Article

Degeneration-Aware Localization with Arbitrary Global-Local Sensor Fusion

Xiaqing Ding [1,2] ![ORCID]**, Fuzhang Han** [1,2]**, Tong Yang** [1,2]**, Yue Wang** [1,2,***] **and Rong Xiong** [1,2]

[1] College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China; xqding@zju.edu.cn (X.D.); 21960024@zju.edu.cn (F.H.); tong.yang@zju.edu.cn (T.Y.); rxiong@zju.edu.cn (R.X.)

[2] State Key Laboratory of Industrial Control and Technology, Hangzhou 310027, China

[*] Correspondence: ywang24@zju.edu.cn

Abstract: Global localization is a fundamental ability for mobile robots. Considering the limitation of single type of sensor, fusing measurements from multiple sensors with complementary properties is a valuable task for study. In this paper, we propose a decoupled optimization-based framework for global–local sensor fusion, which fuses the intermittent 3D global positions and high-frequent 6D odometry poses to infer the 6D global localization results in real-time. The fusion process is formulated as estimating the relative transformation between global and local reference coordinates, translational extrinsic calibration, and the scale of the local pose estimator. We validate the full observability of the system under general movements, and further analyze the degenerated movement patterns where some related system state would be unobservable. A degeneration-aware sensor fusion method is designed which detects the degenerated directions before optimization, and adds constraints specifically along these directions to relieve the effect of the noise. The proposed degeneration-aware global–local sensor fusion method is validated in both simulation and real-world datasets with different sensor configurations, and shows its effectiveness in terms of accuracy and robustness compared with other decoupled sensor fusion methods for global localization.

Keywords: multi-sensor fusion; global localization; degeneration-aware state estimation

1. Introduction

Precise global localization is of high importance in many applications, such as navigation and mapping. When there is no access to a pre-built environment map, the robot can only obtain the global positioning information via global-aware sensors like global positioning system (GPS) and ultra wide band (UWB). Generally these sensors can access global positional information with bounded error. However, due to the installation limitation of the ground stations and the signal block from obstacles, their measurements might be intermittent and suffer from large noise. Besides, the 3 degree-of-freedom (3D) positioning measurements are inadequate for some applications that require 6 degree-of-freedom (6D) global pose estimation. On the other hand, local sensors that measure the local environmental or kinematic information can achieve consecutive 6D pose estimation in real time [1–5]. However, the poses are estimated relative to the origins of the trajectories, which is unobservable in the global reference frame. The estimation results would drift along with time and distance.

In this paper, we set to integrate the global and local sensors to achieve 6D global drift-free localization. To do this, the extrinsic calibration between sensors is required, and so is the mapping that transforms the local estimations to the global coordinate. Besides, the scale of the local poses is also required since some local estimators such as monocular visual odometry can not recover it. Many researches investigate the sensor-fusion problem on specific sensor configurations, and the methods can be divided into two categories.

One kind of fusing methods tightly couple the measurements from different sensors under the filtering or optimization based frameworks [6,7]. Generally, the reference frames

of the local estimators are directly transformed to the global coordinates after initialization. During localization, the poses of the local estimators are propagated on the global coordinate. These methods are usually carefully designed for specific sensor configurations, and cannot be generalized to arbitrary configuration. To the best of our knowledge, none of these methods are compatible with the local estimators that cannot recover the absolute scale. Another kind of methods utilize the outputs from different pose estimation systems as observation, and loosely couple them to achieve global localization [2,8,9]. In this way, the pose estimation systems are decoupled, and the erroneous estimations or measurements from one system would not affect the others. Besides, the loosely-coupled methods are applicable to arbitrary sensor configurations due to the independency of the pose estimation front-ends [8–10].

However, few of these methods have analyzed the observability of the global–local sensor fusion system. As noted in [6], the sensor fusion systems would still remain globally unobservable even if the global measurements are given. Besides, the observability of the system would degenerate when the robot moves along certain motion patterns [11,12]. Many studies analyze the observability to guide the online movements or calibration design [11,13], but few of them pays attention to avoiding degeneration from worsening localization performance. One possible solution to address degeneration is to add heuristic constraints on the system, e.g., priors like constant velocity or planar motion models, which however would introduce false constraints on observable subspace if the hypothesis is violated. Some works [14,15] address the degeneration occurred in laser odometry and calibration, respectively. They first determine the unobservable directions according to the eigenvalues of systems, and truncate the state update along with these directions. These methods theoretically prevent noise from deteriorating unobservable states, but the eigenvalue threshold for observability distinction might change from case to case, and the optimizer has to be adapted for the truncated update.

In this paper, we propose a novel global–local sensor fusion framework that estimates the translational extrinsic calibration, the scale of the local poses, and the relative transformation from the reference frame of the local estimator to the global coordinate within a sliding-window, which are utilized to map the real-time local poses to the global frame for localization. We prove the full observability of this formulation under general motion patterns. To avoid the degeneration from deteriorating state estimation, we distinguish the degenerated subspace before each optimization, and add prior constraints specifically along with the unobservable directions. We validate the effectiveness of the proposed method in both simulated and real-world datasets with different sensor configurations. The experimental results also show that our method can effectively restrain the drift along degenerated directions and work out robust global localization results. To summarize, the contributions of the paper are listed as follow

- A novel global-local sensor fusion framework is proposed which is applicable to arbitrary sensor configurations. The observability and degenerated motion patterns of the proposed system are detailedly analyzed;
- A degeneration-aware optimization approach is designed to avoid the sensor fusion framework being deteriorated by the noise on the degenerated directions;
- Both simulated and real-world datasets with multiple sensor configurations are utilized to validate the generalization and effectiveness of the degeneration-aware sensor fusion framework for global localization.

2. Related Works

The study in this paper is relative to multi-sensor fusion and degeneration-aware state estimation. Considering the limitation of the single type of sensor, multi-sensor fusion is a widely researched area in robotics. Generally, the sensors selected for fusion possess complementary properties. For example, the monocular visual inertial system fuses external visual information with internal inertial information to estimate 6D pose with scale. In this paper, we pay attention to the fusion between local-aware sensors

(e.g., light detection and ranging (LIDAR), camera, inertial measurement unit (IMU)), and global-aware sensors (e.g., GPS, Motion Capture) for localization. The fusion frameworks can be divided into filtering and optimization-based methods. Typically, most of the filtering-based frameworks are developed based on Extended Kalman Filter (EKF) [6,8,16] and utilize global and local information to update the propagated inertial data. To reduce the first-order linearization error occurred in the EKF, some systems achieve sensor fusion taking advantages of the other filtering solutions such Partical Filter [17,18] or Sigma-point Kalman Filter [19,20]. Lynen et al. [8] propose a general multi-sensor fusion (MSF) method that can utilize arbitrary global or local measurements to update the propagated pose, in which the scale of the local estimator and the extrinsic calibration could also be estimated during the localization. However, the inertial instrument is indispensable and the observability is not validated in this method. Lee et al.[6] tightly couple the visual, inertial, and GPS information for global localization, and simultaneously estimates the extrinsic calibration and time-offset between sensors. Optimization-based methods are proven to be more accurate than filtering-based methods [21]. During optimization, the system states are iteratively estimated based on all of the valid measurements, which however require a large number of computational resources. Many works relieve the computational burden by only maintaining a sliding-window of states to achieve efficient optimization [9,22,23]. However, few of these fusion methods address the degeneration problem during online localization.

Observability is the vital property of dynamic systems which reflects whether the inner states of the systems can be estimated from the observed measurements [24]. In the global–local sensor fusion tasks, Lee et al. [6] prove that if directly adding the relative transformation between the reference frames of the visual inertial odometry (VIO) and GPS systems into the original VIO state for estimation, there still exists four unobservable directions even though the global measurements exist. They address this problem by transforming the states of the local VIO system to the global frame to achieve full observability, which is inherently equivalent to marginalizing out the relative transformation between the reference frames of VIO and GPS systems after initialization, and directly estimating the state of the local VIO system in the global frame. This parameterization is also utilized in some of the other global–local sensor fusion methods [7,10]. Another approach decouples the local and global estimators and estimates the relative transformation between their reference frames [25]. In this way the output of the high-frequency local estimator can be directly transformed to the global frame without latency, and makes the system more robust as the local and global estimators are decoupled. In this paper, besides the relative transformation between reference frames, we further add the scale of the local estimator and the extrinsic calibration between local and global sensors into the state, which makes the system applicable to arbitrary sensor configurations. We also prove its full observability under general movements.

Even though the fusion system is proved to be fully observable in general conditions, the observability would degenerate if the robot moves with insufficiently excited motion. Ref. [11] proves in detail that there exist several motion patterns that would introduce unobservable directions even if the global measurement is provided in the inertial navigation system (INS) aided framework. However, few works propose efficient solutions towards preventing degeneration from deteriorating estimation results. Refs. [14,15] address the degeneration problem by projecting the incremental update to the observable direction, which needs to modify the update step in general optimizers. In this paper, we automatically detect the degenerated directions before optimization, and specifically add constraints based on the calculated directions to prevent the unobservable parameters from being deteriorated by noise, which can be easily plugged in many existing systems developed with general optimizers.

3. Method

In this section, first the notations used throughout the paper are presented. Then we introduce the formulation of the global–local sensor fusion system and its detailed implementation for batch optimization. As a special discussion, we show that some commonly-appeared motions in robotics are actually degenerated for sensor fusion. Finally, we present our degeneration-aware optimization method that is designed to address the degeneration problem occurred during online localization.

3.1. Notation

The nomenclature used throughout the paper is defined in Table 1. We consider that the local estimator outputs 6D transformation T_C^O from the local sensor frame C to the reference frame of the odometry O. We assume that the global estimator outputs 3D positional measurement p_G^W of the global sensor frame G on the robot with respect to a fixed world frame W. To fuse the outputs from the global and local estimators, we estimate the extrinsic calibration T_G^C between sensors, the scale s of the local estimator, and the relative transformation T_O^W during online localization, which constitute the system state x. $T(s)$ denotes that the transformation T is mapped to the real scale by s. We draw the relation of the frames in Figure 1.

Table 1. Nomenclature.

Notation	Explanation
C	The sensor frame of the local estimator;
G	The sensor frame of the global estimator;
O	The reference frame of the local estimator;
W	The reference frame of the global estimator, which also is the world frame of the localization system;
L	The first frame in the sliding-window;
T	The transformation matrix in $\mathbb{SE}(3)$, where T_B^A represents the relative transformation from frame B to frame A defined in frame A;
R	The rotation matrix in $\mathbb{SO}(3)$, where R_B^A represents the rotation from frame B to frame A;
q	Unit quaternion in Hamilton notation, with q_B^A corresponding to R_B^A;
p	Translation vector in $\mathbb{R}^3$, where p_B^A denotes the translation from frame B to frame A defined in frame A. $p_{B_x}^A$, $p_{B_y}^A$ and $p_{B_z}^A$ denote the components of p_B^A along x, y, z axises;
x	The system state;
s	The scale of the local estimator in $\mathbb{R}^+$;
t	The timestamp of the received message;
n	The number of global measurements in the sliding-window;
$\epsilon_a, \epsilon_b, \epsilon_r$	The thresholds used for degeneration detection.

3.2. Global–Local Sensor Fusion System

The proposed sensor fusion system is demonstrated in Figure 2. It receives outputs from arbitrary local and global estimators, and produces the global localization results at the highest frequency of the local estimator. We estimate the transformation T_O^W to transform the output of the local estimator to the global frame instead of directly estimating the current pose in the global frame. To restrict the computational requirement and estimate the scale accurately, the fusion process is carried out on a sliding-window of recent n global sensor measurements and the related local sensor frames within the corresponding time duration, and we denote the first local sensor frame in the sliding-window as L. The length of the sliding-window in this work is maintained according to the distance, but can be

modified by any other criterion. Upon the sliding-window based formulation, we change the estimated relative transformation from T_O^W to T_L^W and the system state is maintained accordingly as

$$x = \{R_L^W, p_L^W, p_G^C, s\}$$

Generally, the global sensors possess lower frequency than the local estimators. Therefore, the system performs the fusion process each time it receives the measurement from the global estimator. As shown in Figure 2, during each fusion process, we first align the newly received global measurement with the two closest local frames to obtain the accurate local pose at the exact timestamp of the global measurement. Then, we detect whether there exists degenerated subspace according to the information within the sliding-window. If the degeneration occurs, the corresponding degenerated directions would be calculated, which further are utilized to construct prior constraints for degeneration-aware optimization.
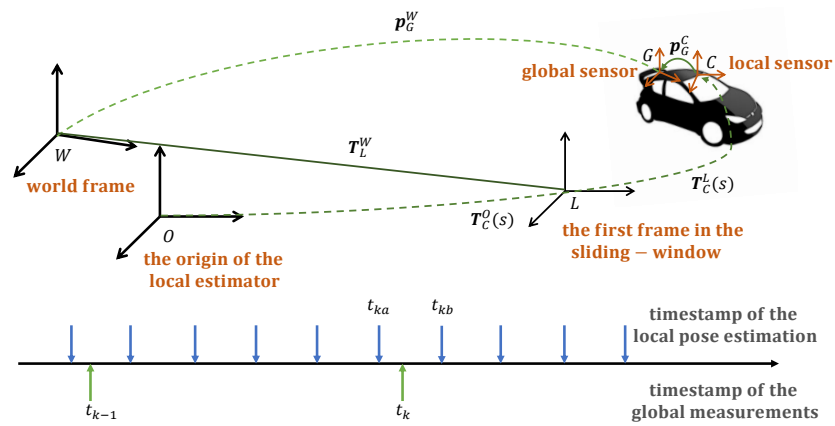

Figure 1. The demonstration of the frames and timeline utilized throughout the paper.
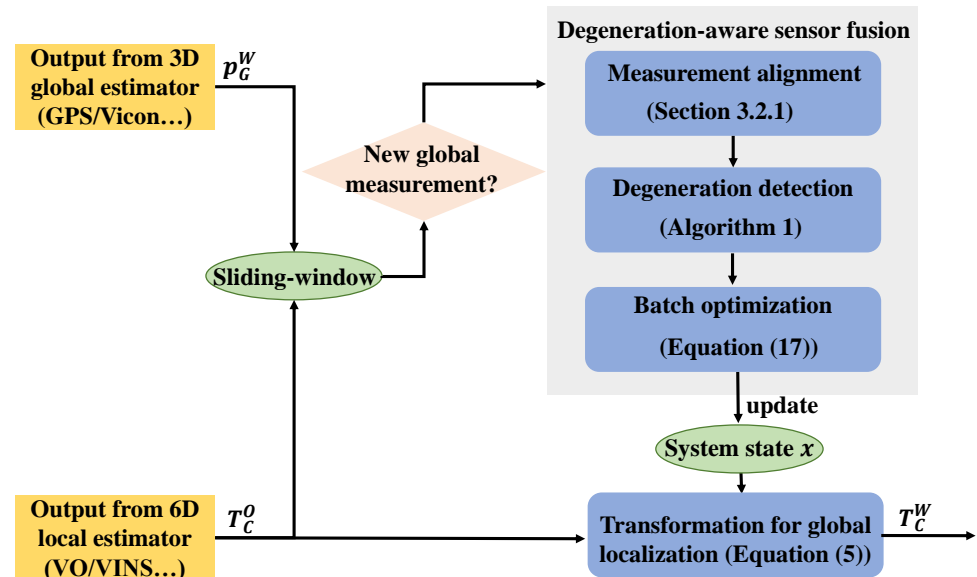

Figure 2. The pipeline of the proposed degeneration-aware sensor fusion framework.

3.2.1. Measurement Alignment

We first align the global and local measurements according to the timestamps. The global measurement $p_{G_k}^W$ received at timestamp t_k is aligned with two closest local mea-

surements at timestamp t_{ka} and t_{kb} as shown in Figure 1. We interpolate the corresponding local pose $T_{C_k}^O$ at the timestamp t_k according to

$$
\begin{aligned}
\lambda &= \frac{t_k - t_{ka}}{t_{kb} - t_{ka}} \\
q_{C_k}^O &= \text{slerp}(q_{C_{ka}}^O, q_{C_{kb}}^O, \lambda) \\
p_{C_k}^O &= (1 - \lambda) p_{C_{ka}}^O + \lambda p_{C_{kb}}^O
\end{aligned}
\tag{1}
$$

in which $\text{slerp}(\cdot)$ denotes the spherical linear interpolation function [26].

3.2.2. General Batch Optimization for Localization

After aligning the newly received global measurement, the system state $x = \{R_L^W, p_L^W, p_G^C, s\}$ is optimized according to the time-aligned measurements within the current sliding-window. The error function constructed by the measurements at timestamp t_k is formulated as

$$
e_{p_k} = \bar{p}_{G_k}^W - p_{G_k}^W
\tag{2}
$$

in which $\bar{p}_{G_k}^W$ is derived by

$$
\begin{aligned}
T_{C_k}^L &= \begin{bmatrix} R_{C_k}^L & p_{C_k}^L \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} = (T_L^O)^{-1} T_{C_k}^O \\
T_{G_k}^W &= \begin{bmatrix} R_{G_k}^W & p_{G_k}^W \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} = T_L^W T_{C_k}^L(s) T_G^C \\
&= \begin{bmatrix} R_L^W & p_L^W \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \begin{bmatrix} R_{C_k}^L & s p_{C_k}^L \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \begin{bmatrix} R_G^C & p_G^C \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \\
\bar{p}_{G_k}^W &= R_L^W R_{C_k}^L p_G^C + s R_L^W p_{C_k}^L + p_L^W \\
&= R_L^W (R_{C_k}^L p_G^C + s p_{C_k}^L) + p_L^W
\end{aligned}
\tag{3}
$$

Summing up all the valid error functions in the sliding-window, the cost function for the general localization based optimization is defined as

$$
f = \sum_k \rho(e_{p_k}^T \Omega_e e_{p_k})
\tag{4}
$$

in which Ω_e denotes the information matrix that can be derived as the inverse of the measurement covariance matrix. $\rho(\cdot)$ represents the Huber's robust kernel function [27] which can suppress the effect of outliers. In this work, we minimize this non-linear least-square cost function iteratively for system state optimization based on the Levenberg-Marquardt algorithm [28,29].

Given the latest optimized state x, we can transform the continuous output of the local estimator $T_{C_t}^L$ to the world frame for real-time 6D global localization. The 6D global localization at timestamp t can be calculated as

$$
T_{C_t}^W = \begin{bmatrix} R_L^W R_{C_t}^L & s R_L^W p_{C_t}^L + p_L^W \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}
\tag{5}
$$

3.3. Observability Analysis

For the optimization formulated in Equation (4), we first analyze its observability in the general occasion. We linearize Equation (2) at the current estimate to compute the Jacobian

$$
H_k = \frac{\partial e_{p_k}}{\partial x} = \begin{bmatrix} \Xi_k & I_{3\times 3} & R_L^W R_{C_k}^L & R_L^W p_{C_k}^L \end{bmatrix}_{3\times 10}
\tag{6}
$$

in which Ξ_k corresponds to the Jacobian of R_L^W on Manifold

$$\Xi_k = -R_L^W[(R_{C_k}^L p_G^C + s p_{C_k}^L) \times] \tag{7}$$

We denote $[a \times]$ as the 3×3 skew-symmetric for vector $a = [a_x, a_y, a_z]^T$, which is formulated as

$$[a \times] = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \tag{8}$$

As the system state x changes slowly across time, the corresponding state transition matrix $\Phi(t_0, t_k)$ between timestamps t_0 and t_k is identity. The observability matrix M can be constructed following [30]

$$M = \begin{bmatrix} \vdots \\ H_k \Phi(t_0, t_k) \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ H_k \\ \vdots \end{bmatrix} \tag{9}$$

As in general cases the columns in H_k are linear independent. Thus based on Equation (9) it can be noticed that in general cases the rank of M achieves 10 when $n \geq 4$. Therefore, when there are more than 4 global measurements in the sliding-window, the system is observable.

3.4. Degeneration Analysis

In this subsection, we summarize the common degenerated motion patterns in the robotics area and derive their corresponding unobservable directions.

3.4.1. Pure Translation along One Axis

This is a common motion pattern in autonomous driving. When the robot moves along a straight line, $R_{C_i}^L$ is invariant for any $i \in [m, m + n - 1]$, in which m is utilized to denote the beginning of the sliding-window. The translational measurement of the local estimator can be derived as

$$p_{C_i}^L = \alpha_i \bar{p}_{m_0}^L \tag{10}$$

in which $\bar{p}_{m_0}^L$ is the unit vector of the translational direction and α_i is a scalar variable that can be derived as $\alpha_i = \|p_{C_i}^L\| / \|\bar{p}_{m_0}^L\|$. Thus, in H_i the third block is constant, and the first two blocks are also linear dependent. The right null-space matrix N of the observability matrix M can be derived as

$$N = \begin{bmatrix} 0_{3 \times 3} & \bar{p}_{m_0}^L \\ -R_L^W R_{C_i}^L & R_L^W [R_{C_i}^L p_G^C \times] \bar{p}_{m_0}^L \\ I_{3 \times 3} & 0_{3 \times 1} \\ 0 & 0 \end{bmatrix}_{10 \times 4} \tag{11}$$

$MN = 0$ can be verified as $H_i N = 0$. Therefore, in this occasion, there is a four-dimensional unobservable subspace which is constructed as span(N). The formation of N shows that the first three column vectors are related to the translational components $\{p_L^W, p_G^C\}$, while the last column vector is related to one direction of R_L^W which is parallel to the translational direction $\bar{p}_{m_0}^L$.

3.4.2. Random 3D Translation

This would occur when the robot performs pure translation in 2D/3D space without rotating. In this case, Equation (10) is violated while $R_{C_i}^L$ keeps invariant, which decreases the dimension of N to 3

$$N = \begin{bmatrix} \mathbf{0}_{3\times3} \\ -R_L^W R_{C_i}^L \\ I_{3\times3} \\ 0 \end{bmatrix}_{10\times3} \tag{12}$$

We can find that R_L^W is observable now, and there remain three unobservable directions that are related to the translational components $\{p_L^W, p_G^C\}$.

3.4.3. One-Axis Rotation

When the robot turns along a fixed rotation axis $\boldsymbol{\omega} \in \mathbb{R}^3$ with non-zero translational velocity, only the translational components along $\boldsymbol{\omega}$ remain unobservable. In this case, $R_{C_i}^L$ can be derived according to the Rodrigues' rotation formula

$$R_{C_i}^L = R_{C_m}^L (I_{3\times3} + \sin(\theta_i)[\boldsymbol{\omega}\times] + (1 - \cos(\theta_i))[\boldsymbol{\omega}\times]^2) \tag{13}$$

in which θ_i denotes the rotation angle of $R_{C_i}^{C_m}$. As $[\boldsymbol{\omega}\times]\boldsymbol{\omega} = \mathbf{0}_{3\times1}$, we can derive that multiplying the third block of H_i by $\boldsymbol{\omega}$ leads to

$$R_L^W R_{C_i}^L \boldsymbol{\omega} = R_L^W R_{C_m}^L (I_{3\times3} + \sin(\theta_i)[\boldsymbol{\omega}\times] + (1 - \cos(\theta_i))[\boldsymbol{\omega}\times]^2)\boldsymbol{\omega} = R_L^W R_{C_m}^L \boldsymbol{\omega} \tag{14}$$

The result is constant for any $i \in [m, m+n-1]$. Thus, the null-space matrix that satisfies $MN = \mathbf{0}$ can be derived as

$$N = \begin{bmatrix} \mathbf{0}_{1\times3} & -(R_L^W R_{C_m}^L \boldsymbol{\omega})^T & \boldsymbol{\omega}^T & 0 \end{bmatrix}^T_{10\times1} \tag{15}$$

It should be noticed that when the robot performs uniform motion, more dimensions maybe unobservable. One case is that, when both the angular velocity and the translational velocity are constant, the dimension of the unobservable subspace is expanded to 2 as one direction of R_L^W that is along with the rotational axis would be undistinguishable. Another example is that, when the directions of the constant angular and translational velocity are perpendicular, the scale of the local estimator would be unobservable. These are likely to happen when the robot performs planar rotational movement, such as a car turns with constant velocity in autonomous driving. Detailed proof of the two claims are derived in the Appendix A.

3.5. Degeneration-Aware Batch Optimization

When there exists an unobservable subspace in the system, the related states are unconstrained and susceptible to measurement noise. To address this problem, we first detect the degenerated directions before the optimization process, then add prior constraints specifically along these directions.

Similarly to [14], we calculate the eigenvalues $\{\lambda_l\}$ and the corresponding eigenvectors $\{v_l\}$ by performing SVD decomposition on the Hessian matrix $H^T H$, in which $H = [H_m^T \cdots H_{m+n-1}^T]^T$ is linearized at the latest estimated state before each optimization and $l = 0 \cdots 9$. We assume that $\{\lambda_l\}$ is sorted and λ_9 corresponds to the smallest eigenvalue. Theoretically the eigenvalues corresponding to the unobservable dimensions are precisely zero, while in practice they are not because of the system noise. It is apparent that a constant threshold ϵ is not sufficient to distinguish zero eigenvalues for all kinds of systems, calling for more flexible construction of the threshold. We also quantitatively analyze the influence of noise on eigenvalues in Section 4.1.

We propose a heuristic method to detect the unobservable subspace as shown in Algorithm 1. The absolute thresholds ϵ_a and ϵ_b define the coarse but valid upper and lower

bounds distinguishing the unobservable and observable subspaces. For the eigenvalue that distributes between the two bounds, it is difficult to directly categorize it by a threshold due to the unrestricted range of noise. We compute its ratio with the adjacent eigenvalues to indicate whether they have the same type of observability. If the ratio is larger than threshold ϵ_r, we assume that the adjacent eigenvalues have the same type of observability. As the method traverses from the smallest eigenvalue, the eigenvalue is categorized as relating to the unobservable subspace if it shares the same type of observability with its previous one.

Algorithm 1: Detecting unobservable directions.

    Data: $\epsilon_a, \epsilon_b, \epsilon_r, \{e_i\}$ ($i = m, \cdots, m+n-1$, The set of error functions in the sliding-window)
    Result: V_u (The set of unobservable directions)
1 Linearize $\{e_k\}$ to get H;
2 $\{\lambda_l, v_l\} \leftarrow \text{SVD}(H^T H), l = 0 \cdots 9$;
3 $l = 9, V_u = \{\}$;
4 **while** $l > 0$ **do**
5   **if** $\lambda_l > \epsilon_a$ **then**
6   **break**;
7   **if** $\lambda_l < \epsilon_b$ **then**
8 $V_u \leftarrow \{V_u, v_l\}$; $l--$; **continue**;
9 $\alpha_{post} = \lambda_l / \lambda_{l-1}$;
10   **if** $l==9$ **then**
11   **if** $\alpha_{post} < \epsilon_r$ **then**
12 $V_u \leftarrow \{V_u, v_l\}$
13   **break**
14 $\alpha_{pre} = \lambda_{l+1} / \lambda_l$;
15   **if** $\alpha_{pre} > \epsilon_r$ **then**
16 $V_u \leftarrow \{V_u, v_l\}$; $l--$;
17   **else**
18   **if** $\alpha_{pre} < \epsilon_r$ *and* $\alpha_{post} < \epsilon_r$ **then**
19 $V_u \leftarrow \{V_u, v_l\}$;
20   **break**;

After detecting out the degenerated subspace with dimension r, we collect the corresponding eigenvectors $V_u = \{v_0, \cdots, v_{r-1}\}$ which denote the degenerated directions. Instead of projecting the incremental update to the observable directions during optimization, we restrict the update in unobservable directions by adding constraints

$$
\begin{aligned}
e_{R_j} &= \log((\hat{R}_L^W)^T R_L^W) v_{j(0:3)} \\
e_{p_{Lj}^W} &= (\hat{R}_L^W)^T (p_L^W - \hat{p}_L^W) v_{j(3:6)} \\
e_{e_j} &= (\hat{p}_G^C - p_G^C) v_{j(6:9)} \\
e_{s_j} &= (\hat{s} - s) v_{j(9)}
\end{aligned}
\tag{16}
$$

in which $j \in [0, r-1]$. $e_{R_j}, e_{p_{Lj}^W}, e_{e_j}$ and e_{s_j} denote the constraints applied on R_L^W, p_L^W, p_G^C and s, respectively. $(\hat{\cdot})$ denotes the related state estimation before optimization. $v_{j(a:b)}$ represents the vector that is compromised with the ath to bth elements of v_j. These error terms can be considered as priors to prevent the states from drifting in the unobservable subspace, and are only added for optimization when $\|v_{j(a:b)}\| > 0.1$, which means that

the prior constraint on this state is activated. Finally the cost function in the proposed degeneration-aware batch optimization is

$$f = \sum_{i=m}^{m+n-1} \rho(e_{p_i}^T \Omega_e e_{p_i}) + \sum_{j \in S_R} e_{R_j}^T \Omega_p e_{R_j} + \sum_{j \in S_{p_L^W}} e_{p_{Lj}^W}^T \Omega_p e_{p_{Lj}^W}$$
$$+ \sum_{j \in S_e} e_{e_j}^T \Omega_p e_{e_j} + \sum_{j \in S_s} e_{s_j} \Omega_{ps} e_{s_j} \tag{17}$$

in which $S_R, S_{p_L^W}, S_e$ and S_s denote the sets of activated prior constraints on the system states, respectively. Ω_p and Ω_{ps} are the information matrices of the prior constraints.

4. Experimental Results

We validate the effectiveness of our degeneration-aware sensor fusion method through both simulated and real-world datasets. In simulation experiments, we first show the sensitivity of the eigenvalues with respect to the measurement noise. Then we create four sequences of simulated trajectories that correspond to the common degenerated motion patterns analyzed in Section 3.4, and utilize them to demonstrate the effect of degeneration on state estimation and the effectiveness of our proposed method in terms of preventing the state from drifting along the degenerated directions. In real-world experiments, we show the localization performance on indoor EuRoC [31] and outdoor KAIST [32] datasets. EuRoC is a visual-inertial dataset collected by a micro aerial vehicle that performs 6D movements. We utilize the 3D Leica MS50 laser tracker as the 3D global estimator and only utilize the left images to perform monocular visual odometry, which serves as the local pose estimator with no access to the absolute scale. KAIST dataset is collected in outdoor urban environment on a car and we utilize the GPS as global estimator and stereo-IMU information to perform local visual-inertial odometry. During the experiments, we set the length of the sliding-window as 10 m for indoor dataset and 50 m for outdoor datasets. All the experiments are executed on a laptop with a 2.70 GHz Intel Core i7-7500U CPU. The codes are implemented in C++ and the optimization process is implemented with Levenberg-Marquardt algorithms in g2o [33].

In the simulation experiments, different motion patterns are controlled by setting different translational and rotational velocities for the local sensor. The minimum sampling interval is 0.1 s, and the ground truth of the trajectories can be integrated based on the velocities. The trajectories of the global sensor can be calculated based on the ground truth of the system state using Equation (3). The global measurements are simulated by corrupted the ground truth of the trajectories with Gaussian noise. To simulate the drift of the local estimator, we add Gaussian noise on the relative pose in each sampling interval, and integrate the corrupted relative pose to formulate the local measurements. As the ground truth of the system state can be accessed in the simulation experiments, the performances of different methods are evaluated by comparing the state estimation errors.

In real-world experiments, we evaluate the performances of different methods by the absolute trajectory error (ATE) and rotational error [34]. For each global pose $T_{C_t}^W$ estimated at timestamp t, we find its corresponding ground truth pose $T_{C_t}^{W*}$ provided by the datasets according to the timestamp. The localization error is derived as
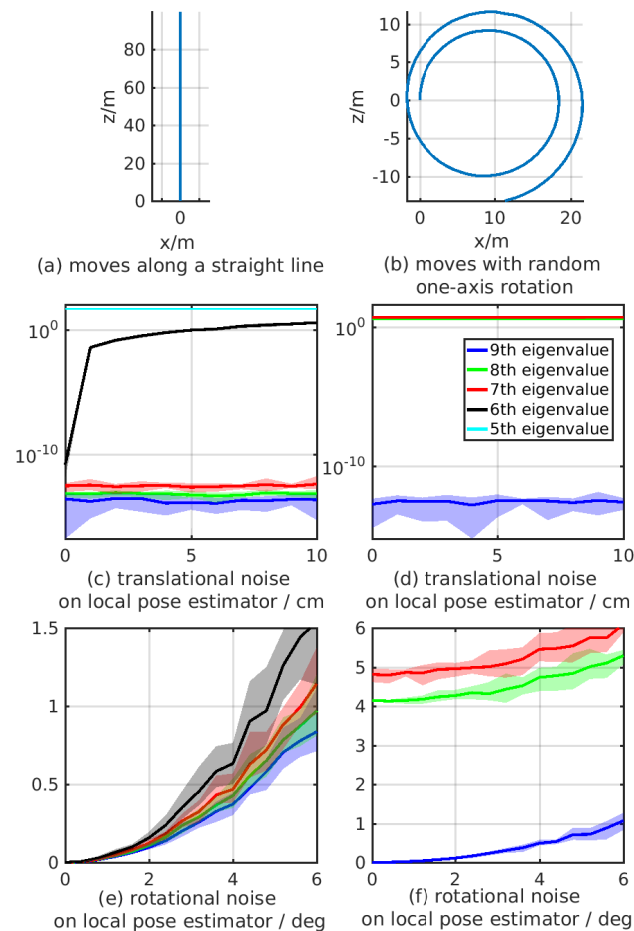
$$E_t = (T_{C_t}^{W*})^{-1} T_{C_t}^W \tag{18}$$

The ATE error relates to the norm of the translational component in E_t, and the rotational error relates to the rotational component in E_t.

4.1. Sensitivity of the Eigenvalues

To quantitatively demonstrate the influence of the noise on the eigenvalues, we create simulated trajectories that are related to two types of the degenerated motion patterns shown on the first row of Figure 3. We add Gaussian noise on the simulated measurement

and show the changing of the related eigenvalues. As the derivation of eigenvalues relates to the measurements of the local estimator, we, respectively, add Gaussian noises with standard deviations from 0 to 10 cm on the translational parts of the local measurements, and add Gaussian noises with standard deviations from 0 to 6 deg on the rotational parts. To avoid the influence of randomness, for each standard deviation we generate 200 trajectories for eigenvalue evaluation. Figure 3a shows the trajectory generated when the robot moves along a straight line and the eigenvalues drawn in Figure 3c,e are influenced by translational and rotational noises on the local estimators with different standard deviations, respectively. To keep the scale of the axis for better demonstration, we do not draw the value of the 5th eigenvalue influenced by the rotational noise in Figure 3e, which are around the same values as shown in the results that influenced by the translational noise. The right column includes the results tested when the robot moves with random one-axis rotation. The trajectory is shown in Figure 3b. The eigenvalues drawn in Figure 3d,f are influenced by the translational and rotational Gaussian noises, respectively. The curves in each figures denote the means of the eigenvalues and the colored region surrounding the curve represents the statistic of 200 trials. Some colored regions are hard to be distinguished as they are too close to the curve in the related axis scale.

Figure 3. Eigenvalues of the system calculated on two degenerated trajectories with different measurement noises. The curves in each figures denote the means of the eigenvalues and the colored region surrounding the curve represents the statistic of 200 trials.
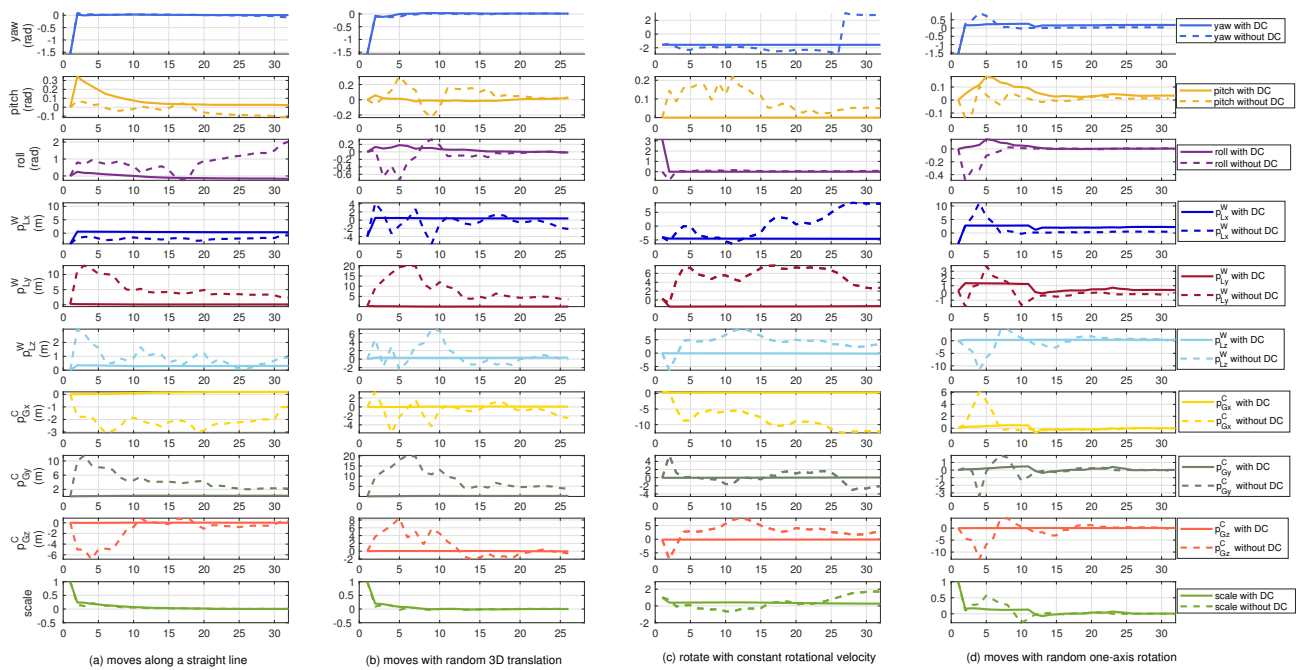
Based on our analysis in Section 3.4, when the robot moves along a straight line, there should be four unobservable directions, which means that the smallest four eigenvalues should be close to zero. When the robot moves with non-constant one-axis rotation, there should be only one unobservable direction. We, respectively, draw the smallest

five and three system eigenvalues tested on the two trajectories in the two columns of Figure 3. The results show that when the noise is small, the eigenvalues corresponding to the unobservable subspace are close to zeros (i.e., the 6–9th eigenvalues on the first column and the 9th eigenvalue on the second column), and become larger when the noise grows. Although the eigenvalues corresponding to the observable subspace are generally large. Thus we could set the coarse upper bound ϵ_a and lower bound ϵ_b to distinguish eigenvalues. However, we can notice that the 6th eigenvalue on the left column is close to the 7th and 8th eigenvalues on the right column. If we utilize one constant threshold, it is difficult to distinguish the unobservable and observable subspaces in this situation. In our proposed detection method, the difference between the adjacent eigenvalues is taken into consideration, which contributes to handle this problem. Based on the analysis of these results, we set $\epsilon_a = 5, \epsilon_b = 10^{-2}, \epsilon_r = 0.1$ throughout the left of the experiments.

4.2. Simulation Results

In this subsection, we utilize simulated data collected under degenerated movement patterns to perform ablation study on the effectiveness of the proposed degeneration-aware sensor fusion. Specifically, we evaluate the accuracy of the estimated system state optimized with general optimization formulated in Equation (4) and the optimization with the degeneration-aware constraints (DC) in Equation (17), respectively. The simulated data includes four types of trajectories in which the robot moves (a) along a straight line, (b) with random 3D translation, (c) with perpendicular constant rotational and translational velocity, and (d) with random one-axis rotation. The motion patterns are controlled by the translational and rotational velocities. We set the sampling time as 0.1 s and the trajectories are integrated based on the velocity. We add zero-mean Gaussian noise on the relative pose within each sampling time to simulate the measurement noise of the local estimator and integrate the relative poses to simulate the local measurements. The 3D global measurements are collected every 1 s along the ground truth of each trajectory and are also corrupted by the noise. The standard deviations of the rotational and translational noises for the local estimator are 0.05 rad and 0.05 m. The standard deviation of the translational noise for the global estimator is 0.5 m.

The estimation errors for the system state x are drawn in Figure 4 and we represent the rotational error of R_L^W in Euler Angle parameterization denoting as "yaw", "pitch", and "roll" in the first three rows of the figure. Based on Equation (11), when the robot moves along a straight line, the dimension of the unobservable subspace is four, which relates to one rotational axis of R_L^W along the moving direction and 3D translational directions that are coupled in p_L^W and p_G^C. These can be reflected from the results in the first column of Figure 4 indicated by dashed lines which do not converge in the related dimensions. On the other hand, the results optimized with the proposed degeneration-aware constraints, though could not converge to the ground truth in the unobservable directions as no valid information is provided, do not diverge during the whole process. The results in the second to fourth columns also validate our analysis in Section 3.4. Additionally, it is interesting to notice that, as one direction of the unobservable space might be related to several state variables, r dimensions of unobservable directions would cause the divergence of more than r state variables.

Figure 4. The evaluated state estimation errors tested on the simulation dataset. From top to the bottom the figures in each row present the estimation errors on each element of the system state x. The x-axis denotes the time line with the unit of second.
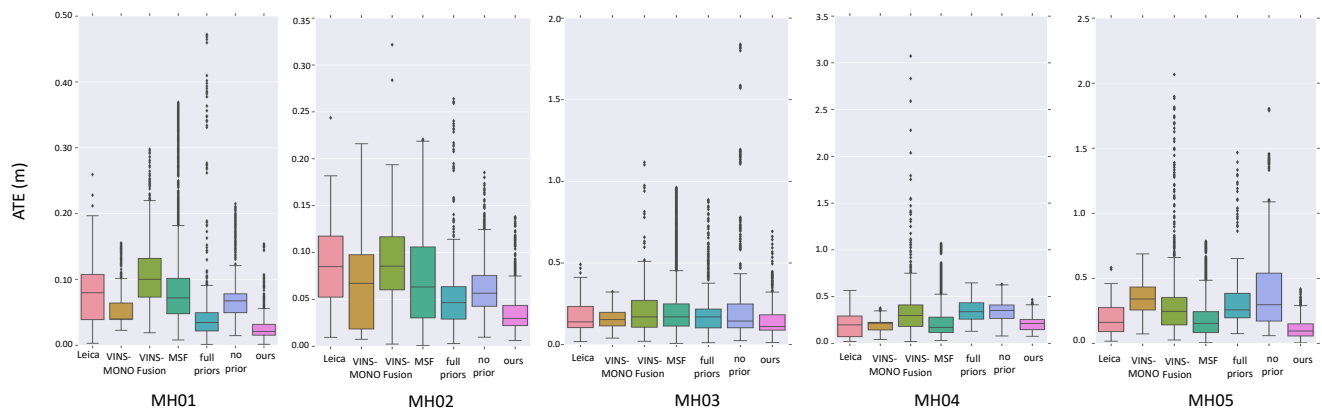
4.3. Real World Experiments

We evaluate the global localization performance of our proposed method on sequences MH01–MH05 in EuRoC datasets and sequence Urban 38 in KAIST dataset.We implement the monocular visual odometry based on ORB-SLAM [1] and the stereo visual inertial odometry based on Openvins [3]. We utilize the absolute trajectory error (ATE) [34] to evaluate the global localization performance and compare our results with the open-sourced multi-sensor fusion methods MSF [8] and VINS-Fusion [10].

4.3.1. EuRoC Dataset

In EuRoC dataset we utilize monocular vision to perform local pose estimation. The output of the monocular visual odometry does not include the absolute scale, which can not be utilized for global localization in many other global–local fusion methods [7,9,10]. Thus, to test the performance of VINS-Fusion [10], we change its local estimator into VINS-MONO [2], which is the classical monocular VIO provided by the VINS-Fusion system.

We draw ATE errors evaluated by different methods in Figure 5, and list the rotational errors in Table 2. The results of VINS-MONO is evaluated by aligning the trajectories to the ground truth. The results show that our method could successfully fuse 3D global positions and 6D scaleless local poses for global localization, and outperform the compared methods in terms of both translational and rotational localization accuracy. We can notice that in many sequences VINS-MONO also performances well as the trajectories are not long, but the fusion results in VINS-Fusion show larger errors as the extrinsic parameter is not estimated in VINS-Fusion. Besides, compared with MSF and VINS-Fusion, our method does not require IMU data to support sensor fusion, which further demonstrates its efficiency and versatility for sensor fusion based global localization.

Figure 5. The average trajectory errors of different localizers tested on EuRoC dataset.

We also perform ablation study to evaluate the effectiveness of the degeneration-aware optimization. We first evaluate the localization performance tested on the general batch optimization formulated in Equation (4). The results are denoted as "no prior" in Figure 5. Besides, to validate the benefit of adding prior constraints specifically on the degenerated directions, we also test the localization performance by adding the full prior constraints for the whole system state. The results are denoted as "full priors" in Figure 5. From the results we can infer that, the degeneration constraints can successfully improve the localization accuracy as the influence from the noise in degenerated directions is restrained. Besides, adding constraints especially on degenerated directions can also improve the localization accuracy as the observable elements in the system state could be optimized directly without influenced by heuristic constraints which might be inaccurate.

Table 2. Median rotational errors (deg) tested on EuRoC datasets.

	MH01	**MH02**	**MH03**	**MH04**	**MH05**
ours	1.05	0.63	1.23	1.57	0.91
full priors	1.51	1.01	1.52	4.98	3.26
no prior	1.80	0.75	2.70	2.12	1.96
VINS-Fusion	1.42	1.13	2.35	2.56	2.52
MSF	1.40	1.63	3.15	3.29	3.73

4.3.2. KAIST Dataset

In KAIST dataset, we utilize the GPS as the global estimator, which is prone to drift when the car stops at the crossroads. Additionally, the GPS signal is inaccurate when the covariance is large. Therefore, during online localization, we drop the GPS messages with large covariance (the threshold is set as 60) and the drifted messages indicating that the car is moving while the local estimator denoting that the car does not move.

We list the localization performance of GPS and the stereo-IMU version of Openvins [3] in Table 3 as baseline, and also compare our results with MSF [8], VINS-Fusion[10], and full priors aided results. The translational and heading errors are listed in Tables 3 and 4. The corresponding trajectories are drawn in Figure 6. The dropped GPS signals are also marked in the trajectory.
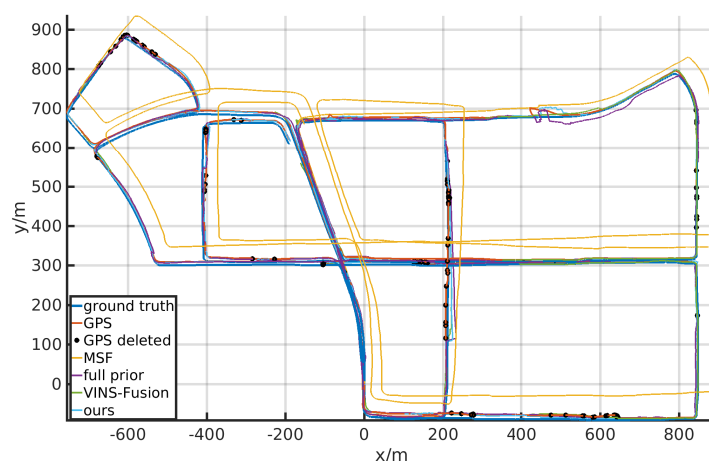
Table 3. ATE error (m) tested on KAIST dataset.

	GPS	**Openvins**	**MSF**	**VINS-Fusion**	**Full Priors**	**No Prior**	**Ours**
mean	5.98	14.43	21.14	9.64	9.20	/	5.58
rmse	7.15	16.46	22.09	11.11	12.21	/	6.71
std	3.91	7.91	14.47	5.52	6.40	/	3.73

Table 4. Median heading errors (deg) tested on KAIST dataset.
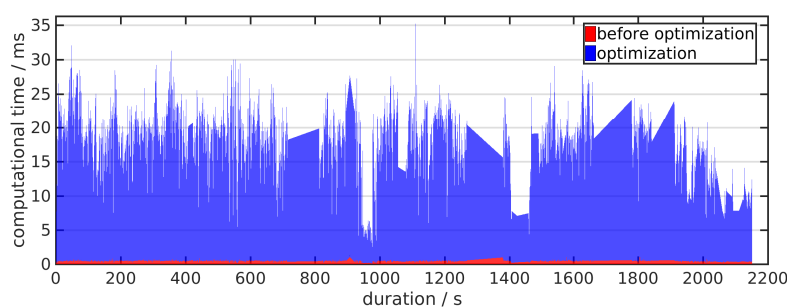
MSF	Full Priors	VINS-Fusion	Ours
12.8	1.48	1.56	1.43

Figure 6. The localization trajectories of different localizers tested on Urban 38 of the KAIST dataset.

Compared with EuRoC datasets, the robot performs planar movements in Kaist dataset, in which the degeneration would likely occur. Thus, the results of "no prior" are largely influenced by the degeneration and fail to perform the whole localization. The results also show that our method achieves better performance than the compared methods and could provide drift-free global localization results.

4.4. Computational Cost Evaluation

To validate the ability of achieving real-time localization, we evaluate the computational time cost on the multi-sensor fusion process. The computational time is defined as the duration between the time at which a new global measurement is received and the time the system state is updated by the optimization results, which is independent with the processes in the front-ends. We collect the computational time tested on the KAIST dataset and draw the results in Figure 7, in which we specifically color the the time cost on "measurement alignment" and "degeneration detection" as red. As the fusion process is taken place within a sliding-window, the computational time would not increase with time. The average processing time is 18.4 ms, which satisfies the requirement of achieving real-time global–local sensor fusion. Besides, the average time of "before optimization" is 0.45 ms, which demonstrates that our solution for the degeneration motion would not introduce large computational burden towards the fusion process.

Figure 7. Computational time of the fusion process tested on KAIST dataset. The red component denotes the time cost on "measurement alignment" and "degeneration detection". Although the blue component denotes the time cost on degeneration-aware batch optimization.

5. Conclusions

To achieve drift-free 6D global localization, in this paper we propose a general optimization-based framework that fuses the 3D intermittent global positioning information and 6D local odometry for real-time global pose estimation. The system state is specifically formulated to be fully observable under general motions. To address the impact from generated motion patterns, we propose a degeneration-aware solution to robustly detect the degenerated directions that are further utilized as indicators to add prior constraints. Though extra time is cost for degeneration detection, the whole processing time could satisfy the requirement of real-time sensor fusion. The effectiveness of our proposed method is validated on both simulated and real-world dataset. In the future work, we set to extend the configuration of our framework from 3D–6D global–local combination to multiple types of pose estimators with general degree of freedom, and utilize the multi-sensor information to further detect the failure in the local pose estimators for outlier rejection.

Abbreviations

The following abbreviations are used in this manuscript:

3/6D	3/6 Degree-of-Freedom
DC	Degeneration Constraint
EKF	Extended Kalman Filter
GPS	Global Positioning System
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
LIDAR	Light Detection and Ranging
MSF	Mult-Sensor Fusion
UWB	Ultra Wide Band
VIO	Visual Inertial Odometry

Appendix A. Derivation of the Degeneration Analysis on One-Axis Rotation with Constant Velocity

As the rotational velocity $\boldsymbol{\omega}$ and translational velocity $\boldsymbol{v}$ are constant in this motion pattern, the robot can be considered as moving along a spiral curve or a circle. In the following, we denote that for a 3D vector $\boldsymbol{a}$, it can be decomposed as $\boldsymbol{a} = \boldsymbol{a}_{\omega\times} + \boldsymbol{a}_{\omega}$, in which $\boldsymbol{a}_{\omega\times}$ represents the component that is perpendicular to $\boldsymbol{\omega}$ and $\boldsymbol{a}_{\omega}$ represents the component that is parallel to $\boldsymbol{a}$. As $[\boldsymbol{a}\times]\boldsymbol{\omega} = [\boldsymbol{a}_{\omega\times}\times]\boldsymbol{\omega}$, we can derive $\boldsymbol{\Xi}_k\boldsymbol{\omega}$ as

$$
\begin{aligned}
\boldsymbol{\Xi}_k\boldsymbol{\omega} &= -\boldsymbol{R}_L^W[(\boldsymbol{R}_{C_k}^L\boldsymbol{p}_G^C + s\boldsymbol{p}_{C_k}^L)\times]\boldsymbol{\omega} \\
&= -\boldsymbol{R}_L^W\left[(\boldsymbol{R}_{C_k}^L\boldsymbol{p}_{G\omega\times}^C + s\boldsymbol{p}_{C_k\omega\times}^L)\times\right]\boldsymbol{\omega} \\
&= -\boldsymbol{R}_L^W[\boldsymbol{p}_{G_k\omega\times}^L\times]\boldsymbol{\omega}
\end{aligned} \tag{A1}
$$

in which we set $\boldsymbol{p}^L_{G_k\omega\times} = \boldsymbol{R}^L_{C_k}\boldsymbol{p}^C_{G\omega\times} + s\boldsymbol{p}^L_{C_k\omega\times}$. We draw the projected trajectory in Figure A1b, which should be a circle with the radius ρ. As the following derivation only focus on the variables on the projected trajectory, we reuse the notations of L, C, G to represent their projected variables. For the sake of the derivation, we define a new reference frame R. The position of R is defined as the center of the projected circle, and we set its orientation the same as the reference frame L. As $\|\boldsymbol{p}^R_{RC_k}\| = \|\boldsymbol{p}^R_{RL}\|$, we can transform $\boldsymbol{p}^R_{RC_k}$ to $\boldsymbol{R}^L_{C_k}\boldsymbol{p}^R_{RL}$. $\boldsymbol{p}^L_{G_k\times}$ can be decomposed as

$$
\begin{aligned}
\boldsymbol{p}^L_{G_k\omega\times} &= \boldsymbol{R}^L_R(\boldsymbol{p}^R_{LR} + \boldsymbol{p}^R_{RC_k} + \boldsymbol{p}^R_{C_kG}) \\
&= \boldsymbol{R}^L_R(\boldsymbol{p}^R_{LR} + \boldsymbol{R}^L_{C_k}\boldsymbol{p}^R_{RL}) + \boldsymbol{R}^L_{C_k}\boldsymbol{p}^C_G
\end{aligned}
\tag{A2}
$$

in which $\boldsymbol{p}^A_{BD}$ denotes the vector from frame B to frame D representing in the coordinate of A. As $\boldsymbol{R}^L_R = \boldsymbol{I}_{3\times3}$, Equation (A2) equals to

$$
\boldsymbol{p}^L_{G_k\omega\times} = \boldsymbol{p}^R_{LR} + \boldsymbol{R}^L_{C_k}(\boldsymbol{p}^R_{RL} + \boldsymbol{p}^C_G)
\tag{A3}
$$

Thus, Equation (A1) can be derived as

$$
\boldsymbol{\Xi}_k\boldsymbol{\omega} = -\boldsymbol{R}^W_L([\boldsymbol{p}^R_{LR}\times]\boldsymbol{\omega} + [\boldsymbol{R}^L_{C_k}(\boldsymbol{p}^R_{RL} + \boldsymbol{p}^C_G)\times]\boldsymbol{\omega})
\tag{A4}
$$

As $\boldsymbol{R}^L_{C_k}$ rotates along $\boldsymbol{\omega}$, we can achieve that $(\boldsymbol{R}^L_{C_k})^T\boldsymbol{\omega} = \boldsymbol{\omega}$, which leads to

$$
\begin{aligned}
[\boldsymbol{R}^L_{C_k}(\boldsymbol{p}^R_{RL} + \boldsymbol{p}^C_G)\times]\boldsymbol{\omega} &= \boldsymbol{R}^L_{C_k}[\boldsymbol{p}^R_{RL} + \boldsymbol{p}^C_G\times](\boldsymbol{R}^L_{C_k})^T\boldsymbol{\omega} \\
&= \boldsymbol{R}^L_{C_k}[\boldsymbol{p}^R_{RL} + \boldsymbol{p}^C_G\times]\boldsymbol{\omega}
\end{aligned}
\tag{A5}
$$

Thus besides Equation (15), the other vector that compromises the null-space matrix is

$$
\boldsymbol{N} = [\boldsymbol{\omega}^T \quad (\boldsymbol{R}^W_L[\boldsymbol{p}^R_{LR}\times]\boldsymbol{\omega})^T \quad ([\boldsymbol{p}^R_{RL} + \boldsymbol{p}^C_G\times]\boldsymbol{\omega})^T \quad 0]^T_{10\times1}
\tag{A6}
$$

which demonstrates that besides Equation (15), there exists one dimension of state $\boldsymbol{R}^W_L$ which is along with the rotational axis $\boldsymbol{\omega}$ becoming unobservable.

If the rotational and translational velocity is perpendicular, the trajectory is equal to a circle which we also demonstrate in Figure A1b. We can decompose $\boldsymbol{p}^L_{C_k}$ into
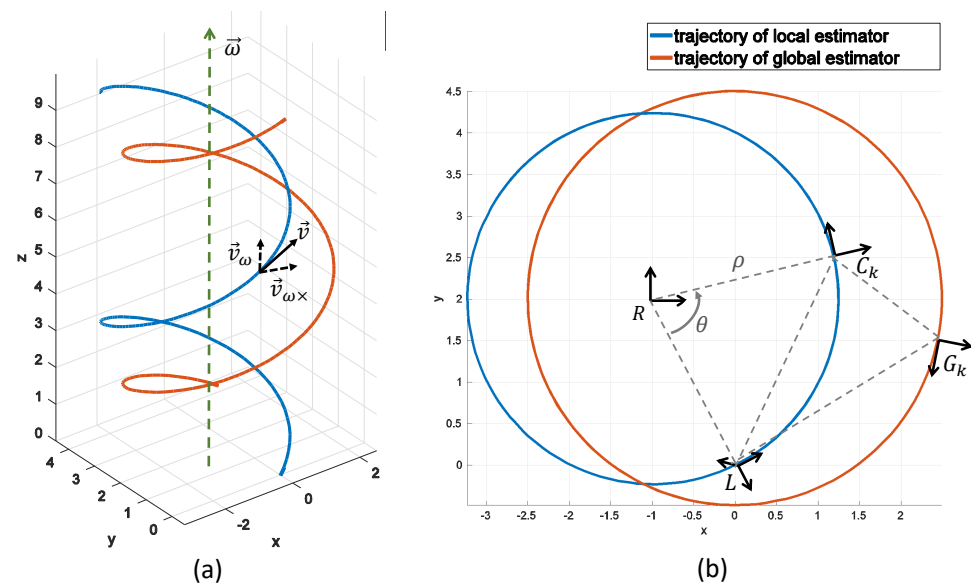
$$
\begin{aligned}
\boldsymbol{p}^L_{C_k} &= \boldsymbol{p}^L_{LR} + \boldsymbol{p}^L_{RC_k} = \boldsymbol{p}^L_{LR} + \boldsymbol{p}^R_{RC_k} \\
&= \boldsymbol{p}^L_{LR} + \boldsymbol{R}^L_{C_k}\boldsymbol{p}^R_{RL}
\end{aligned}
\tag{A7}
$$

Thus, under this circumstance, the null-space matrix is further expanded with

$$
\boldsymbol{N} = [\boldsymbol{0}_{1\times3} \quad (-\boldsymbol{R}^W_L\boldsymbol{p}^L_{LR})^T \quad -(\boldsymbol{p}^R_{RL})^T \quad 1]^T_{10\times1}
\tag{A8}
$$

which means the scale is degenerated to be unobservable.

This completes the proof.

Figure A1. This figure demonstrates (**a**) the 3D trajectory of one-axis rotation movement with constant angular and translational velocity and (**b**) its projected trajectory viewed along the rotational axis.

References

1. Mur-Artal, R.; Montiel, J.M.M.; Tardos, J.D. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans. Robot.* **2015**, *31*, 1147–1163. [CrossRef]
2. Qin, T.; Li, P.; Shen, S. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Trans. Robot.* **2018**, *34*, 1004–1020. [CrossRef]
3. Geneva, P.; Eckenhoff, K.; Lee, W.; Yang, Y.; Huang, G. Openvins: A research platform for visual-inertial estimation. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Virtual Conference, 31 May–31 August 2020; pp. 4666–4672.
4. Zhang, J.; Singh, S. LOAM: Lidar Odometry and Mapping in Real-time. *Robot. Sci. Syst.* **2014**, *2*, 9.
5. Zhang, C.; Chen, L.; Yuan, S. ST-VIO: Visual-Inertial Odometry Combined With Image Segmentation and Tracking. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 8562–8570. [CrossRef]
6. Lee, W.; Eckenhoff, K.; Geneva, P.; Huang, G. Intermittent gps-aided vio: Online initialization and calibration. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Virtual Conference, 31 May–31 August 2020; pp. 5724–5731.
7. Cioffi, G.; Scaramuzza, D. Tightly-coupled Fusion of Global Positional Measurements in Optimization-based Visual-Inertial Odometry. *arXiv* **2020**, arXiv:2003.04159.
8. Lynen, S.; Achtelik, M.W.; Weiss, S.; Chli, M.; Siegwart, R. A robust and modular multi-sensor fusion approach applied to mav navigation. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 3923–3929.
9. Mascaro, R.; Teixeira, L.; Hinzmann, T.; Siegwart, R.; Chli, M. GOMSF: Graph-Optimization based Multi-Sensor Fusion for robust UAV pose estimation. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–26 May 2018; pp. 1421–1428.
10. Qin, T.; Cao, S.; Pan, J.; Shen, S. A general optimization-based framework for global pose estimation with multiple sensors. *arXiv* **2019**, arXiv:1901.03642.
11. Yang, Y.; Geneva, P.; Eckenhoff, K.; Huang, G. Degenerate motion analysis for aided ins with online spatial and temporal sensor calibration. *IEEE Robot. Autom. Lett.* **2019**, *4*, 2070–2077. [CrossRef]
12. Jones, E.S.; Soatto, S. Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. *Int. J. Robot. Res.* **2011**, *30*, 407–430. [CrossRef]
13. Fu, B.; Wang, Y.; Ding, X.; Jiao, Y.; Tang, L.; Xiong, R. LiDAR-camera calibration under arbitrary configurations: Observability and methods. *IEEE Trans. Instrum. Meas.* **2019**, *69*, 3089–3102. [CrossRef]
14. Zhang, J.; Kaess, M.; Singh, S. On degeneracy of optimization-based state estimation problems. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–20 May 2016; pp. 809–816.
15. Maye, J.; Furgale, P.; Siegwart, R. Self-supervised calibration for robotic systems. In Proceedings of the 2013 IEEE Intelligent Vehicles Symposium (IV), Gold Coast City, Australia, 23–26 June 2013; pp. 473–480.

16. Weiss, S.; Achtelik, M.W.; Chli, M.; Siegwart, R. Versatile distributed pose estimation and sensor self-calibration for an autonomous MAV. In Proceedings of the 2012 IEEE International Conference on Robotics and Automation (ICRA), St. Paul, MN, USA, 14–18 May 2012; pp. 31–38.
17. Yang, N.; Tian, W.F.; Jin, Z.H.; Zhang, C.B. Particle filter for sensor fusion in a land vehicle navigation system. *Meas. Sci. Technol.* **2005**, *16*, 677. [CrossRef]
18. Suhr, J.K.; Jang, J.; Min, D.; Jung, H.G. Sensor fusion-based low-cost vehicle localization system for complex urban environments. *IEEE Trans. Intell. Transp. Syst.* **2016**, *18*, 1078–1086. [CrossRef]
19. Van Der Merwe, R.; Wan, E.; Julier, S. Sigma-point Kalman filters for nonlinear estimation and sensor-fusion: Applications to integrated navigation. In Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, Providence, Rhode Island, 16–19 August 2004; pp. 5120–5149.
20. Wendel, J.; Metzger, J.; Moenikes, R.; Maier, A.; Trommer, G. A performance comparison of tightly coupled GPS/INS navigation systems based on extended and sigma point Kalman filters. In Proceedings of the 18th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2005), Long Beach, CA, USA, 13–16 September 2005; pp. 456–466.
21. Strasdat, H.; Montiel, J.M.; Davison, A.J. Visual SLAM: Why filter? *Image Vision Comput.* **2012**, *30*, 65–77. [CrossRef]
22. Rehder, J.; Gupta, K.; Nuske, S.; Singh, S. Global pose estimation with limited GPS and long range visual odometry. In Proceedings of the 2012 IEEE International Conference on Robotics and Automation (ICRA), St. Paul, MN, USA, 14–18 May 2012; pp. 627–633.
23. Merfels, C.; Stachniss, C. Pose fusion with chain pose graphs for automated driving. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, South Korea, 9–14 October 2016; pp. 3116–3123.
24. Kalman, R.E. On the general theory of control systems. *IRE Trans. Autom. Cont.* **1959**, *4*, 110–110. [CrossRef]
25. Surber, J.; Teixeira, L.; Chli, M. Robust visual-inertial localization with weak GPS priors for repetitive UAV flights. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Marina Bay Sands, Singapore, 29 May–3 June 2017; pp. 6300–6306.
26. Shoemake, K. Animating rotation with quaternion curves. In Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques, San Francisco, CA, USA, 22–26 July 1985; pp. 245–254.
27. Huber, P.J. *Robust Statistics*; John Wiley & Sons: Hoboken, NJ, USA, 2004.
28. Levenberg, K. A method for the solution of certain non-linear problems in least squares. *Q. Appl. Math.* **1944**, *2*, 164–168. [CrossRef]
29. Marquardt, D.W. An algorithm for least-squares estimation of nonlinear parameters. *J. Soc. Ind. Appl. Math.* **1963**, *11*, 431–441. [CrossRef]
30. Li, M.; Mourikis, A.I. High-precision, consistent EKF-based visual-inertial odometry. *Int. J. Rob. Res.* **2013**, *32*, 690–711. [CrossRef]
31. Burri, M.; Nikolic, J.; Gohl, P.; Schneider, T.; Rehder, J.; Omari, S.; Achtelik, M.W.; Siegwart, R. The EuRoC micro aerial vehicle datasets. *Int. J. Robot. Res.* **2016**, *35*, 1157–1163. [CrossRef]
32. Jeong, J.; Cho, Y.; Shin, Y.S.; Roh, H.; Kim, A. Complex urban dataset with multi-level sensors from highly diverse urban environments. *Int. J. Robot. Res.* **2019**, *38*, 642–657. [CrossRef]
33. Kümmerle, R.; Grisetti, G.; Strasdat, H.; Konolige, K.; Burgard, W. G2o: A general framework for graph optimization. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 9–13 May 2011; pp. 3607–3613.
34. Sturm, J.; Engelhard, N.; Endres, F; Burgard, W.; Cremers, D. A benchmark for the evaluation of RGB-D SLAM systems. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Algarve, Portugal, 7–12 October 2012; pp. 573–580.