Challenge dataset [29] and the M2DGR dataset Since these scenarios do not have degeneration of LIO, we select some periods in each scenario to limit the range of LiDAR measurements to produce degeneration. In addition, for gate03, we also added a period of time when the LiDAR data was lost. [30], which have a rich set of sensor types as shown in Fig 4.

1) **Accuracy**: We first use these challenging datasets to evaluate the accuracy of state estimation. We compare our method with other vision-based methods (VINS-MONO [31]), LiDAR-based method (LIO-SAM [32], Fast-LIO2), and sensor-fusion methods (LVI-SAM [8], LOCUS [11]). We

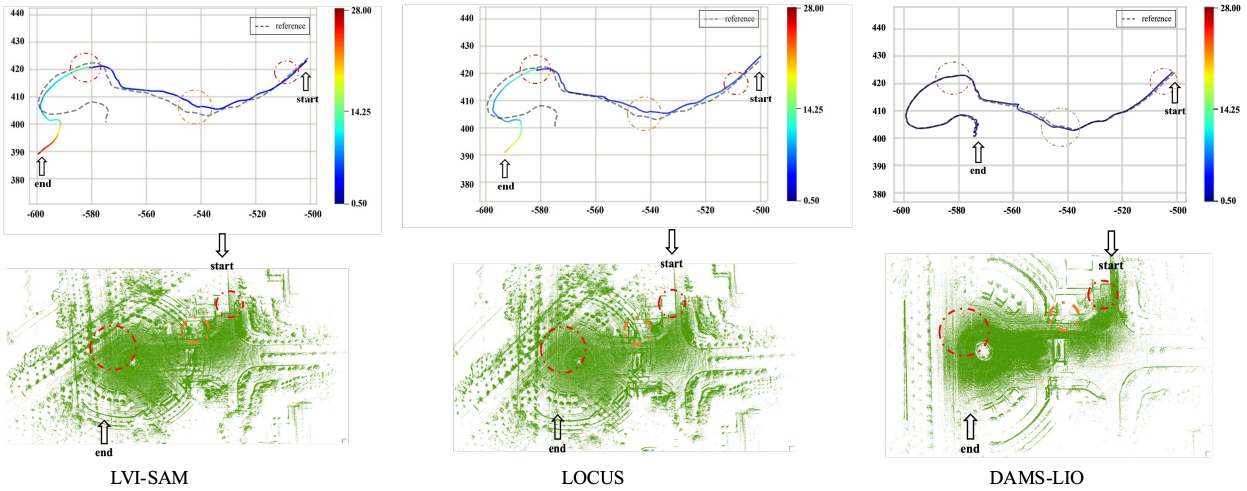


Fig. 5: Map and trajectory comparison of different sensor-fusion methods in the urban data sequence. Red dashed circles indicate limited LiDAR sensing distance, and green dashed means a drop in LiDAR measurement, which severely degrades the performance of LOCUS and LVI-SAM at the end.

use the trajectory output by VINS-MONO as the odometer input for LOCUS and our method. The EVO package [33] is adopted to calculate the translational part of ATE against the ground truth. The maximum and mean results of ATEs are presented in Table II. It can be seen that our method has lower trajectory error and lower error fluctuation range in most scenarios. Besides, when there is a large error in the odometry (i.e., the max error of VINS-MONO is large), methods such as LOCUS that use the odometry as the initial matching value are more likely to fail. In contrast, the proposed method still maintains good accuracy. For some scenarios, the degraded environment does not cause a large error in the LiDAR odometer (e.g. long corridors can be swept to the end), when the introduction of other odometers with larger errors will reduce the accuracy of our algorithm, such as anymal1.

2) **Robustness**: Our critical insight is that our method can have accuracy estimation relying on the fusion of LiDAR and pose measurement even with poor observations of other odometry and LiDAR points, which highlights the robustness of systems. Therefore, we select gate03 in dataset M2DGR to manually add some difficult scenarios, including the restricted maximum LiDAR sensing distance of 5m in the 15-20s and 165-180s time periods and the loss of LiDAR data at 100-105s (the part in the red and green dashed circle in Fig. 5). The comparison of trajectory accuracy and map-building results of the three different methods is shown in Fig 5. It can be seen that compared to LVI-SAM and LOCUS, our method is less sensitive to sensor data drop and poor sensor data. In addition, compared to LOCUS, since we consider both odometry pose and LiDAR point observations in the update process, it allows us to guarantee the accuracy of the estimation despite the poor data from LiDAR (limited measurement range) and poor odometry pose input (The result of VINS-MONO is poor at night). While methods like LOCUS, which only use the odometry as the initial value, will produce large errors and affect the map-building results.

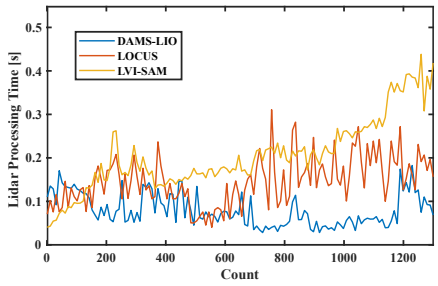


Fig. 6: Comparison of the running time of each system.

3) **Efficiency**: We compare the computation time taken by different methods to process the LiDAR data in operation, starting from the reception of LiDAR data until the pose is estimated. All efficiency tests are performed on a laptop (intel i5-9300 @2.4GHz×8) running ubuntu 16.04 LTS. As shown in Fig 6, our method can always maintain a lower runtime than other methods, with the runtime growth. Since the LiDAR data is published at 10 Hz, and the average processing time of our method in Fig 6 is mostly less than 0.1s, which indicates that our system achieves an efficiency of real-time operation.

VI. Conclusions

Multi-sensor fusion is a promising solution to the problem of state estimation in the perceptually-challenge environment while adopting a fusion scheme that can balance robustness and accuracy is intractable. To handle this problem, we propose the DAMS-LIO, a degeneration-aware and modular sensor-fusion LiDAR-inertial odometry system that incorporates both LiDAR and odometry pose measurements when degeneration is detected in LIO system. The CRLB theory analysis and extensive experiments are performed to demonstrate that our method has high accuracy, robustness, and efficiency. In future work, we intend to evaluate the observability information obtained from the odometry and retain only the most informative segments to improve accuracy and robustness.

REFERENCES

[1] M. Erdelj, E. Natalizio, K. R. Chowdhury, and I. F. Akyildiz, "Help from the sky: Leveraging uavs for disaster management," *IEEE Pervasive Computing*, vol. 16, no. 1, pp. 24–32, 2017.

[2] Y. Pan, X. Xu, X. Ding, S. Huang, Y. Wang, and R. Xiong, "Gem: online globally consistent dense elevation mapping for unstructured terrain," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–13, 2020.

[3] Y. Jiao, Y. Wang, B. Fu, X. Ding, Q. Tan, L. Chen, and R. Xiong, "2-entity ransac for robust visual localization in changing environment," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 2478–2485.

[4] Y. Jiao, L. Liu, B. Fu, X. Ding, M. Wang, Y. Wang, and R. Xiong, "Robust localization for planar moving robot in changing environment: A perspective on density of correspondence and depth," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 4006–4012.

[5] T. Shan and B. Englot, "Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 4758–4765.

[6] S. Zhao, H. Zhang, P. Wang, L. Nogueira, and S. Scherer, "Super odometry: Imu-centric lidar-visual-inertial estimator for challenging environments," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 8729–8736.

[7] X. Jing, X. Ding, R. Xiong, H. Deng, and Y. Wang, "Dxq-net: differentiable lidar-camera extrinsic calibration using quality-aware flow," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 6235–6241.

[8] T. Shan, B. Englot, C. Ratti, and D. Rus, "Lvi-sam: Tightly-coupled lidar-visual-inertial odometry via smoothing and mapping," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 5692–5698.

[9] X. Zuo, P. Geneva, W. Lee, Y. Liu, and G. Huang, "Lic-fusion: Lidar-inertial-camera odometry," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 5848–5854.

[10] J. Lin, C. Zheng, W. Xu, and F. Zhang, "R ^{2} live: A robust, real-time, lidar-inertial-visual tightly-coupled state estimator and mapping," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7469–7476, 2021.

[11] M. Palieri, B. Morrell, A. Thakur, K. Ebadi, J. Nash, A. Chatterjee, C. Kanellakis, L. Carlone, C. Guaragnella, and A.-a. Agha-Mohammadi, "Locus: A multi-sensor lidar-centric solution for high-precision odometry and 3d mapping in real-time," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 421–428, 2020.

[12] S. Khattak, H. Nguyen, F. Mascarich, T. Dang, and K. Alexis, "Complementary multi–modal sensor fusion for resilient robot pose estimation in subterranean environments," in *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2020, pp. 1024–1029.

[13] T. Rouček, M. Pecka, P. Čížek, T. Petříček, J. Bayer, V. Šalanskỳ, D. Heřt, M. Petrlík, T. Báča, V. Spurnỳ *et al.*, "Darpa subterranean challenge: Multi-robotic exploration of underground environments," in *International Conference on Modelling and Simulation for Autonomous Systems*. Springer, 2019, pp. 274–290.

[14] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "Fast-lio2: Fast direct lidar-inertial odometry," *IEEE Transactions on Robotics*, 2022.

[15] J. D. Gorman and A. O. Hero, "Lower bounds for parametric estimation with constraints," *IEEE Transactions on Information Theory*, vol. 36, no. 6, pp. 1285–1301, 1990.

[16] J. Zhang and S. Singh, "Laser–visual–inertial odometry and mapping with high robustness and low drift," *Journal of field robotics*, vol. 35, no. 8, pp. 1242–1264, 2018.

[17] Y. Yang, P. Geneva, K. Eckenhoff, and G. Huang, "Degenerate motion analysis for aided ins with online spatial and temporal sensor calibration," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 2070–2077, 2019.

[18] M. Li and A. I. Mourikis, "High-precision, consistent ekf-based visual-inertial odometry," *The International Journal of Robotics Research*, vol. 32, no. 6, pp. 690–711, 2013.

[19] A. I. Mourikis, S. I. Roumeliotis *et al.*, "A multi-state constraint kalman filter for vision-aided inertial navigation." in *ICRA*, vol. 2. Citeseer, 2007, p. 6.

[20] J. Sola, J. Deray, and D. Atchuthan, "A micro lie theory for state estimation in robotics," *arXiv preprint arXiv:1812.01537*, 2018.

[21] D. He, W. Xu, and F. Zhang, "Kalman filters on differentiable manifolds," *arXiv preprint arXiv:2102.03804*, 2021.

[22] X. Ding, F. Han, T. Yang, Y. Wang, and R. Xiong, "Degeneration-aware localization with arbitrary global-local sensor fusion," *Sensors*, vol. 21, no. 12, p. 4042, 2021.

[23] J. Domhof, R. Happee, and P. Jonker, "Multi-sensor object tracking performance limits by the cramer-rao lower bound," in *2017 20th International Conference on Information Fusion (Fusion)*. IEEE, 2017, pp. 1–8.

[24] C. Blanc, P. Checchin, S. Gidel, and L. Trassoudaine, "Data fusion performance evaluation for range measurements combined with cartesian ones for road obstacle tracking," in *2007 IEEE International Conference on Vehicular Electronics and Safety*. IEEE, 2007, pp. 1–6.

[25] M. Kowalski, Y. Bar-Shalom, P. Willett, B. Milgrom, and R. Ben-Dov, "Crlb for multi-sensor rotational bias estimation for passive sensors without target state estimation," in *Signal Processing, Sensor/Information Fusion, and Target Recognition XXVIII*, vol. 11018. SPIE, 2019, pp. 33–41.

[26] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 2012.

[27] B. Fu, F. Han, Y. Wang, Y. Jiao, X. Ding, Q. Tan, L. Chen, M. Wang, and R. Xiong, "High-precision multicamera-assisted camera-imu calibration: Theory and method," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–17, 2021.

[28] P. Geneva, K. Eckenhoff, W. Lee, Y. Yang, and G. Huang, "Openvins: A research platform for visual-inertial estimation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4666–4672.

[29] M. Tranzatto, T. Miki, M. Dharmadhikari, L. Bernreiter, M. Kulkarni, F. Mascarich, O. Andersson, S. Khattak, M. Hutter, R. Siegwart *et al.*, "Cerberus in the darpa subterranean challenge," *Science Robotics*, vol. 7, no. 66, p. eabp9742, 2022.

[30] J. Yin, A. Li, T. Li, W. Yu, and D. Zou, "M2dgr: A multi-sensor and multi-scenario slam dataset for ground robots," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2266–2273, 2021.

[31] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[32] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping," in *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2020, pp. 5135–5142.

[33] M. Grupp, "evo: Python package for the evaluation of odometry and slam." https://github.com/MichaelGrupp/evo, 2017.