FEJ2: A Consistent Visual-Inertial State Estimator Design

Chuchu Chen, Yulin Yang, Patrick Geneva, and Guoquan Huang

Abstract— **In this paper, we propose a novel consistent state estimator design for visual-inertial systems. Motivated by first-estimates Jacobian (FEJ) based estimators – which uses the first-ever estimates as linearization points to preserve proper observability properties of the linearized estimator thereby improving the consistency – we carefully model measurement linearization errors due to its Jacobian evaluation and propose a methodology which still leverages FEJ to ensure the estimator's observability properties, but additionally explicitly compensate for linearization errors caused by poor *first* estimates. We term this estimator FEJ2, which directly addresses the discrepancy between the best Jacobian evaluated at the *latest* state estimate and the first-estimates Jacobian evaluated at the *first-time-ever* state estimate. We show that this process explicitly models that the FEJ used is imperfect and thus contributes additional error which, as in FEJ2, should be modeled and consistently increase the state covariance during update. The proposed FEJ2 is evaluated against state-of-the-art visual-inertial estimators in both Monte-Carlo simulations and real-world experiments, which has been shown to outperform existing methods and to robustly handle poor first estimates and high measurement noises.**

I. Introduction

In the past decades, visual-inertial navigation systems (VINS) have become eminent in robotics, AR/VR, and autonomous applications due to its low-cost, small form, and complementary nature [1]. VINS looks to fuse platform dynamic information from an inertial measurement unit (IMU) and aiding camera sensor(s) to produce an accurate and robust 6 degrees-of-freedom (d.o.f) pose and its corresponding uncertainty. Two classes of estimator designs typically used are nonlinear optimization-based approaches [2]–[5] and light-weight filter-based ones (e.g., an extended Kalman filter (EKF)) [6]–[10]. The latter remains popular due to its efficiency, simplicity, and accuracy.

The observability and consistency of such estimators have attracted significant research efforts due to its ability to provide: (i) the minimal conditions for initialization, (ii) insights into what states are unrecoverable, and (iii) degenerate motions which can hurt the performance of the system [11]–[14]. An estimator is consistent when its errors are zero-mean (unbiased) and the covariance matrix is equal to that reported by the estimator [15, Section 5.4]. Consistent estimators are of particular interest for real-world applications as the uncertainty of the pose is crucial for safe navigation through unknown environments. EKF-based VINS estimators have been shown to be *inconsistent* due to spurious information gain along unobservable directions and have required the creation of "observability aware" filters which explicitly correct for these inaccurate information gains causing filter over-confidence [7], [8], [16].

From the system observability perspective, the standard EKF VINS should not gain information in the unobservable subspace which has been analytically shown to correspond to the 4 d.o.f global yaw and position of the platform [13]. However, linearization discrepancies of the nonlinear system models cause systems to mistakenly gain information along unobservable directions. For example, when Jacobians are evaluated at the *current* state estimates, the linearized error-state system has one less unobservable direction than the underlying nonlinear system, causing information gain along the global yaw of the platform and reporting an over-confident system covariance [7].

To solve this issue, estimators such as robocentric VINS [9], [17], invariant filters [18]–[21], observability-constrained [8], and first-estimates Jacobian [7], [16], [22] methodologies have been proposed. The robocenteric VINS [9], [17] reformulates their estimation problem with respect to a moving local frame, rather than the fixed global frame which alleviates inconsistency issues, but requires extra states to be estimated and causes more unobservable directions. Invariant VINS [18]–[20], [23] associates filter uncertainty to an invariant error state on the manifold, which does not change under any stochastic observable transformation, thus maintaining consistency. However, if features are maintained in the state vector, the propagation of it will become computationally expensive because these features are involved in the covariance propagation [21]. Observability-constrained (OC) estimators [8] guarantee the observability properties by directly enforcing and maintaining the system unobservable subspace (e.g., the observability matrix's nullspace). The direct modifications of their state transition and Jacobian matrices ensure no gain of spurious information while also allowing for the best estimate to be used during linearization. Nonetheless, the Jacobians used do not strictly follow the first-order Taylor series expansion and thus are not theoretically the optimal.

First-estimates Jacobian (FEJ) [7], [16], [22] methodologies guarantee the observability properties by observing that if the linearized system state transition matrix and Jacobians are evaluated at the same estimate over all time periods, the system maintains its observability properties. In practice, a natural choice of linearization point is to use the *first* state estimate for all future timesteps, thus ensuring that the 4 d.o.f unobservable VINS subspace does not gain spurious information. One crucial downside of FEJ, which is discussed in detail in the following sections, is that the Jacobian can be erroneous due to the poor first state estimates. This can cause non-negligible linearization errors, as compared

This work was partially supported by the University of Delaware (UD) College of Engineering, the NSF IIS-1924897, the ARL (W911NF-19-2-0226, W911NF-20-2-0098), and Google ARCore. P. Geneva is also supported by the UD University Doctoral Fellowship.

The authors are with the Robot Perception and Navigation Group (RPNG), University of Delaware, Newark, DE 19716, USA. Email: {ccchu,yuyang,pgeneva,ghuang}@udel.edu

to using the best estimate, and hurt estimator accuracy and consistency since an unmodeled source of error is being introduced [24]. Ideally, we would want to be able to use the best estimate during linearization or take into account this additional source of error while also ensuring observability properties. The proposed FEJ2 methodology looks to directly address this weakness. The main contributions of this work include:

- We propose a novel visual-inertial state estimator design, named FEJ2, which addresses the sensitivities of FEJ to poor initial estimates. The proposed FEJ2 ensures both accurate and consistent estimation while being robust to large initialization errors.
- We explicitly model the linearization errors that FEJ neglects, due to using the first estimate, to ensure the estimator models all sources of error. The proposed FEJ2 is theoretically proven to consistently estimate the state covariance during update.
- A visual-inertial estimator which leverages FEJ2 is presented and verified in both Monte-Carlo simulations and real-world experiments against existing state-of-the-art consistent VINS estimators. FEJ2 is shown to outperform existing methods and is able to robustly handle poor first estimates and high measurement noises.

II. Visual-Inertial Systems

In this section, we briefly overview the EKF-based visual-inertial system which fuses IMU readings and environmental feature measurements.

A. Propagation

The state vector $\mathbf{x}$ consists of the IMU state $\mathbf{x}_I$ and the feature state $\mathbf{x}_f$.

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix}\mathbf{x}_I^\top & \mathbf{x}_f^\top\end{bmatrix}^\top &(1)\\ &= \begin{bmatrix}{}^I_G\bar{q}^\top & {}^G\mathbf{p}_I^\top & {}^G\mathbf{v}_I^\top & \mathbf{b}_g^\top & \mathbf{b}_a^\top \mid {}^G\mathbf{p}_{f_1}^\top \cdots {}^G\mathbf{p}_{f_M}^\top\end{bmatrix}^\top\end{aligned}$$

where ${}^I_G\bar{q}$ is the unit quaternion[1] that represents the rotation ${}^I_G\mathbf{R}$ from global frame $\{G\}$ to the IMU frame $\{I\}$; ${}^G\mathbf{p}_I$ and ${}^G\mathbf{v}_I$ are the IMU position and velocity in $\{G\}$, respectively; $\mathbf{b}_g$ and $\mathbf{b}_a$ are the gyroscope and accelerometer biases; and the feature state $\mathbf{x}_f$ comprises the global position of M landmarks. We use $\mathbf{x}_k$ to represent the state at timestep t_k. The IMU kinematics are used to evolve the state from time t_k to t_{k+1} [25]:

$$\mathbf{x}_{I_{k+1}} = \mathbf{f}(\mathbf{x}_{I_k}, \mathbf{a}_{m_k}, \boldsymbol{\omega}_{m_k}) \tag{2}$$

where the linear acceleration $\mathbf{a}_{m_k}$ and angular velocity $\boldsymbol{\omega}_{m_k}$ measurements are contaminated by zero-mean white Gaussian noises. Features remain static, and have zero dynamics ${}^G\dot{\mathbf{p}}_{f_i} = \mathbf{0}$. With the linearized state transition matrix $\hat{\boldsymbol{\Phi}}(k+1, k)$ from time t_k to t_{k+1} computed using *current* state estimates, the IMU noise covariance $\mathbf{Q}_k$ and state covariance propagation can be written as:

$$\mathbf{P}_{k+1|k} = \hat{\boldsymbol{\Phi}}(k+1,k)\mathbf{P}_{k|k}\hat{\boldsymbol{\Phi}}^\top(k+1,k) + \mathbf{Q}_k \tag{3}$$

$$\mathbf{Q}_k = \int_{t_k}^{t_{k+1}} \hat{\boldsymbol{\Phi}}(k+1,\tau)\mathbf{G}(\tau)\mathbf{Q}\mathbf{G}^\top(\tau)\hat{\boldsymbol{\Phi}}^\top(k+1,\tau)\mathrm{d}\tau$$

where $\mathbf{G}$ is the Jacobian matrix with respect to noise and $\mathbf{P}_{k+1|k}$ represents the covariance at timestep t_{k+1} computed using measurements up to timestep t_k. Full details can be found in the companion technical report [27].

B. Measurement Update

Assuming a calibrated perspective camera, the bearing measurement of the ith feature at timestep t_{k+1} can be related to the state by the following:

$$\mathbf{z}_{k+1} = \mathbf{h}(\mathbf{x}_{k+1}) + \mathbf{n}_{k+1} =: \boldsymbol{\Lambda}({}^{C_{k+1}}\mathbf{p}_{f_i}) + \mathbf{n}_{k+1} \tag{4}$$

$$\boldsymbol{\Lambda}([x\ y\ z]^\top) = [x/z\ y/z]^\top, \quad {}^{C_{k+1}}\mathbf{p}_{f_i} = \begin{bmatrix}x & y & z\end{bmatrix}^\top \tag{5}$$

$${}^{C_{k+1}}\mathbf{p}_{f_i} = {}^C_I\mathbf{R}\,{}^{I_{k+1}}_G\mathbf{R}\left({}^G\mathbf{p}_{f_i} - {}^G\mathbf{p}_{I_{k+1}}\right) + {}^C\mathbf{p}_I \tag{6}$$

where $\boldsymbol{\Lambda}(\cdot)$ is the camera perspective projection model; $\mathbf{n}_{k+1} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{k+1})$ is the white Gaussian noise; $\{{}^C_I\mathbf{R}, {}^C\mathbf{p}_I\}$ are the camera-IMU transformation. Linearizing Eq. (4) with respect to the *current* state estimate $\hat{\mathbf{x}}_{k+1}$ we get:

$$\mathbf{z}_{k+1} \simeq \mathbf{h}(\hat{\mathbf{x}}_{k+1}) + \hat{\mathbf{H}}_{k+1}(\mathbf{x}_{k+1} \boxminus \hat{\mathbf{x}}_{k+1}) + \mathbf{n}_{k+1} \tag{7}$$

$$\Rightarrow \mathbf{r}_{k+1} \triangleq \mathbf{z}_{k+1} - \mathbf{h}(\hat{\mathbf{x}}_{k+1}) \simeq \hat{\mathbf{H}}_{k+1}\tilde{\mathbf{x}}_{k+1} + \mathbf{n}_{k+1} \tag{8}$$

where $\hat{\mathbf{H}}_{k+1}$ denotes the Jacobian evaluated at $\hat{\mathbf{x}}_{k+1}$. $\hat{\mathbf{H}}_{k+1}$ only contains non-zero blocks for the pose and the ith feature and, thus is computed as ($k+1$ subscripts are dropped for brevity):

$$\hat{\mathbf{H}} = \nabla\hat{\mathbf{h}}_i \begin{bmatrix}\hat{\mathbf{H}}_\theta & \hat{\mathbf{H}}_p & \mathbf{0}_{3\times 9} \mid \cdots & \hat{\mathbf{H}}_{f_i} \cdots\end{bmatrix} \tag{9}$$

$$\hat{\mathbf{H}}_\theta = \lfloor {}^I_G\hat{\mathbf{R}}\left({}^G\hat{\mathbf{p}}_{f_i} - {}^G\hat{\mathbf{p}}_I\right)\rfloor, \ \hat{\mathbf{H}}_p = -\hat{\mathbf{H}}_f = -{}^I_G\hat{\mathbf{R}} \tag{10}$$

$$\nabla\hat{\mathbf{h}}_i = \frac{1}{\hat{z}_i{}^2}\begin{bmatrix}\hat{z}_i & 0 & -\hat{x}_i\\ 0 & \hat{z}_i & -\hat{y}_i\end{bmatrix} {}^C_I\mathbf{R} \tag{11}$$

This residual can then directly update the state [28].

C. Observability Analysis

System observability plays a crucial role in state estimation [11], [29]. Understanding system observability provides a deep insight about the system's geometrical properties and determines the minimal measurement modalities needed. With the state transition matrix, Eq. (3), and measurement Jacobian, Eq. (8), we construct the observability matrix [13]:

$$\mathcal{O} \triangleq \begin{bmatrix}\mathbf{H}_0\boldsymbol{\Phi}(0,0)\\ \mathbf{H}_1\boldsymbol{\Phi}(1,0)\\ \vdots\\ \mathbf{H}_{k+1}\boldsymbol{\Phi}(k+1,0)\end{bmatrix} \tag{12}$$

If $\mathcal{O}$ is of full column rank, the system is fully observable. However, VINS is partial observable with a nullspace $\mathbf{N}$ satisfying $\mathcal{O}\mathbf{N} = \mathbf{0}$ (see [13], [16]). The nullspace $\mathbf{N}$ describes the state unobservable subspace can not be recovered with measurements.

The nullspace $\mathbf{N}$ for VINS should be of *4* d.o.f and relates to the global yaw and translation [13]. The standard EKF, which always computes the state transition $\hat{\boldsymbol{\Phi}}(k,0)$ and measurement Jacobian $\hat{\mathbf{H}}_k$ using the current state estimates, makes the global orientation appear to be observable and thus reduces the nullspace to only *3* d.o.f dimension [7].

[1] Throughout the paper $\hat{\mathbf{x}}$ is used to denote the *current* best estimate of a random variable $\mathbf{x}$ with $\tilde{\mathbf{x}} = \mathbf{x} \boxminus \hat{\mathbf{x}}$ denotes the error state. For the quaternion error state, we employ JPL multiplicative error [25] and use $\delta\boldsymbol{\theta} \in \mathbb{R}^3$ defined by the error quaternion i.e., $\delta\bar{q} = \bar{q} \otimes \hat{\bar{q}}^{-1} \simeq [\frac{1}{2}\delta\boldsymbol{\theta}^\top\ 1]^\top$. The "$\boxplus$" and "$\boxminus$" operations map elements to and from a given manifold and equate to simple "+" and "-" for vector variables [26].

This causes the filter gain *extra* information and become overconfident with estimation results. Moreover, a valid state transition matrix should have the semi-group property [30]:

$$\mathbf{\Phi}(k+1,k-1)=\mathbf{\Phi}(k+1,k)\mathbf{\Phi}(k,k-1) \tag{13}$$

$\hat{\mathbf{\Phi}}$ [see Eq. (3)] evaluated at *current* state estimate violates this property (see [7], [30, Lemma 4.1]). The above issues cause inconsistency and inaccuracy.

III. Consistent Visual-Inertial Estimator Design

Leveraging the observability-based consistent estimator designs [16] and motivated by the great success of such estimators in visual-inertial systems [7], [8], we aim to improve this observability-based design by addressing its significant caveat of (potentially) large linearization errors. To this end, in what follows, after reviewing the observability-based methodology, we present in detail the proposed FEJ2 consistent estimator design methodology.

A. Observability-based Methodology

1) OC-VINS: OC [8], [13] maintains the proper system observability properties by enforcing the initial (and thus the current) unobservable nullspace $\mathbf{N}_0$ which spans *correct* system unobservable directions:

$$\mathbf{H}_k\mathbf{\Phi}(k,0)\mathbf{N}_0=\mathbf{0}, \quad \forall k\geq 0 \tag{14}$$

This property is maintained through finding the best $\mathbf{\Phi}'(k+1,k)$ and $\mathbf{H}_k^{'}$ such that:

$$\mathbf{N}_{k+1}=\mathbf{\Phi}^{'}(k+1,k)\mathbf{N}_k \ , \quad \mathbf{H}^{'}_k\mathbf{N}_k=\mathbf{0}, \quad \forall k\geq 0 \tag{15}$$

where $\mathbf{N}_k$ denotes the system unobservable subspace at timestep k *propagated* from the initial chosen $\mathbf{N}_0$. Hence, the system maintains *correct* unobservable directions with $\mathbf{\Phi}'(k+1,k)$ and $\mathbf{H}^{'}_k$. The modified $\mathbf{\Phi}'(k+1,k)$ becomes a valid state translation matrix with the semi-group property (e.g., it satisfies Eq. (13)). It is important to note that OC heavily relies on $\mathbf{N}_0$ and poor initial estimates in $\mathbf{N}_0$ will hurt the system performance.

2) FEJ-VINS: FEJ-based VINS [7], [22] directly select the first ever available estimate, denoted by $\bar{\mathbf{x}}$, for each state variable as linearization points at all future timesteps. During propagation, FEJ evaluates the state translation matrix $\bar{\mathbf{\Phi}}(k+1,k)$ with *propagated* state estimates instead of the *updated* ones to ensure $\bar{\mathbf{\Phi}}(k+1,k)$ has a valid semi-group property (e.g., satisfies Eq. (13) [30, Lemma 4.2]). FEJ also selects the first state estimates $\bar{\mathbf{x}}$ to construct measurement Jacobian $\bar{\mathbf{H}}_{k+1}$ and its linearized residual as:

$$\bar{\mathbf{r}}_{k+1}\simeq\mathbf{z}_{k+1}-\mathbf{h}(\hat{\mathbf{x}}_{k+1})\simeq\bar{\mathbf{H}}_{k+1}\tilde{\mathbf{x}}_{k+1}+\mathbf{n}_{k+1} \tag{16}$$

Since there are no linearization point changes, the unobservable subspace dimensions are automatically preserved, which can be verified as follows:

$$\bar{\mathbf{H}}_{k+1}\bar{\mathbf{\Phi}}(k+1,0)\mathbf{N}_0=\mathbf{0}, \quad \forall k\geq 0 \tag{17}$$

We now show that the underlying FEJ assumption can possibly incur large linearization errors by re-deriving the linearized measurement with a first order Taylor series expansion at $\bar{\mathbf{x}}$ (subscript $k+1$ dropped for clarity):

$$\begin{aligned}\mathbf{z}&\simeq\mathbf{h}(\bar{\mathbf{x}})+\bar{\mathbf{H}}(\mathbf{x}\boxminus\bar{\mathbf{x}})+\mathbf{n} && (18)\\ &=\mathbf{h}(\bar{\mathbf{x}})+\bar{\mathbf{H}}(\hat{\mathbf{x}}\boxminus\bar{\mathbf{x}})+\bar{\mathbf{H}}(\mathbf{x}\boxminus\hat{\mathbf{x}})+\mathbf{n} && (19)\\ &\simeq\mathbf{h}(\hat{\mathbf{x}})+\bar{\mathbf{H}}\tilde{\mathbf{x}}+\mathbf{n} && (20)\end{aligned}$$

FEJ estimator uses Eq. (20) during update (both residual and error state are at the current estimates). Although $\bar{\mathbf{H}}$ ensures the system unobservable subspace, it might lead the system to suffer from un-optimal updates due to poor $\bar{\mathbf{x}}$. Specifically, FEJ assumes the following:

$$\mathbf{h}(\hat{\mathbf{x}})\simeq\mathbf{h}(\bar{\mathbf{x}})+\bar{\mathbf{H}}(\hat{\mathbf{x}}\boxminus\bar{\mathbf{x}}) \tag{21}$$

This approximation can introduce extra linearization errors caused by the estimate change $\hat{\mathbf{x}}\boxminus\bar{\mathbf{x}}$, especially when *poor* initial estimates $\bar{\mathbf{x}}$ are used. This motivates the proposed FEJ2 which looks to directly take into account this additional linearization error.

B. FEJ2 Methodology

The above estimator designs from observability perspective avoid spurious information erroneously flooding into the unobservable directions of the state space, thus improving consistency. As they rely on initial estimates to construct Jacobians, the estimator may suffer from the poor state initialization. To address this issue, we seek to linearize the system at the *current* state estimates $\hat{\mathbf{x}}$ for the smallest linarization errors and use the *first-estimates Jacobian* $\bar{\mathbf{H}}$ to avoid extra information gain along unobservable directions.

To achieve this, FEJ2 derives a more accurate linear model to approximate the nonlinear system as:

$$\begin{aligned}\mathbf{z}_{k+1}&\simeq\mathbf{h}(\hat{\mathbf{x}}_{k+1})+\hat{\mathbf{H}}_{k+1}\tilde{\mathbf{x}}_{k+1}+\mathbf{n}_{k+1} && (22)\\ &=\mathbf{h}(\hat{\mathbf{x}}_{k+1})+(\bar{\mathbf{H}}_{k+1}+\hat{\mathbf{H}}_{k+1}-\bar{\mathbf{H}}_{k+1})\tilde{\mathbf{x}}_{k+1}+\mathbf{n}_{k+1}\\ \hat{\mathbf{r}}_{k+1}&=\mathbf{z}_{k+1}-\mathbf{h}(\hat{\mathbf{x}}_{k+1}) && (23)\\ &\simeq\bar{\mathbf{H}}_{k+1}\tilde{\mathbf{x}}_{k+1}+\Delta\mathbf{H}_{k+1}\tilde{\mathbf{x}}_{k+1}+\mathbf{n}_{k+1} && (24)\end{aligned}$$

where $\Delta\mathbf{H}_{k+1}=\hat{\mathbf{H}}_{k+1}-\bar{\mathbf{H}}_{k+1}$ indirectly captures linearization points change between the *first* and *current* state estimates. Therefore, we project Eq. (24) onto the left nullspace of $\Delta\mathbf{H}$ using the QR decomposition, i.e., $\Delta\mathbf{U}_{k+1}^{\top}\Delta\mathbf{H}_{k+1}=\mathbf{0}$.

$$\Delta\mathbf{H}_{k+1}=\begin{bmatrix}\Delta\mathbf{Q}_{k+1} & \Delta\mathbf{U}_{k+1}\end{bmatrix}\begin{bmatrix}\Delta\mathbf{T}_{k+1}\\ \mathbf{0}\end{bmatrix} \tag{25}$$

where $\begin{bmatrix}\Delta\mathbf{Q}_{k+1} & \Delta\mathbf{U}_{k+1}\end{bmatrix}$ is a unitary matrix, columns of $\Delta\mathbf{Q}_{k+1}$ and $\Delta\mathbf{U}_{k+1}$ form basis for the range and nullspace of $\Delta\mathbf{H}_{k+1}$, respectively. $\Delta\mathbf{T}_{k+1}$ is an upper triangular matrix. Multiplying $\Delta\mathbf{U}_{k+1}$ to Eq. (24) results in the following the measurement residual $\mathbf{r}^{*}_{k+1}$ and Jacobian $\mathbf{H}^{*}_{k+1}$:

$$\begin{aligned}&\Delta\mathbf{U}_{k+1}^{\top}\hat{\mathbf{r}}_{k+1}=\Delta\mathbf{U}_{k+1}^{\top}\bar{\mathbf{H}}_{k+1}\tilde{\mathbf{x}}_{k+1}+\Delta\mathbf{U}_{k+1}^{\top}\mathbf{n}_{k+1}\\ &\Rightarrow \quad \mathbf{r}^{*}_{k+1}=\mathbf{H}^{*}_{k+1}\tilde{\mathbf{x}}_{k+1}+\mathbf{n}^{*}_{k+1} && (26)\end{aligned}$$

where $\mathbf{n}_{k+1}$ is zero mean Gaussian noise with covariance $\mathbf{R}_{k+1}$ [31] and $\mathbf{n}^{*}\sim\mathcal{N}(\mathbf{0},\Delta\mathbf{U}_{k+1}^{\top}\mathbf{R}_{k+1}\Delta\mathbf{U}_{k+1})$. We then update the filter with the modified $\mathbf{r}^{*}_{k+1}$ and $\mathbf{H}^{*}_{k+1}$.

As evident from the above, the QR factorization is leveraged to find the left nullspace $\Delta\mathbf{U}$ and marginalize $\Delta\mathbf{H}$ via $\Delta\mathbf{U}^{\top}\Delta\mathbf{H}=\mathbf{0}$ (subscript $k+1$ dropped for clarity). In order to find such nullspace, $\Delta\mathbf{H}_{m\times n}$ is required to be full column rank, where m and n are the size of measurements and states, respectively. Assuming $\texttt{rank}(\Delta\mathbf{H})=n$, by the rank-nullity theorem, we find $\texttt{nullity}(\Delta\mathbf{U})=m-n$, that is, the dimension of the left nullspace $\dim(\Delta\mathbf{U})$ and also the remaining measurement residual size [$\mathbf{r}^*$ in Eq. (24)] after the nullspace operation. While this is the case for many multi-view scenarios, some systems may have less visual observations than the dimensions of the state (i.e., $m \not> n$). For example, assuming M environmental features are observed by monocular camera [see Eq. (4)], we can find the size of measurements $m = 2M$, while the state consists the 6 d.o.f IMU pose and the 3D feature positions, thus $n = 6 + 3M$. We notice that $2M \not> 6 + 3M, \forall M \geq 0$. If this occurs, it is not difficult to address in an engineering sound way as discussed in Section III-D.2.

C. FEJ2 Properties

We now show that FEJ2 not only preserves the proper observability properties but compensates for large linearization errors as in FEJ-based estimators. Note that, in what follows, we drop the subscripts for clarity.

Lemma 3.1: FEJ2 employs a linearized system model that has an unobservable subspace of correct dimensions and structure, and shares the same (initial) nullspace of the observability matrix as the FEJ and OC.

Proof: The observability matrix of the FEJ ($\bar{\mathcal{O}}$) and OC ($\mathcal{O}'$) assume the following conditions:

$$\text{OC:}\ \ \mathcal{O}'\mathbf{N}_0 = \mathbf{0} \qquad \text{FEJ:}\ \ \bar{\mathcal{O}}\mathbf{N}_0 = \bar{\mathbf{H}}\bar{\mathbf{\Phi}}\mathbf{N}_0 = \mathbf{0} \tag{27}$$

where $\mathbf{N}_0$ is the selected initial system unobservable subspace introduced in Section III-A.1. It is not difficult to show that FEJ2 shares the same structure of observability matrix with FEJ and OC:

$$\mathcal{O}^*\mathbf{N}_0 = \mathbf{H}^*\bar{\mathbf{\Phi}}\mathbf{N}_0 = \Delta\mathbf{U}^\top\bar{\mathbf{H}}\bar{\mathbf{\Phi}}\mathbf{N}_0 = \Delta\mathbf{U}^\top\bar{\mathcal{O}}\mathbf{N}_0 = \mathbf{0}$$

This proves that FEJ2 keeps the *correct* unobservable subspace. ■

Lemma 3.2: FEJ2 has larger covariance estimates than FEJ.

Proof: Without losing generality, we assume the noise covariance is isotropic and identical (i.e., $\mathbf{R} = \mathbf{I}$), which however can be easily extended to general cases by noise whitening. After QR factorization for $\Delta\mathbf{H}$ as in Eq. (25), due to the unitary property we have $\Delta\mathbf{Q}\Delta\mathbf{Q}^\top + \Delta\mathbf{U}\Delta\mathbf{U}^\top = \mathbf{I}$. The information matrix of FEJ2 can then be derived as:

$$\begin{align} \mathbf{\Sigma}^* &= \mathbf{H}^{*\top}\mathbf{H}^* = (\Delta\mathbf{U}^\top\bar{\mathbf{H}})^\top(\Delta\mathbf{U}^\top\bar{\mathbf{H}}) \tag{28} \\ &= \bar{\mathbf{H}}^\top\bar{\mathbf{H}} - \bar{\mathbf{H}}^\top\Delta\mathbf{Q}\Delta\mathbf{Q}^\top\bar{\mathbf{H}} \tag{29} \\ &\triangleq \bar{\mathbf{\Sigma}} - \Delta\mathbf{\Sigma} \tag{30} \end{align}$$

Notice that $\bar{\mathbf{\Sigma}} = \bar{\mathbf{H}}^\top\bar{\mathbf{H}}$ is the FEJ information matrix, $\Delta\mathbf{\Sigma}$ is a positive semidefinite matrix. Eq. (30) shows FEJ2 deflates the information matrix of FEJ, which, in turn, inflates the covariance matrix $\mathbf{P}$ of system since $\mathbf{P} = \mathbf{\Sigma}^{-1}$. ■

Basically, FEJ2 projects the measurement residual function onto the left nullspace of $\Delta\mathbf{H}$. As $\Delta\mathbf{H}$ is the difference between Jacobians evaluate with $\bar{\mathbf{x}}$ and $\hat{\mathbf{x}}$, a general consideration is it represents the linearization errors. In addition, since $\hat{\mathbf{H}} = \bar{\mathbf{H}} + \Delta\mathbf{H}$, where $\bar{\mathbf{H}}$ ensures that the linearized system keeps the same observability property as the underlying nonlinear system, $\Delta\mathbf{H}$ causes *extra* information gain along unobservable directions. At this point, we see that FEJ2 is a more consistent and accurate estimator which guards the system observability properties and better models the system uncertainty compared to FEJ.

D. FEJ2 Applications

FEJ2 can be implemented to different VINS frameworks as a base estimator design. In this section we continue to address the FEJ2 nullpace projection operation and explain how to utilize FEJ2 idea into SLAM and multi-state constraint Kalman filter (MSCKF) based visual-inertial odometry (VIO).

1) FEJ2-EKF SLAM: In the SLAM system, the state vector includes the current IMU state and the features Eq. (1). The standard measurement Jacobians are showed in Eq. (10). To simplify the discussion we assume the robot directly receives relative position measurements of features, thus $\nabla\mathbf{h} = \mathbf{I}$. In the evaluation of the first-estimates Jacobian $\bar{\mathbf{H}}_{k+1}$, we always utilize the feature estimates from the first time they were observed and initialized, if the first estimate value of the ith feature is $\bar{\mathbf{p}}_{f_i}$, we have:

$$\bar{\mathbf{H}}_{\theta_i} = \lfloor {}^I_G\bar{\mathbf{R}}({}^G\bar{\mathbf{p}}_{f_i} - {}^G\bar{\mathbf{p}}_I)\rfloor \quad \bar{\mathbf{H}}_{p_i} = -\bar{\mathbf{H}}_{f_i} = -{}^I_G\bar{\mathbf{R}} \tag{31}$$

Note that in EKF-SLAM, the IMU pose estimates $\{{}^I_G\hat{\mathbf{R}}, {}^G\hat{\mathbf{p}}_I\}$ are only used one-time for system linearization and evolve to the next timetep immediately, hence ${}^I_G\bar{\mathbf{R}} = {}^I_G\hat{\mathbf{R}}$ and ${}^G\bar{\mathbf{p}}_I = {}^G\hat{\mathbf{p}}_I$. Since only the IMU state evolves in $\bar{\mathbf{H}}_{p_i}$ and $\bar{\mathbf{H}}_{f_i}$, $\Delta\mathbf{H}_{f_i} = \Delta\mathbf{H}_{p_i} = \mathbf{0}$. Stacking measurements of all detected features at timestep t_{k+1}, the structure of $\Delta\mathbf{H}$ [27] is:

$$\Delta\mathbf{H} = \begin{bmatrix} \Delta\mathbf{H}_{\theta_{m\times 3}} & \cdots & \mathbf{0} & \cdots & \end{bmatrix}_{m\times n} \tag{32}$$

As $\Delta\mathbf{H}_\theta$ is the only non-zero matrix block, we apply the FEJ2 nullspace operation [see Eq. (26)] to $\Delta\mathbf{H}_\theta$ instead of the full state. It relaxes the matrix rank requirement and only needs $m > 3$, which is common in practice.

2) FEJ2-MSCKF VIO: Within a MSCKF VIO framework, the state vector includes the latest IMU state $\mathbf{x}_I$ a sliding window of cloned poses and features [6]. Since these MSCKF features are marginalized instantly without relinearization in the future, FEJ, FEJ2 or standard EKF updates for them are identical.

More attention is paid to the SLAM features which can be reliably tracked longer than the current sliding window. Those features are initialized into the active state vector and used for update until lost tracking. The state vector at timestep t_{k+1} thus include:

$$\mathbf{x}_{k+1} = \begin{bmatrix} \mathbf{x}_{I_{k+1}}^\top & \mathbf{x}_{\text{clone}}^\top & | & \cdots & {}^G\mathbf{p}_{f_i}^\top & \cdots & \end{bmatrix} \tag{33}$$

where $\mathbf{x}_{\text{clone}}$ refers to all the active cloned poses. As normally the measurements from latest image just contain the Jacobians for the current IMU pose and the feature states, the structure of $\Delta\mathbf{H}$ with respect to $\mathbf{x}_{I_{k+1}}$ and SLAM features $\mathbf{x}_f$ can be simplified as:

$$\Delta\mathbf{H} = \begin{bmatrix} \Delta\mathbf{H}_I & \Delta\mathbf{H}_f \end{bmatrix} = \begin{bmatrix} \Delta\mathbf{H}_\theta & \Delta\mathbf{H}_p & \Delta\mathbf{H}_f \end{bmatrix}_{m\times n}$$

However, as mentioned before, $\Delta\mathbf{H}$ will not be a full column rank matrix if the sensor platform only receives monocular measurements for state update. Even though the left nullspace for $\Delta\mathbf{H}$ exists with stereo measurements, the numerical issue occasionally happen from our experiences. We highlight that $\Delta\mathbf{H}_f$ is a *small* and *sparse* matrix [see

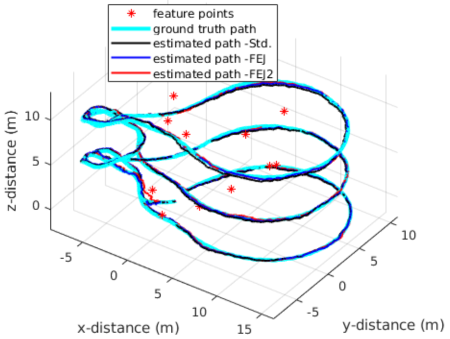

Fig. 1: A typical run for proposed estimators on udel_gore dataset.

TABLE I: The average RMSE and NEES over 50 Monte-Carlo simulations for the relative position measurement model with different estimators.

Estimator	RMSE Ori. (deg)	RMSE Pos. (m)	NEES Ori.	NEES Pos.
STD	1.458	0.376	4.086	4.896
OC	0.472	0.323	2.397	4.119
FEJ	0.491	0.311	2.493	3.887
FEJ2	**0.446**	**0.305**	**2.287**	**3.692**

Eq. (10)], each block row of $\Delta\mathbf{H}_f$ only contains one non-zero block to the ith feature, it could lead to numeral instability when employing QR factorization. Another potential challenge of applying FEJ2 is that, the size of measurements reduces from m to $m - n$ after the nullspace operation, Eq. (26), which causes some geometrical information loss.

To this end, we advocate for the FEJ2 design which only project the left nullspace of $\Delta\mathbf{H}_I$ instead of the full $\Delta\mathbf{H}$, where $\Delta\mathbf{H}_I$ is a tall matrix and typically full column rank since multiple features are continuously tracked. As investigated below, applying FEJ2 to only this sub-Jacobian has significant performance gains.

IV. Monte-Carlo Simulations

A series of Monte-Carlo simulations for both SLAM and MSCKF-VIO were conducted under various conditions to validate the preceding theoretical analyses and demonstrate the capability of FEJ2. Due to space limitation, we refer our technical report [27] for more results. In the following sections, we use STD to represent standard estimator in the tables and figures.

A. SLAM Simulation

We first ran a SLAM simulation (as introduced in Section III-D.1) where a robot following the 3D trajectory shown in Fig. 1 continuously observes fixed number of landmarks at every timestep. Accordingly, landmarks will be initialized into state vector at the first time. The OpenVINS simulator [10] is leveraged to simulate IMU readings and 20 environmental features. In this simulation, the robot continuously receives two feature position measurements of each landmark corrupted by perturbations of 18% of the max sensing range. We report the averaged Root Mean Square Error (RMSE) [32] and Normalized Estimation Error Squared (NEES) [15] over 50 Monte-Carlo runs in Table I.

It is clear that FEJ2 achieves the smallest RMSE compared to the other estimators, thus is the most accurate. Ideally, a consistent estimator should have position and orientation NEES values smaller or close to 3. The position and orientation NEES values of FEJ2 are similar to FEJ and OC, around 3, which verify our analysis that FEJ2 is a consistent estimator. We note that both the orientation and position NEES values of FEJ2 are smaller than those of OC and FEJ, which further verify our analysis that FEJ2 has a larger covariance than FEJ. Hence, FEJ2 should be more conservative and make the system less *sensitive* to noise.

B. MSCKF-VIO Simulation

In this section, we present the estimation performance within a hybrid MSCKF-based VIO system [10], [33] using different simulation setups and measurement noises (shown in Table II). Specifically, we first simulate monocular camera measurements with relative low IMU noises, and then a stereo camera with increased IMU noises to further challenge the system.

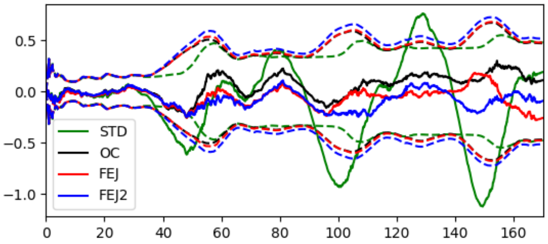

Fig. 3: IMU pitch angle errors (solid lines) and ±3σ bounds (dashed lines) for udel_gore simulations using stereo measurements and 2 pixel measurement noise with different estimators. Note that the ±3σ bounds of the OC (black) and the FEJ (red) are almost identical.

Fig. 3 shows a representative IMU orientation angle estimation error and the corresponding 3σ bounds of uncertainty with stereo camera using different VINS estimators for a single run. As evident, the standard (STD) estimator is inconsistent with estimation errors frequently out of the 3σ bounds. The errors for OC, FEJ and FEJ2 are within their 3σ bounds, indicating their consistency, the 3σ bounds of FEJ2 are slightly broader than the others, showing that FEJ2 is more conservative. These results strengthen our claim that FEJ2 has larger system covariance estimation (Lemma 3.2).

50 run Monte-Carlo simulations are also conducted. Fig. 2 shows the averaged NEES and RMS errors of robot pose with stereo measurements under 2 pixel noise values. While Table III presents the average values of RMSE and NEES when measurement noise are 1 and 3 pixel with both monocular and stereo measurements. Note that in stereo and monocular simulations, different IMU noises are used as showed in Table II.

When measurement noises are small, the average RMSE of OC, FEJ and FEJ2 are very similar to each other since the linearization errors are also small. In the meanwhile, standard (STD) VINS, which is inconsistent, performs the worst with largest estimation errors. When we increase measurement noises, the linearization errors will become more influential and we can observe more apparent accuracy improvements of the FEJ2 compared to other VINS estimators. When testing

TABLE II: Key simulation and estimator parameters for each sensor.

Parameter	Values mono / stereo
Traj. Length (m)	225
IMU Freq. (hz)	400
Camera Freq. (hz)	10
No. SLAM Feat.	50 / 60
Camera Clones	11
Gyro. White Noise	1.6968e-04 / 1.6968e-03
Gyro. Random Walk	1.93963e-05 / 1.93963e-04
Accel. White Noise	2.0000e-03 / 2.0000e-02
Accel. Random Walk	3.0000e-03 / 3.0000e-03

TABLE III: The average RMSE and NEES over 50 Monte-Carlo simulations with simulated monocular and stereo measurements with 1 and 3 pixel noise models.

Noise (pixel)	Est.	RMSE Ori. (deg) mono / stereo	RMSE Pos. (m) mono / stereo	NEES Ori. mono / stereo	NEES Pos. mono / stereo
1	STD	0.412 / 0.344	0.130 / 0.109	23.874 / 15.447	4.911 / 4.874
	OC	0.242 / 0.257	0.119 / 0.100	3.290 / 3.599	3.540 / 3.416
	FEJ	0.242 / 0.256	0.120 / 0.100	3.284 / 3.438	3.617 / 3.322
	FEJ2	**0.238** / **0.238**	**0.118** / **0.095**	**3.150** / **3.324**	**3.443** / **2.965**
3	STD	2.139 / 0.888	0.402 / 0.310	407.221 / 33.852	13.212 / 7.235
	OC	0.716 / 0.723	0.301 / 0.300	3.964 / 4.395	5.051 / 4.839
	FEJ	0.861 / 0.704	0.289 / 0.298	4.965 / 4.163	4.763 / 4.656
	FEJ2	**0.650** / **0.663**	**0.264** / **0.277**	**3.198** / **3.790**	**3.581** / **3.636**

with unrealistic noise (i.e., 8 pixel), FEJ and OC diverge quickly due to the large linearization errors, while FEJ2 can still work and outperform the others.

V. Real-World Experiments

Building upon OpenVINS [10], we further test our FEJ2 based VINS estimators on the Euroc Mav dataset [34] and TUM-VI dataset [35] with both monocular and stereo configurations. Due to the space limitation, we refer our tech-report for more results [27]. In the experiments, we keep 11 clones and at most 50 SLAM features in the state vector. For other tracked features we perform MSCKF update and marginalize them out of state vectors. All estimators were run ten times on each dataset to compensate for randomness and the averaged Absolute Trajectory Error (ATE) [32] values are reported . The results are shown in Table IV. It is clear that OC, FEJ and FEJ2 produce smaller ATE than the standard EKF. Although these three consistent estimators performe very close to each other, FEJ2 still achieves better accuracy on average, especially in monocular camera scenarios. Compared with stereo, FEJ and OC are more likely to suffer from bad feature triangulation due to less visual constraints in the monocular case. This leads to relatively larger linearization errors, which FEJ2 has focused on. These results, along with those from Monte-Carlo simulations presented in the previous section, support that our proposed FEJ2 improves the VINS estimators in both consistency and accuracy.

VI. Conclusions and Future Work

In this work, we develop FEJ2, which is a novel consistent estimator design for VINS. After studying in depth the issues of FEJ-VINS algorithm, we propose to accurately model linearization errors by projecting the measurement residual onto the left nullspace of the error Jacobian. Theoretical proofs are also given to demonstrate this operation keeps the correct unobservable subspace and compensates linearization errors, thus, improve the performance. We implement FEJ2-VINS, results from extensive Monte-Carlo simulations with different configurations and real-world experiments on different datasets show the proposed FEJ2 can achieve better accuracy and consistency than other state-of-the-art estimators. In the future, we plan to investigate how to leverage different feature representations and ramifications of online sensor calibration.

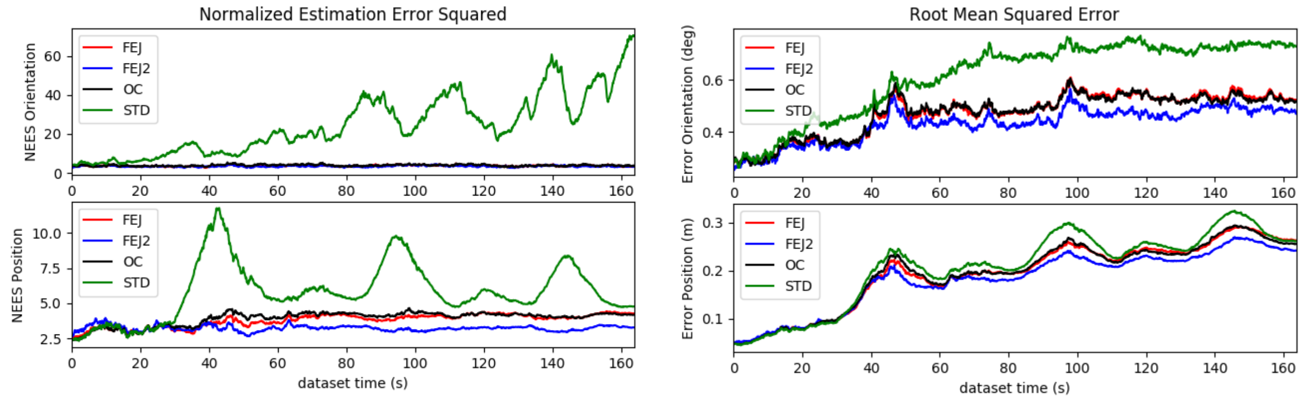

Fig. 2: The average RMSE and NEES over 50 Monte-Carlo simulation for IMU orientation and position with stereo measurements and 2 pixel noises.

TABLE IV: Absolute trajectory error (ATE) for each estimator in units of degree/meters averaged over 10 runs.

	Est.	V1_01_easy	V1_02_med.	V1_03_dif	V2_01_easy	V2_02_med.	V2_03_dif.	MH_01_easy	MH_02_easy	MH_03_med.	MH_04_dif.	MH_05_dif.
mono	STD	0.956 / 0.076	1.783 / 0.080	2.638 / 0.074	0.951 / 0.098	1.856 / 0.085	**1.415** / 0.154	1.958 / 0.246	2.337 / 0.275	1.355 / 0.155	1.640 / 0.306	1.872 / 0.438
	OC	0.554 / 0.077	0.615 / 0.071	2.933 / 0.068	0.880 / 0.093	1.595 / 0.077	1.835 / 0.188	1.346 / 0.190	1.121 / 0.161	1.580 / 0.162	1.114 / 0.277	1.124 / 0.421
	FEJ	0.872 / 0.056	0.574 / **0.052**	**2.079** / 0.096	0.928 / **0.092**	1.599 / 0.074	1.874 / 0.168	1.556 / 0.107	0.920 / 0.156	1.204 / **0.142**	1.109 / **0.261**	1.212 / **0.347**
	FEJ2	**0.679** / **0.053**	**0.564** / 0.059	2.346 / **0.061**	**0.791** / 0.101	**1.233** / **0.047**	1.808 / **0.146**	**1.503** / **0.098**	**0.740** / **0.105**	**1.137** / **0.142**	**0.783** / 0.282	**1.145** / 0.391
stereo	STD	0.792 / 0.061	1.958 / 0.059	2.551 / 0.053	1.078 / 0.055	1.693 / 0.064	2.337 / 0.077	1.656 / 0.183	2.185 / 0.215	1.641 / 0.115	1.331 / 0.223	2.042 / 0.352
	OC	0.615 / 0.071	1.772 / 0.046	**2.468** / **0.045**	1.098 / 0.059	1.231 / 0.051	**1.052** / 0.061	1.606 / 0.123	1.136 / 0.161	1.007 / 0.183	1.151 / 0.289	**0.899** / 0.288
	FEJ	**0.547** / **0.052**	**1.702** / 0.079	2.498 / **0.045**	1.172 / 0.058	1.268 / 0.049	1.118 / **0.058**	**1.102** / 0.117	**0.968** / **0.142**	**1.157** / 0.157	1.090 / **0.214**	1.288 / 0.208
	FEJ2	0.564 / 0.059	1.770 / **0.045**	2.503 / 0.047	**0.975** / **0.053**	**1.202** / **0.047**	1.101 / 0.062	1.193 / **0.070**	1.366 / 0.174	1.824 / **0.138**	**1.038** / 0.273	1.124 / **0.163**

References

[1] G. Huang, "Visual-inertial navigation: A concise review," in *Proc. International Conference on Robotics and Automation*, Montreal, Canada, May 2019.

[2] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual-inertial odometry using nonlinear optimization," *International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2015.

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

[4] V. Usenko, N. Demmel, D. Schubert, J. Stückler, and D. Cremers, "Visual-inertial mapping with non-linear factor recovery," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 422–429, 2019.

[5] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, "Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam," *IEEE Transactions on Robotics*, pp. 1–17, 2021.

[6] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Rome, Italy, Apr. 10–14, 2007, pp. 3565–3572.

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

[8] J. Hesch, D. Kottas, S. Bowman, and S. Roumeliotis, "Camera-IMU-based localization: Observability analysis and consistency improvement," *International Journal of Robotics Research*, vol. 33, pp. 182–201, 2014.

[9] M. Bloesch, M. Burri, S. Omari, M. Hutter, and R. Siegwart, "Iterated extended kalman filter based visual-inertial odometry using direct photometric feedback," *The International Journal of Robotics Research*, vol. 36, no. 10, pp. 1053–1072, 2017.

[10] P. Geneva, K. Eckenhoff, W. Lee, Y. Yang, and G. Huang, "Openvins: A research platform for visual-inertial estimation," in *Proc. of the IEEE International Conference on Robotics and Automation*, Paris, France, 2020. [Online]. Available: https://github.com/rpng/open_vins

[11] G. Huang, "Improving the consistency of nonlinear estimators: Analysis, algorithms, and applications," Ph.D. dissertation, Department of Computer Science and Engineering, University of Minnesota, 2012. [Online]. Available: https://conservancy.umn.edu/handle/11299/146717

[12] A. Martinelli, "Closed-form solution of visual-inertial structure from motion," *International Journal of Computer Vision*, vol. 106, no. 2, pp. 138–152, Jan 2014.

[13] J. Hesch, D. Kottas, S. Bowman, and S. Roumeliotis, "Consistency analysis and improvement of vision-aided inertial navigation," *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 158–176, 2013.

[14] Y. Yang and G. Huang, "Observability analysis of aided ins with heterogeneous features of points, lines and planes," *IEEE Transactions on Robotics*, vol. 35, no. 6, pp. 399–1418, Dec. 2019.

[15] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York: John Willey and Sons, 2001.

[16] G. Huang, A. I. Mourikis, and S. I. Roumeliotis, "Observability-based rules for designing consistent EKF SLAM estimators," *International Journal of Robotics Research*, vol. 29, no. 5, pp. 502–528, Apr. 2010.

[17] Z. Huai and G. Huang, "Robocentric visual-inertial odometry," *International Journal of Robotics Research*, Apr. 2019.

[18] K. Wu, T. Zhang, D. Su, S. Huang, and G. Dissanayake, "An invariant-ekf vins algorithm for improving consistency," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sept 2017, pp. 1578–1585.

[19] T. Zhang, K. Wu, J. Song, S. Huang, and G. Dissanayake, "Convergence and consistency analysis for a 3D invariant-ekf slam," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 733–740, April 2017.

[20] M. Brossard, S. Bonnabel, and A. Barrau, "Invariant Kalman filtering for visual inertial SLAM," in *21st International Conference on Information Fusion*, ser. 21st International Conference on Information Fusion. Cambrige, United Kingdom: University of Cambridge, Jul. 2018.

[21] Y. Yang, C. Chen, W. Lee, and G. Huang, "Decoupled right invariant error states for consistent visual-inertial navigation," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1627–1634, 2022.

[22] G. Huang, A. I. Mourikis, and S. I. Roumeliotis, "A first-estimates Jacobian EKF for improving SLAM consistency," in *Proc. of the 11th International Symposium on Experimental Robotics*, Athens, Greece, Jul. 2008.

[23] M. Brossard, A. Barrau, and S. Bonnabel, "Exploiting symmetries to design ekfs with consistency properties for navigation and slam," *IEEE Sensors Journal*, vol. 19, no. 4, pp. 1572–1579, 2019.

[24] P. Geneva and G. Huang, "Map-based visual-inertial localization: A numerical study," in *Proc. International Conference on Robotics and Automation*, Philadelphia, USA, May 2022.

[25] N. Trawny and S. I. Roumeliotis, "Indirect Kalman filter for 3D attitude estimation," University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep., Mar. 2005.

[26] C. Hertzberg, R. Wagner, U. Frese, and L. SchröDer, "Integrating generic sensor fusion algorithms with sound state representations through encapsulation of manifolds," *Information Fusion*, vol. 14, no. 1, pp. 57–77, Jan. 2013.

[27] C. Chen and G. Huang, "FEJ2-EKF: A consistent estimator for slam (technical report)," University of Delaware, Tech. Rep. RPNG-2022-FEJ2, 2022. [Online]. Available: http://udel.edu/~ghuang/papers/tr_fej2.pdf

[28] P. S. Maybeck, *Stochastic Models, Estimation, and Control*. London: Academic Press, 1979, vol. 1.

[29] A. Martinelli, "State estimation based on the concept of continuous symmetry and observability analysis: The case of calibration," *IEEE Transactions on Robotics*, vol. 27, no. 2, pp. 239–255, 2011.

[30] G. Huang, M. Kaess, and J. Leonard, "Towards consistent visual-inertial navigation," in *Proc. of the IEEE International Conference on Robotics and Automation*, Hong Kong, China, May 2014, pp. 4926–4933.

[31] Y. Yang, J. Maley, and G. Huang, "Null-space-based marginalization: Analysis and algorithm," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vancouver, Canada, Sep. 2017, pp. 6749–6755.

[32] Z. Zhang and D. Scaramuzza, "A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 7244–7251.

[33] M. Li and A. I. Mourikis, "Optimization-based estimator design for vision-aided inertial navigation," in *Robotics: Science and Systems*, Berlin, Germany, June 2013, pp. 241–248.

[34] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The euroc micro aerial vehicle datasets," *The International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016.

[35] D. Schubert, T. Goll, N. Demmel, V. Usenko, J. Stückler, and D. Cremers, "The tum vi benchmark for evaluating visual-inertial odometry," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1680–1687.