Map-Based Visual-Inertial Localization: Consistency and Complexity

Zhuqing Zhang, Yanmei Jiao, Shoudong Huang, Senior Member, IEEE, Rong Xiong, Member, IEEE, and Yue Wang

Abstract—Drift-free localization is essential for autonomous vehicles. In this letter, we address the problem by proposing a filter-based framework, which integrates the visual-inertial odometry and the measurements from the pre-built map. In this framework, the transformation between the odometry frame and the pre-built map frame is augmented into the system state vector and estimated on the fly. Besides, we maintain the map keyframe poses and employ the Schmidt extended Kalman filter to update the state partially so that the uncertainty of the map information can be consistently considered with low computational complexity. Moreover, we theoretically demonstrate that the ever-changing linearization points of the estimated augmented state make the original four-dimensional unobservable subspace vanish, leading to the inconsistent estimation in practice. To relieve this problem, we employ the first-estimate Jacobian (FEJ) technique to maintain the correct observability properties of the augmented system. Furthermore, we introduce an observability-constrained updating method to compensate for the significant accumulated error after the long-term absence of map-based measurements. Finally, by evaluating the system through both simulation and real-world experiments, we confirm that the system has good consistency and low computational complexity.

Index Terms—Localization, visual-inertial SLAM.

I. INTRODUCTION

Accurate localization is primary for autonomous vehicles. With equipped sensors (e.g., cameras) on the vehicle, the environment features can be extracted. By matching these features with those in a pre-built map, the pose of the vehicle in the environment features can be extracted. By matching these features with those in a pre-built map, the pose of the vehicle in the pre-built map frame, which leads to the low computational complexity (O(1)). This mechanism is also applied in Matlab [12]. However, the state formulation such as [16] and [12] ignores the uncertainty of the map and the correlation between the current state and the map information, bringing overconfident estimation.

Apart from the filter-based VIL systems, optimization-based VIL systems are also proposed, where the global measurements are fused into the system by formulating a factor graph and is optimized by iterative algorithms [7], [8], [13], [19]. However, lightweight visual-inertial odometry (VIO) [5], [6], [7]. However, these VIO systems will suffer from unbounded drift for long-term running. One way to compensate for the drift is to introduce the global positioning system (GPS) [8], [9], [10]. Nevertheless, GPS can sometimes be unstable, especially in the city canyon or parking lot environments. Another widely used method is to employ a pre-built map to bound the drift [11], [12], [13]. However, less attention is paid to the theoretical analysis for the such fusion of map-based measurements and VIO. In this letter, we focus on the computational complexity and consistency of the map-based VIL. 

A typical visual map could contain m keyframes and n point features (m ≪ n) where m may be in the order of magnitude 10^2 and n 10^4. Besides, the map is actually not perfect, i.e., not 100% accurate, and the uncertainty of the map should be considered to consistently fuse the map. To make the localization system run in real-time, different trade-offs have been made in the previous efforts.

To integrate map information, an intuitive and theoretically sound way is to keep all the map-related variables (e.g. map features) in the state vector, and the map-based observations are used to update the state through an extended Kalman filter (EKF), equivalent to the formulation of EKF simultaneous localization and mapping (EKF-SLAM). However, since the computational complexity of updating covariance is O(n^2), this makes the system difficult to run in real-time, especially for a large map. To solve this problem, [14] proposes a system based on Schmidt-EKF [15], which does not update the map state, leading to a conservative yet consistent estimation, reducing the computational complexity to O(n). However, the method in [14] still stores the whole map in the state vector, which makes the state dimension quite high and needs lots of memory (O(n^2)) for the covariance. Another framework for efficient fusion is multi-sensor fusion [16], which follows a loosely coupled fashion. Also, [16] regards multiple map feature measurements as an integrated measurement to update the relative transformation between the local odometry reference frame and the map reference frame, which leads to the low computational complexity (O(1)). This mechanism is also applied in Maplab [12]. However, the state formulation such as [16] and [12] ignores the uncertainty of the map and the correlation between the current state and the map information, bringing overconfident estimation.

In this letter, when we mention the word “map,” it refers to the pre-built visual map, instead of the online-built map.
these methods regard maps as perfect, leading to inconsistent estimation. Besides, compared with the filter-based VIL, the superior accuracy of the optimization-based VIL is based on the cost of more computation resources.

A general idea in the most of methods mentioned above is to introduce a variable, i.e., the relative transformation between the VIO reference frame and the map reference frame. We call this variable as the augmented variable, and the system online estimates the augmented variable as the augmented system. However, as analyses in this paper (Section IV-A), estimating the augmented variable online can make the system inconsistent, producing an overconfident estimation. Even though there are many works analyzing the observability properties of the VIO system [17, 20, 21, 22], these analyses cannot reflect the observability of the augmented system. To the best of our knowledge, the observability properties of the augmented system have not been analyzed in previous works and remain vague.

In this paper, we propose a filter-based drift-free localization framework with low computational complexity yet good consistency. To be specific, the keyframe poses from the pre-built map are maintained in the state to consider the uncertainty of the map. To avoid the large computation of updating the map keyframes, whose computational complexity is \( O(m^2) \), we apply the Schmidt-EKF to reduce the computational complexity to \( O(m) \). Moreover, we theoretically demonstrate that for the ideal case, the augmented system has four dimensions of unobservable subspace, whereas, for the real case, such unobservable subspace vanishes, resulting in inconsistency. To relieve this problem, we introduce the first-estimate Jacobian (FEJ) technique to fix the linearization points. In addition, to correct the large drift, we propose an error-compensated updating step while keeping the linearization point fixed, which can be useful when map-based measurements become available again after a long-term absence. Finally, we evaluate the proposed localization system using simulation data as well as datasets collected from three types of vehicles, showing the theory validity and system effectiveness.

To summarize, the main contributions of this paper are threefold:

- We propose a map-based drift-free visual-inertial localization system that maintains consistent state estimation and low computational complexity.
- Map uncertainty is taken into consideration while keeping the low computational complexity with Schmidt-EKF.
- Theoretically analyze the observability properties of the augmented system, which inspires the FEJ, and the error compensated updating techniques for better estimation.

## Table I

| Frame | Description                                                                 |
|-------|-----------------------------------------------------------------------------|
| \( L \) | The local inertial reference frame, which is a fixed frame defined by the initial pose of VIO (cf. \( L \) in Fig. 1). |
| \( G \) | The pre-built map reference frame. All the map-related information (including map keyframes and map features) is based on this frame (cf. \( G \) in Fig. 1). |
| \( I_k \) | The IMU (body) frame at timestamp \( k \), which is attached to the robot. VIO online estimates the transformation between \( L \) and \( I_k \), i.e., \( L T_{I_k} \). |
| \( C_k \) | The camera (image) frame at timestamp \( k \). The camera online observes features in \( C_k \). The extrinsics between the camera and the IMU are assumed to be known. |
| \( KF\_1 \) \( KF\_2 \) | The \( i^{th} \) image keyframe of the pre-built map, whose pose \( L T_{KF\_i} \) is represented in \( G \) (cf. \( KF\_1 \) and \( KF\_2 \) in Fig. 1). After the map is built, the keyframes \( \{ KF\_i \} \) are fixed. |

### Frame definition

As is shown in Fig. 1 and Table I, for the augmented system there are five kinds of frames, i.e., \( L, I_k, C_k, G, \) and \( KF\_i \), where \( L, I_k, \) and \( C_k \) are VIO-related frames, and \( G \) and \( KF\_i \) are map-related frames.\(^2\)

**Drift-free visual-inertial localization** Based on the frames defined above, there are two unknown transformations, one is \( ^G T_L \), i.e., the augmented variable, and the other is \( ^L T_{I_k} \). VIO yields the estimation of the latter with drift. The problem of a drift-free VIL system is then defined as estimating the former and correcting the latter, so that the transformation between \( G \) and \( I_k \) is drift-free by leveraging the local feature\(^3\) tracking and map-based measurements. By concatenating \( ^L T_{I_k} \) and \( ^G T_L \), i.e., \( ^G T_{I_k} = ^G T_L ^L T_{I_k} \), we can localize the body pose in the map frame. The design principles of our VIL algorithm are consistency and low computational complexity.

### III. System Modeling and Filter Design

The system model consists of two parts. The VIO part and the map-based measurements part. MSCKF-style filtering [5] is applied for VIO, while Schmidt filtering [15] is applied for map-based measurement updating.

#### A. Visual-Inertial Odometry

This subsection briefly introduces the employed VIO in our framework, Open-VINS [6], which is an open-sourced implementation of MSCKF [5]. For the details, please refer to [5], [6].

Following the definition of [6], the state vector at time step \( k \) is defined as:

\[
\begin{align*}
-x_{VIOk} &= [x_{I_k}^T \ x_{cl_k}^T]^T, \\
x_{I_k} &= [I_k q_L^T \ L v_{I_k}^T \ L p_{I_k}^T \ b_{I_k}^T \ b_{n_k}^T]^T, \\
x_{cl_k} &= [L T_{I_{k-1}}^T \ldots \ L T_{I_{k-N}}^T] \Delta T = [x_{cl_{k-1}}^T \ldots \ x_{cl_{k-N}}^T]^T,
\end{align*}
\]

where \( I_k q_L \) is a unit quaternion variable. Its corresponding rotation matrix \( I_k R_L \) rotates a 3D vector from \( L \) to \( I_k \). \( L v_{I_k} \) is the position of the body in \( L \) at time step \( k \). \( L v_{I_k} \) is the velocity of the body in \( L \).
of the body in frame $L$. $b_{gk}$ and $b_{ak}$ are the gyroscope and accelerometer bias, respectively. The cloned state $x_{clk}$ consists of the $N$ latest (cloned) historical body poses $x_{clk,i}, i = 1 \ldots N$. $x_{clk}$ is the so-called sliding window.

**State propagation** When we receive IMU data at time step $k$, the state vector and covariance can be propagated from time step $k$ to $k+1$:

$$x_{k+1|k} = f(x_k, a_{mk}, n_{mk}, \omega_{mk}, n_{gk}), \quad (2)$$

$$P_{k+1|k} = \Phi_k P_k \Phi_k^T + G_k Q_k G_k^T, \quad (3)$$

where $f$ represents the IMU kinematics equations, $a_{mk}$ and $\omega_{mk}$ are the measurements (body linear and angular velocity) derived from the IMU, $n_{mk}$ and $n_{gk}$ are the zero-mean Gaussian white noise of the IMU measurements. The subscript $k+1|k$ means the propagated value from the time step $k$ to $k+1$, with all the measurements up to time step $k$ being processed. $P_k$ and $Q_k$ are the system state and the noise covariance matrices, respectively. $\Phi_k$ and $G_k$ are the Jacobians of (2) with respect to the system state and the noise, respectively.

**Observation function** For each tracked local feature $f_i$, it can be observed by a set of historical body poses in the cloned state $x_{clk}$. By reprojecting this local feature from 3D to the 2D image plane, the observation error function can be constructed and used to update the state following the steps of EKF. Besides, in order not to maintain the local features in the state, the feature-related parts are marginalized by the null-space projection. For the detailed description, please refer to [5], [6].

### B. Map-Based Measurement

This subsection forms the key augmentation of the VIO system. With the map-based measurement introduced below, the map information can be consistently incorporated into the system to ease the drift of the odometry.

**Augmented state formulation** When we first get the matches between the map keyframe (e.g. $KF_1$ in Fig. 1) and the current frame $C_k$, the relative transformation $\hat{T}_{C_k}$ can be obtained through the techniques of 3D-2D pose estimation e.g., EPnP [24]. This value, combined with the matched keyframe pose $G_{KF_1}$ and the current frame pose $\hat{T}_{C_k}$, will be used to initialize $G_{TF}$. After the initialization, $x_T \triangleq G_{TF}$ will be augmented into the state vector:

$$x_k \triangleq [x_{1:O}^T \ x_{1:T}^T \ x_{K}^T \ x_T^T]^T. \quad (4)$$

Besides, the covariance of the state would also be augmented. As the initial estimation of $x_T$ may be inaccurate, its covariance can be assigned with a big value.

It should be noted that to consider the uncertainties of the map keyframe poses, every time there are matched keyframes, their poses are added to the state vector:

$$x_k = [x_{1:O}^T \ x_{1:T}^T \ x_{K}^T \ x_T^T]^T, \quad (5)$$

$$x_{KF} = [G_{TF}^T \ G_{TF_2}^T \ \ldots \ \ G_{TF_{N_{KF}}}^T] \triangleq [x_{KF_1}^T \ \ldots \ \ x_{KF_{N_{KF}}}^T]^T, \quad (6)$$

where $x_{KF}$ is the pose of the map keyframe $KF_i$.

For brevity, we divide the state into two parts: the active part $x_A \triangleq [x_{1:O}^T \ x_{1:T}^T]^T$, and the nuisance part $x_N \triangleq x_{K}^T$, leading to the following expressions:

$$x_k = [x_A^T \ x_N^T]^T, \quad (7)$$

$$P_k = [P_{AAk} \ P_{ANk} \ P_{ANk} \ P_{NN_{KF}}],$$

where $P_{AAk}$ and $P_{NN_{KF}}$ are the self-covariance of $x_{Ak}$ and $x_{N_{KF}}$, respectively, and $P_{AN_{KF}}$ is the cross-covariance between $x_{Ak}$ and $x_{N_{KF}}$. The partition in (7) is useful for the Schmidt-EKF updating, which is introduced in Section III-C.

**Observation functions** As is shown in Fig. 1, suppose there is a map feature $f_1$, whose 3D coordinate $x_{KF_1}$ is anchored in the map keyframe $KF_1$. At the time step $k$, this feature is observed by the current camera $C_k$. Then, $KF_1$ can be reprojected into the 2D current image frame by $C g$:

$$C z_{f_1} = C g(x_{KF_1}, x_{Ak}, KF_1) + C n_{f_1}, \quad (8)$$

where $C z_{f_1}$ is the 2D pixel observation of $f_1$ in the current image. $C n_{f_1}$ is the observation noise with Gaussian distribution. Linearizing (8) at the estimated values, we have:

$$\hat{A} z_{f_1} = \hat{A} g(KF_{f_1}) + \hat{A} n_{f_1}, \quad (9)$$

where $\hat{A} r_{f_1}$ is the 2D pixel observation error. $C H_{x_{Ak}}$ is the Jacobian of $\hat{C} g$ with respective to the estimated $x_{Ak}$, and $C H_{x_{N_{Ak}}}$ and $C H_{x_{N_{KF}}}$ have the similar definitions. $KF_{f_1}$ can also be reprojected into the anchored keyframe $KF_1$ and (maybe) the other map keyframes that observe $f_1$ (like $KF_2$ in Fig. 1) through the functions $\hat{g}$ and $\hat{g}$:

$$\hat{A} r_{f_1} = \hat{A} H_{x_{KF_1} f_1} KF_{f_1} + \hat{A} n_{f_1}, \quad (10)$$

$$O z_{f_1} = O g(x_{KF_1}, x_{KF_2}, KF_{f_1}) + O n_{f_1}, \quad (11)$$

where $O z_{f_1}$ and $O z_{f_1}$ are the 2D pixel observation of $f_1$ in the Anchored keyframe $KF_1$ and the Other keyframe $KF_{f_1}$, respectively. $\hat{A} n_{f_1}$ and $\hat{A} n_{f_1}$ are the observation noises.

In summary, whenever there are feature matchings between the current image and a single (or multiple) map keyframe(s), for each 3D map feature $f_i$ anchored in the keyframe $KF_{f_i}$, we can formulate the observation functions by (9), (10) (and (11)). Stacking these functions together, we have the observation function with the following form:

$$M r_{f_i} = M H_{x_{Ak}} x_{Ak} + M H_{x_{N_{Ak}}} x_{N_{Ak}} + M H_{x_{KF_{f_i} f_i}} KF_{f_i} + M n_{f_i}, \quad (12)$$

where $M r_{f_i}$ is the stack of $\hat{A} r_{f_1}$ and $O r_{f_1}$, and $M H_{x_{Ak}}$, $M H_{x_{N_{Ak}}}$, $M H_{x_{KF_{f_i} f_i}}$ and $M n_{f_i}$ can be derived in a similar way.

**Null space projection** Note that as we do not maintain the map features in the state, the map feature related part in (12), i.e., $M H_{x_{KF_{f_i} f_i}} KF_{f_i}$ should be marginalized by the null-space projection technique introduced in [5].

For each landmark, it has at least two measurements (in the current image and a single (or multiple) map keyframe(s)), leading to the row number of $H_{KF_{f_i} f_i}$ (being equal or greater than four) being more than its column number (in our case is three). This fact guarantees that we can find a left null space of $H_{KF_{f_i} f_i}$. By multiplying both sides of (12) with the left null space of $H_{KF_{f_i} f_i}$, we have:

$$r_{f_i}^* = J H_{x_{Ak}}^* x_{Ak} + J H_{x_{N_{Ak}}}^* x_{N_{Ak}} + n_{f_i}^*, \quad (13)$$

In this paper, we use $\hat{x}$ to represent the estimated value of $x$. We use $\hat{x}$ to represent the error between the true value $x$ and the estimated value $\hat{x}$. 

Therefore, the system state can be updated as $x_{k+1}$:
Note that (13) only considers one matched map feature. We need to stack (13) for all matched map features to get the final observation function. By stacking multiple $r_k^f, H^*_Ak, H^*_Nk, H^*_{rk}$ and $n^*_k$, we get $r^*_k, H^*_Ak, H^*_Nk$ and $n^*_k$, which leads to the following expression:

$$
r^*_k = 
\begin{bmatrix}
  H^*_Ak \\
  H^*_Nk 
\end{bmatrix}
\begin{bmatrix}
  \hat{x}_Ak \\
  \hat{x}_Nk 
\end{bmatrix} + n^*_k,
$$

(14)

where $H^*_Nk$ represents the Jacobian matrices related to the map keyframes, and $H^*_Ak$ represents the Jacobian matrices related to the VIO state and the augmented variable. Considering the state partition of (7), we have the following concise expression:

$$
r^*_k = H^*_k \hat{x}_k + n^*_k,
$$

(15)

which is used to perform the map-based updating.

### C. Map-Based Updating

**Limitation of standard EKF update** For the standard EKF, when map-based measurements come, the Kalman gain of the update is computed by:

$$
K_k = \begin{bmatrix} K_{Ak} \\ K_{Nk} \end{bmatrix} = \begin{bmatrix} P_{AKk} & P_{ANk} \\ P_{NK} & P_{NNk} \end{bmatrix} \begin{bmatrix} H^*_Ak \\ H^*_Nk \end{bmatrix} \begin{bmatrix} H^*_Ak \\ H^*_Nk \end{bmatrix}^{-1} \begin{bmatrix} K_{Ak} \\ K_{Nk} \end{bmatrix} S^{-1}_k,
$$

(16)

where $S_k = H^*_k P_k^{-1} H^*_k + R_k$, $R_k$ is the covariance of the observation noise. And the state covariance is updated by:

$$
P_k = P_k^{-1} \begin{bmatrix} K_{Ak} S_k K^*_Ak & K_{Ak} H^*_k P_{ANk} \\ P_{ANk} & P_{NNk} \end{bmatrix} H^*_k K_{Ak} S_k K^*_Ak.
$$

(17)

From (17), one can see that the computation is dominated by $K_{Ak} S_k K^*_Ak, S_k K^*_Ak$, whose computational complexity is quadratic of the size of map-related variables, limiting the real-time performance of the standard EKF.

**Schmidt Update** To reduce the computational complexity, we employ the Schmidt-EKF to update the state partially. More specifically, instead of directly utilizing (17), we set $K_{Nk}$ in (17) to 0, and the state is updated by:

$$
\hat{x}_Ak = \hat{x}_{Ak}_{k-1} + K_{Ak} r^*_k, \quad \hat{x}_Nk = \hat{x}_{Nk}_{k-1}.
$$

(18)

It can be seen that the active part updating is identical to the standard EKF, while the nuisance part will not be updated. In this way, the computational complexity reduces to linearity. The more detailed procedure of the Schmidt update is given in Section IV-C of our supplementary material [32].

### IV. OBSERVABILITY ANALYSIS AND IMPROVEMENT

In this section, we analyze the observability properties of the augmented system and demonstrate the improvement techniques for consistency and linearization error.

#### A. Observability Analysis

For simplicity, we assume the camera frame and the IMU frame are identical and consider one local feature. Besides, the IMU bias is neglected following [10]. Therefore, we get the following simplified state vector:

$$
x_{sk} = \begin{bmatrix} \ell_k^q L^q \ell_k^p \ell_k^p & L^q \ell_k^p & G^q \ell_k G^p \ell_k \end{bmatrix}^T,
$$

(19)

which includes the body pose $\ell_k^q, \ell_k^p$, velocity $\ell_k^q, \ell_k^p$, a local feature $\ell_k^p$ in the frame $L$, and the relative transformation $G^q \ell_k, G^p \ell_k$ between $L$ and $G$.

To analyze the observability properties of the augmented system, we need to derive the right null space of the system observability matrix. We first assume that all the Jacobian matrices during the whole procedure are evaluated at the ground truth, which is ideal but can demonstrate the theoretical implication:

**Lemma 1:** (Ideal observability) The right null space $\mathcal{N}$ of the system observability matrix, where the Jacobian matrices are evaluated ideally, is spanned by four directions as

$$
\mathcal{N} = \text{span} \begin{bmatrix} I_6 R_L g & 0_3 \\ -L v_{Io} x g & I_3 \\ -P_f x g & I_3 \\ G R_L g & 0_3 \\ 0_{1x1} \end{bmatrix},
$$

(20)

**Proof:** see Appendix A of our supplementary material [32]. However, in real practice, we cannot access the ground truth poses, and can only evaluate the Jacobian matrices at some points with linearization error, which breaks the original observability of the system:

**Theorem 2:** (Real observability) The right null space $\mathcal{N}$ of the system observability matrix, where the Jacobian matrices are evaluated at the changing estimated values, vanishes.

**Proof:** see Appendix A of our supplementary material [32].

**Intuitive explanation** For the ideal system, the augmented system has the same unobservable dimensions as the visual-inertial system [21]. This result is not surprising. We can understand the relation between the proposed result and previous works by regarding the augmented variable as another $6$ DoF feature in the VIO frame and never being marginalized. With or without this $6$ DoF feature does not affect the unobservable subspace of the visual-inertial system. However, the ever-changing linearization point of $G R_L$ introduces spurious information to the system and makes the system mistakenly thinks it can derive absolute values of the estimated state through the provided measurements. Obviously, the estimation is inconsistent [23].

**First-Estimate Jacobian** From the analysis above, we aim to preserve observability to relieve inconsistency. We carefully extend the traditional first-estimate Jacobian (FEJ) [17] to the augmented system in three steps:

- The propagation Jacobian matrix is evaluated at $x_{Ak_{k-1}}$.
- The local feature observation Jacobian matrix is always evaluated at first-estimate of $L^p f_k$ and $x_{Ak_{k-1}}$.
- The map-based measurement Jacobian matrix is always evaluated at first-estimate of $G R_L$ and $x_{Ak_{k-1}}$.

With these steps, the linearization points of $L^p f_k$ and $G R_L$ will never change. Therefore, the right null space of the system observability matrix in real practice keeps the same as that in the ideal case. However, we emphasize that the employment of FEJ avoids changing the linearization point and improves the consistency, but the linearization error still exists [18].

#### B. Observability Constrained Error Compensation

As shown in Section III, in the update step, the observation function follows a general form $h(x)$, and is linearized at $\hat{x}$. We ignore the high-order terms as

$$
z = h(x) + n \approx h(\hat{x}) + H_x (x - \hat{x}) + n,
$$

(21)
where $H_k$ is the Jacobian matrix of the observation function linearized at $\hat{x}$. This approximation is only accurate when $\hat{x}$ is close to the true value. When the error between the estimated value and the true value is big, poor approximation may lead to the failure of filtering.

**Absence of map-based measurements** The map-based measurement can be absent for a long duration due to various reasons, say the environment changes [4]. Thus, the odometry drifts in this duration. When the map-based measurement is available again, the estimation of $LT_{C_k}$ has a large error, severely reducing the linearization accuracy of the map-based observation function.

**Limitation of re-linearization** A simple idea to relieve the large error in the linearization point is to find a more accurate point. To be specific, when we get feature matchings between a map keyframe and the current image, we can calculate a relatively accurate transformation $K^F T_{C_k}$ using EPI [24]. Based on this more accurate $K^F T_{C_k}$, we can calculate a new $LT_{C_k}$ or a new $G T_L$. The new value is then used to linearize the observation function (8). However, as analyzed in Section IV-A, we employ FEJ to preserve the desired observability, which fixes the linearization point, preventing the observation function from being re-linearized.

**Large error compensation** Following the FEJ constraint, when a map-based measurement is acquired, we have the observation function as

$$ r \triangleq z - h(\hat{x}) \approx H_{x_{FEJ}}(x - x_{r}) + n, \quad (22) $$

where $\hat{x}$ is the estimated value after the EKF propagation. $H_{x_{FEJ}}$ is the first-estimate Jacobian matrix. We employ the re-calculated $G T_L$ to replace the one in $x$, denoted as $x_{r}$. Then, we have

$$ r' \triangleq z - h(x_{r}) \approx H_{x_{FEJ}}(x - x_{r}) + n. \quad (23) $$

Note that we have the propagated value at $\hat{x}$, which is different from the $x_{r}$ in (23). To address it, (23) is rewritten as

$$ r' \approx H_{x_{FEJ}}(x - x_{r} + \hat{x} - \hat{x}) + n, \quad (24) $$

and the compensated observation function is defined as:

$$ E_C r \triangleq z - h(x_{r}) - H_{x_{FEJ}}(x - x_{r} + \hat{x} - \hat{x}) = H_{x_{FEJ}} \hat{x} + n. \quad (25) $$

which is employed as the observation function for Schmidt update after the long-term absence of map-based measurements. The trigger of employing this compensated observation function is decided by a threshold, which is the average re-projection error of the map landmarks. In the following section, we will show that (25) improves the accuracy of localization for large scenarios efficiently.

V. EXPERIMENTAL RESULTS

In this section, we would first demonstrate the setting and configuration of simulation, and datasets captured from an aerial vehicle, ground vehicles on campus, and urban areas. Then, extensive experimental results are given to validate the effectiveness of our proposed method.

**A. Datasets and Settings**

**Simulation** We use a $940^m$ trajectory shaped like “number eight” (cf. the orange trajectory of Fig. 2) to generate the IMU data and local features so that the VIO can run online. The map keyframe poses come from another trajectory, which has a similar shape as the previous trajectory (cf. the blue trajectory of Fig. 2). These two trajectories are denoted as EIGHT1 (the blue one) and EIGHT2 (the orange one), respectively.

To test the consistency of our proposed system, we artificially add the noise to the ground truth of EIGHT1 to simulate the *imperfect* map keyframes. The position of the ground truth is perturbed with the Gaussian white noise $n_p \sim \mathcal{N}(0, 0.01 I_{3 \times 3} m^2)$, and the orientation is perturbed with $n_o \sim \mathcal{N}(0, 0.00025 I_{3 \times 3} rad^2)$. Therefore, the deviation for the position is $0.1^m$, and the deviation for the orientation is around $1^\circ$. After the perturbation, the root-mean-squared error (RMSE) of the map trajectory of EIGHT1 is $0.179^m$.

For the map matching information, we first randomly generate 3D map features, then these features are reprojected into the perfect map keyframes and the camera frame of the running trajectory so that the 2D-2D matching features are obtained, after which we add one-pixel Gaussian white noise to the 2D observations. Finally, for each map feature, we utilize the noisy 2D observations and the noisy map keyframe poses to triangulate its 3D position as the estimated 3D map features.

**Real-world Datasets** We use four kinds of datasets to test our proposed algorithm. They are EuRoC [27], Kaist [28], 4Seasons [30], and YQ [29]. The detailed settings of each dataset are given in Table II. The test sequence(s) has a large overlap with the map sequence. For the map information, the map keyframe poses are directly gotten from the corresponding ground truth. Due to the ground truth of EuRoC being derived from the motion capture while the other datasets are derived from RTK, the covariance of the map keyframe poses in EuRoC is much less than those in other datasets. The 3D map features are triangulated across multiple adjacent map keyframe images and the corresponding poses.

**Feature matching** For all real-world datasets, we first utilize R2D2 [25] to extract new features from the current query frame and match them with features in map keyframes. The initial matching information based on the feature descriptor is then fed into a robust pose estimator in [26] to generate accurate feature matches.

### Table II

**Different Settings of the Four Datasets**

| name       | scenario     | map sequence | test sequence(s) | map keyframes covariance |
|------------|--------------|--------------|------------------|--------------------------|
| EuRoC      | machine hall | MH01         | MH02-05          | $1e^{-6} m^2/3e^{-2} rad^2$ |
| Kaist      | urban campus | Urban38      | Urban39          | $0.1m^2/0.1rad^2$         |
| 4Seasons   | industrial area | OfficeLoop1 | OfficeLoop2     | $0.1m^2/0.1rad^2$         |
| YQ         | campus       | YQ1          | YQ2-4            | $0.1m^2/0.1rad^2$         |
B. Ablation Experiments

We first perform ablation studies to validate each proposed algorithm module substantially. We regard Open-VINS [6] as the pure VIO baseline method. We call our basic module MVIL (map-based visual-inertial localization). As mentioned in Section III-B, the number of the matched map keyframe can be one or more. We denote the setting where only one map keyframe is matched at a time as Single Matching (SM) and the one where multiple map keyframes are matched at a time as Multiple Matching (MM). To show the necessity of considering the map uncertainty, a setting that treats the map information as perfect/constant (MC) is used for comparison. Besides, whether or not using FEJ (Section IV-A) is also an essential factor for testing. The mechanism of error compensated updating (ECU) mentioned in Section IV-B is also tested.

Trajectory accuracy Table III gives the ablation study from the simulation data. In the table, the localization accuracy in the map reference frame is evaluated by RMSE/m. We can find that MVIL-MM-FEJ has the best result, which is in accordance with our theory: On the one hand, it has the correct null space of the observability matrix compared with MVIL w/o FEJ and considers the uncertainty of the map compared with MVIL-MM-MC. On the other hand, compared with SM, multiple matching frames provide more constraints to improve the accuracy. When the map is treated as perfect (MVIL-MM-MC), the performance of the algorithm is even worse than that of pure odometry (Open-VINS), which demonstrates the necessity of considering the map uncertainty.

Covariance reliability Due to the fact that what the system online maintains are the VIO pose \( L T_0 \) and the augmented variable \( G T_L \), to demonstrate the consistency of our proposed method, we plot the estimated error of \( L T_0 \) with 3-\( \sigma \) bounds based on the simulation results (cf. Fig. 3). Also, the normalized estimation error squared (NEES) of \( G T_L \) is plotted in Fig. 4. For a consistent algorithm, the estimated error should be always within the 3-\( \sigma \) bounds and the value of NEES should be around one. From the figures, we conclude that MVIL-FEJ has good consistency while MVIL-MC has an overconfident result because it assumes the map is perfect and does not consider the uncertainty of the map.

Linearization strategy We also compare our proposed methods with/without FEJ on EuRoC. The estimated position RMSEs/m in the map frame are shown in Table IV. We find that in most cases, methods that maintain the correct unobservable space have better results. Exceptions can be found for the SM setting. Since we use the first-estimate value of \( G T_L \) to fix Jacobian linearization points, if this value is inaccurate, the linearization error can be the main error source, especially in such a short-term dataset. This phenomenon vanishes for the MM setting as we can use multiple matching information to get a better initial value of \( G T_L \).

Error compensated updating To show the advantage of our proposed error compensated updating (ECU) mechanism, we conduct experiments on the challenging Kaist dataset [28]. The threshold for triggering ECU is set as 20 pixels. The comparison of different algorithms is given in Table V, and the trajectories derived from MVIL-MM-FEJ and MVIL-MM-FEJ-ECU are plotted in Fig. 5.

From the results, we find that there is an obvious promotion in accuracy for both SM and MM scenarios when we employ the ECU. We refer to Fig. 5 for more illustration. In the figure, the pair of stars represents two contiguous matches (for example, the \( i^{th} \) match in red and the \( i + 1^{th} \) match in blue), and so does the pair of diamonds. The motion direction of the car from the red mark to the blue mark is shown by black dotted arrows. The distance between the two stars or the two diamonds is far, which indicates the long absence of map-based measurements. Due to the drift of the VIO, the first-order Taylor series approximation of the observation function is not good, as analyzed in Section IV-B. The effectiveness of ECU is highlighted in Fig. 5. We find that for the MVIL-MM-FEJ, the trajectory around the blue star (cf. the

---

**TABLE III**

| Method          | SM | MM | FEJ | MC | RMSE |
|-----------------|----|----|-----|----|------|
| Open-VINS [6]   | ✓  | ✓  | ✓   | ✓  | 0.135|
| MVIL-MM-FEJ     | ✓  | ✓  | ✓   | ✓  | 0.131|
| MVIL-MM-MC      | ✓  | ✓  | ✓   | ✓  | 0.113|
| MVIL-MM-FEJ 3σ  | ✓  | ✓  | ✓   | ✓  | 0.068|
| MVIL-MM-MC 3σ   | ✓  | ✓  | ✓   | ✓  | 0.057|
| MVIL-MM-FEJ 5σ  | ✓  | ✓  | ✓   | ✓  | 0.179|

“✓” means the corresponding algorithm is selected.

---

**TABLE IV**

| Method          | SM | MM | FEJ | ECU | RMSE |
|-----------------|----|----|-----|-----|------|
| MVIL-MM-FEJ     | ✓  | ✓  | ✓   | ✓   | 0.049|
| MVIL-MM-MC      | ✓  | ✓  | ✓   | ✓   | 0.087|
| MVIL-MM-FEJ 3σ  | ✓  | ✓  | ✓   | ✓   | 0.178|
| MVIL-MM-MC 3σ   | ✓  | ✓  | ✓   | ✓   | 0.201|

“✓” means the corresponding algorithm is selected.

---

**TABLE V**

| Method          | SM | MM | FEJ | ECU | RMSE |
|-----------------|----|----|-----|-----|------|
| IEKF [31]       | ✓  | ✓  | ✓   | ✓   | 17.965|
| MVIL-MM-FEJ     | ✓  | ✓  | ✓   | ✓   | 6.269 |
| MVIL-MM-MC      | ✓  | ✓  | ✓   | ✓   | 5.926 |
| MVIL-MM-FEJ 3σ  | ✓  | ✓  | ✓   | ✓   | 5.141 |

“✓” means the corresponding algorithm is selected.
In this subsection, we make a comparison between our proposed methods with Open-VINS [6], VINS-Fusion [7], Maplab [12] on EuRoC, Kaist, 4Seasons, and YQ.

We regard Open-VINS as a pure odometry baseline to validate the correction of map-based measurements. For VINS-Fusion and Maplab, their localization mode is used. We record their localization results by concatenating the odometry \( L^T I_k \) and the estimated \( G^T L \) instead of recording the optimized trajectory due to the real-time causality.

**Map-based measurement frequency** In the first row of Table VI, we list the minimum/maximum time gap (measured in seconds) between two consecutive map-based measurements on different datasets sequences. We can find that for most of the sequences, there is a long-term absence of map-based measurement, which indicates that the odometry will have a large drift. This situation encourages us to utilize the ECU mechanism to improve the accuracy of the localization.

**Trajectory accuracy** The comparison of experimental results is shown in Table VI. The trajectory accuracy is measured by RMSE/m. The estimated trajectory of Open-VINS is aligned with the ground truth by the first estimated pose. The other localization methods are compared without alignment. All the results are the average of three runs. Note that as EuRoC is recorded in small scenes, the drift of the estimator is not distinct as that in the other three datasets, so the results of MVIL-SM/MM-ECU for EuRoC are omitted.

For Maplab, it only successfully runs on EuRoC dataset, whereas its VIO quickly drifts far away on the other three datasets. Even for EuRoC, the performance of Maplab is still inferior to that of our proposed algorithm as shown in Table VI. For VINS-Fusion, even though it has better performance than Maplab on EuRoC, our methods still show superior performance because our framework considers the uncertainty of the map and fuses the map in a tightly-coupled manner. For large and challenging scenarios (Kaist and YQ), VINS-Fusion diverges due to the significant drifts and the overconfident belief in the noisy map. Even for the non-divergent situation (4Seasons), VINS-Fusion also gives poor performance. On the contrary, our proposed method can perform localization in all of these scenarios, and the ECU mechanism improves the accuracy effectively because the long-term absence of map-based measurements occurs. Combining multiple frames matching information (MM) and error compensated updating (ECU), our method outperforms the others.

**Real-time efficiency** Efficiency is also essential for real-time localization. To show the efficiency of our proposed framework, the time consumption of VINS-Fusion and our method is shown in Fig. 6. One can see that the filter-based solution is more time-saving than VINS-Fusion, thus more appropriate for onboard processing.
VI. CONCLUSION

In this letter, we propose a filter-based localization system, which can consistently and efficiently fuse the pre-built map information to the VIO to bound the drift. To be specific, Schmidt-EKF is employed to reduce the computational complexity while maintaining the map uncertainty. According to the observability analyses, the first-estimate Jacobian technique is proposed to guarantee the dimension of the unobservable subspace being desired four. Moreover, we introduce an observability-constrained updating method to compensate for the significant drift after a long-term absence of map-based measurements, which is useful, especially for the long-term driving in large-scale scenes.

REFERENCES

[1] X. Ding et al., “Persistent stereo visual localization on cross-modal invariant map,” *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 11, pp. 4646–4658, Nov. 2020, doi: 10.1109/TITS.2019.2942760.
[2] H. Yin, Y. Wang, X. Ding, L. Tang, S. Huang, and R. Xiong, “3D LiDAR-based global localization using siamese neural network,” *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 4, pp. 1380–1392, Apr. 2020, doi: 10.1109/TITS.2019.2905046.
[3] C. Xia et al., “Onboard sensors-based self-localization for autonomous vehicle with hierarchical map,” *IEEE Trans. Cybern.*, early access, Mar. 22, 2022, doi: 10.1109/TCYB.2022.3155724.
[4] Y. Jiao, Y. Wang, X. Ding, M. Wang, and R. Xiong, “Deterministic optimality for robust vehicle localization using visual measurements,” *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 6, pp. 5397–5410, Jun. 22, 2021, doi: 10.1109/TITS.2021.3055374.

[5] A. I. Mourikis and S. I. Roumeliotis, “A multi-state constraint kalman filter for vision-aided inertial navigation,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2007, pp. 3565–3572, doi: 10.1109/ROBOT.2007.364024.

[6] P. Geneva, K. Eckenhofer, W. Lee, Y. Yang, and G. Huang, “OpenVINS: A research platform for visual-inertial estimation,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 4666–4672, doi: 10.1109/ICRA40945.2020.9196524.

[7] T. Qin, P. Li, and S. Shen, “VINS-Mono: A robust and versatile monocular visual-inertial state estimator,” *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, Aug. 2018, doi: 10.1109/TRO.2018.2853729.

[8] T. Qin, S. Cao, J. Pan, and S. Shen, “A general optimization-based framework for global pose estimation with multiple sensors,” 2019. [Online]. Available: https://arxiv.org/abs/1901.03642v1

[9] R. Mascaro, L. Teixeira, T. Hinzmann, R. Siegwart, and M. Chli, “GOMSF: Graph-optimization based multi-sensor fusion for robust UAV pose estimation,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 1421–1428, doi: 10.1109/ICRA.2018.8460193.

[10] W. Lee, K. Eckenhofer, P. Geneva, and G. Huang, “Intermittent GPS-aided VIO: Online initialization and calibration,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 5724–5731, doi: 10.1109/ICRA40945.2020.9197029.

[11] S. Lynen, T. Sattler, M. Bosse, J. Hesch, and R. Siegwart, “Get out of my lab: Large-scale, real-time visual-inertial localization,” in *Proc. Robot.: Sci. Syst. Conf.*, 2015, pp. 13–17.

[12] T. Schneider et al., “Maplab: An open framework for research in visual-inertial mapping and localization,” *IEEE Robot. Automat. Lett.*, vol. 3, no. 3, pp. 1418–1425, Jul. 2018, doi: 10.1109/LRA.2018.2890113.

[13] C. Campos, R. Elvira, J. J. G. Rodriguez, J. M. M. Montiel, and J. D. Tardós, “ORB-SLAM3: An accurate open-source library for visual, visual-inertial and multi-map SLAM,” 2020, Accessed: Jul. 23, 2020. [Online]. Available: https://arxiv.org/abs/2007.11898v1

[14] R. C. DuToit, J. A. Hesch, E. D. Nerurkar, and S. I. Roumeliotis, “Consistent map-based 3D localization on mobile devices,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2017, pp. 6253–6260, doi: 10.1109/ICRA.2017.7899741.

[15] P. Geneva, K. Eckenhofer, and G. Huang, “A linear-complexity EKF for visual-inertial navigation with loop closures,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2019, pp. 3535–3541, doi: 10.1109/ICRA.2019.8793836.

[16] S. Lynen, M. W. Achtelik, S. Weiss, M. Chli, and R. Siegwart, “A robust and modular multi-sensor fusion approach applied to MAP navigation,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2013, pp. 3923–3929, doi: 10.1109/IROS.2013.6696917.

[17] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, “Analysis and improvement of the consistency of extended Kalman filter based SLAM,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2008, pp. 473–479, doi: 10.1109/ROBOT.2008.4543252.

[18] G. Mascaro, L. Teixeira, T. Hinzmann, R. Siegwart, and M. Chli, “GOMSF: Graph-optimization based multi-sensor fusion approach applied to MA V navigation,” in *Proc. DAGM German Conf. Pattern Recognit.*, 2019, pp. 404–417.

[19] B. Siciliano et al., “EuRoC—the challenge initiative for European robotics,” in *Proc. IEEE ISRR/Robotik 41st Int. Symp. Robot.*, 2014, pp. 1–7.

[20] J. Jeong et al., “Complex urban dataset with multi-level sensors from highly diverse urban environments,” *Int. J. Robot. Res.*, vol. 38, no. 6, pp. 642–657, 2019.

[21] Y. Jiao, Y. Wang, X. Ding, B. Fu, S. Huang, and R. Xiong, “2-Entity random sample consensus for robust visual localization: Framework, methods, and verifications,” *IEEE Trans. Int. Electron.*, vol. 68, no. 5, pp. 4519–4528, May 2021, doi: 10.1109/TIE.2020.2984070.

[22] R. Wenzel et al., “4Seasons: A cross-season dataset for multi-weather SLAM in autonomous driving,” in *Proc. DAGM German Conf. Pattern Recognit.*, 2020, pp. 404–417.

[23] B. M. Bell and F. W. Cathey, “The iterated Kalman filter update as a Gauss-Newton method,” *IEEE Trans. Autom. Control*, vol. 38, no. 2, pp. 294–297, Feb. 1993, doi: 10.1109/9.250476.

[24] Z. Zhang et al., “Supplementary material: Map-based visual-inertial localization: Consistency and complexity,” 2022. [Online]. Available: https://drive.google.com/file/d/1ecphDnXhtqTgsAxzk3wTKl8lDqxBy6Q/view?usp=sharing