IMU-RGBD Camera Navigation using Point and Plane Features

Chao X. Guo and Stergios I. Roumeliotis

Abstract— **In this paper, we present a linear-complexity 3D inertial navigation algorithm using both point and plane features observed from an RGBD camera. In particular, we study the system's observability properties, and prove that: (i) When observing a single plane feature of known direction, the IMU gyroscope bias is observable. (ii) By observing a single point feature, as well as a single plane of known direction but not perpendicular to gravity, all degrees of freedom of the IMU-RGBD navigation system become observable, up to global translations. Next, based on the results of the observability analysis, we design a consistency-improved, observability-constrained (OC) extended Kalman filter (EKF)-based estimator for the IMU-RGBD camera navigation system. Finally, we experimentally validate the superiority of our proposed algorithm compared to alternative methods in urban scenes.**

I. Introduction and Related Work

Over a short period of time, all six degrees of freedom of a robot's position and orientation (pose) can be obtained directly by integrating the rotational velocity and linear acceleration measurements from an Inertial Measurement Unit (IMU). However, due to the biases and noise in the IMU signals, errors in the robot pose estimates accumulate quickly over time rendering them unreliable. To deal with this problem, most inertial navigation systems (INS) rely on GPS for bounding the estimation error. Unfortunately, for robots operating in urban or indoor environments, the GPS signals are usually either unreliable or unavailable. For this reason, alternative sensors, such as regular and RGBD cameras, are often used to aid the inertial sensors and improve the navigation accuracy.

Compared to regular cameras, RGBD cameras provide both color images and the corresponding 3D point cloud, which simplifies the tasks of triangulating point-feature positions and extracting higher level features, such as planes, from the scene. To date, very few works exist that combine inertial and RGBD measurements for navigation. In [1] and [2], rotational velocity measurements from the IMU are used for facilitating matching 3D point clouds from the RGBD images. The IMU measurements, however, are not fused with the RGBD observations in the estimation process. In [3], the authors solve the IMU-RGBD camera navigation and extrinsic calibration problem by fusing inertial and point feature measurements in an EKF. In [4], IMU and RGBD measurements are used to estimate the position and velocity of a micro air vehicle (MAV). However, both [3] and [4] ignore the fact that common urban or indoor environments mainly consist of planes (e.g., walls orthogonal to each other), which can be easily extracted from RGBD images [5], and provide robust orientation information to the navigation system. Moreover, as shown in [3], when using IMU and only point feature measurements, one degree of rotational freedom (yaw) of the IMU-RGBD camera is unobservable. As a result, the uncertainty and error in the yaw estimates will keep increasing, hence, adversely affecting the positioning accuracy. In this paper, we will show that by observing plane features of known directions, the yaw becomes observable and, thus, its uncertainty remains bounded.

To the best of our knowledge, the only work that considers fusing inertial measurements with plane features is that of [6], which seeks to simultaneously estimate the robot pose and build a map with the observed plane features. However, this algorithm achieves acceptable accuracy only in scenarios where a number of plane features can be detected and tracked. In this paper, we present a linear-complexity inertial navigation algorithm that uses both point and plane features. In particular, we study the system's observability properties, and find its observable modes and unobservable directions. In our algorithm, we process point feature measurements using a tightly-coupled visual-inertial odometry, multi-state constraint Kalman filter (MSC-KF) [7], with complexity linear in the number of observed point features. Additionally, we use the directions of the plane features as measurements in the extended Kalman filter update without including the plane feature poses in the state vector, hence ensuring linear complexity in the number of the observed plane features. In particular, the main contributions of this paper are:

- We study the observability of the IMU-RGBD camera navigation system when using both point and plane feature measurements, and prove that with a single plane feature of known direction, the IMU gyroscope bias is observable. If additionally a single point feature is detected, and the plane's normal vector is not aligned with gravity, all degrees of freedom of the IMU-RGBD camera navigation system, except the global position, become observable.
- Based on the observability analysis, we improve the accuracy and consistency of the IMU-RGBD camera navigation system by employing the observability-constrained extended Kalman filter that enforces the observability requirement [8].
- We present a linear-complexity algorithm for fusing inertial measurements with both point and plane features, and experimentally validate its performance.

The authors are with the Department of Computer Science & Engineering, University of Minnesota, Minneapolis, MN 55455, USA {chaguo|stergios}@cs.umn.edu

This work was supported by the University of Minnesota through the Digital Technology Center (DTC), and the National Science Foundation (IIS-0811946).

The rest of the paper is structured as follows. In Section II, we present the inertial navigation system model using both point and plane feature measurements. In Section III, we briefly describe the methodology we employ for studying the observability properties of unobservable nonlinear systems. In Section IV, we apply this method to the specific IMU-RGBD camera navigation system, and find its unobservable directions. In Section V, we present the OC-EKF algorithm we developed for improving the accuracy and consistency of the inertial navigation system based on its observability properties. In Section VI, we experimentally assess the performance of our proposed algorithm, while in Section VII, we provide concluding remarks and possible directions of future research.

II. VINS Estimator Description

In this section, we first describe the system state and covariance propagation equations using inertial measurements, and then present the measurement model for processing plane and point feature observations.

A. System State and Propagation Model

In the IMU-RGBD camera navigation system, the state vector we estimate is:

$$\mathbf{x} = \begin{bmatrix} {}^I\mathbf{q}_G^T & {}^G\mathbf{v}_I^T & {}^G\mathbf{p}_I^T & {}^G\mathbf{p}_f^T & \mathbf{b}_a^T & \mathbf{b}_g^T \end{bmatrix}^T$$

where ${}^I\mathbf{q}_G$ is the unit quaternion representing the orientation of the global frame $\{G\}$ in the IMU's frame of reference $\{I\}$, ${}^G\mathbf{v}_I$ and ${}^G\mathbf{p}_I$ represent the velocity and position of $\{I\}$ in $\{G\}$, ${}^G\mathbf{p}_f$ denotes the position of the point feature in $\{G\}$, $\mathbf{b}_a$ and $\mathbf{b}_g$ represent the gyroscope and accelerometer biases.

The system model describing the time evolution of the states is:

$$\begin{aligned} {}^I\dot{\mathbf{q}}_G(t) &= \frac{1}{2}\mathbf{\Omega}({}^I\boldsymbol{\omega}(t)){}^I\mathbf{q}_G(t) \\ {}^G\dot{\mathbf{v}}_I(t) &= {}^G\mathbf{a}(t) \\ {}^G\dot{\mathbf{p}}_I(t) &= {}^G\mathbf{v}_I(t) \quad {}^G\dot{\mathbf{p}}_f(t) = \mathbf{0}_{3\times 1} \\ \dot{\mathbf{b}}_a(t) &= \mathbf{w}_{wa} \qquad \dot{\mathbf{b}}_g(t) = \mathbf{w}_{wg} \end{aligned} \tag{1}$$

where ${}^I\boldsymbol{\omega}(t) = \begin{bmatrix} \omega_1 & \omega_2 & \omega_3 \end{bmatrix}^T$ and ${}^G\mathbf{a}(t) = \begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix}^T$ are the system rotational velocity and linear acceleration expressed in $\{I\}$ and $\{G\}$ respectively, $\mathbf{w}_{wa}$ and $\mathbf{w}_{wg}$ are zero-mean white Gaussian noise processes driving the gyroscope and accelerometer biases $\mathbf{b}_g$ and $\mathbf{b}_a$, ${}^G\mathbf{g}$ is the gravitational acceleration in $\{G\}$, $\mathbf{C}({}^I\mathbf{q}_G(t))$ denotes the rotation matrix corresponding to ${}^I\mathbf{q}_G(t)$, and

$$\mathbf{\Omega}({}^I\boldsymbol{\omega}(t)) \triangleq \begin{bmatrix} -\lfloor {}^I\boldsymbol{\omega} \rfloor & {}^I\boldsymbol{\omega} \\ -{}^I\boldsymbol{\omega}^T & \mathbf{0} \end{bmatrix}, \quad \lfloor {}^I\boldsymbol{\omega} \rfloor \triangleq \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}$$

The gyroscope and accelerometer measurements, $\boldsymbol{\omega}_m$ and $\mathbf{a}_m$, are modeled as:

$$\boldsymbol{\omega}_m(t) = {}^I\boldsymbol{\omega}(t) + \mathbf{b}_g + \mathbf{w}_g(t) \tag{2}$$

$$\mathbf{a}_m(t) = \mathbf{C}({}^I\mathbf{q}_G(t))({}^G\mathbf{a}(t) - {}^G\mathbf{g}) + \mathbf{b}_a + \mathbf{w}_a(t) \tag{3}$$

where $\mathbf{w}_g$ and $\mathbf{w}_a$ are zero-mean, white Gaussian noise processes. In order to determine the covariance propagation equation, we define the error-state vector as: [1]

$$\widetilde{\mathbf{x}} = \begin{bmatrix} {}^I\delta\boldsymbol{\theta}_G^T & {}^G\widetilde{\mathbf{v}}_I^T & {}^G\widetilde{\mathbf{p}}_I^T & {}^G\widetilde{\mathbf{p}}_f^T & \widetilde{\mathbf{b}}_a^T & \widetilde{\mathbf{b}}_g^T \end{bmatrix}^T, \tag{4}$$

Then, as shown in [9], the linearized continuous-time error-state equation can be written as:

$$\dot{\widetilde{\mathbf{x}}} = \mathbf{F}_c\widetilde{\mathbf{x}} + \mathbf{G}_c\mathbf{w}, \tag{5}$$

where $\mathbf{w} = \begin{bmatrix} \mathbf{w}_g^T & \mathbf{w}_{wg}^T & \mathbf{w}_a^T & \mathbf{w}_{wa}^T \end{bmatrix}^T$ denotes the system noise, $\mathbf{F}_c$ is the continuous-time error-state transition matrix corresponding to the system state, and $\mathbf{G}_c$ is the continuous-time input noise matrix. The system noise is modelled as a zero-mean white Gaussian process with autocorrelation $\mathbb{E}[\mathbf{w}(t)\mathbf{w}^T(\tau)] = \mathbf{Q}_c\delta(t-\tau)$. To compute the propagated covariance, we need to find the discrete-time state transition matrix from time t_k to t_{k+1}, $\mathbf{\Phi}_k$, and the system noise covariance matrix, $\mathbf{Q}_k$, which, as shown in [10], can been computed as:

$$\mathbf{\Phi}_k = \mathbf{\Phi}(t_{k+1}, t_k) = \exp\left(\int_{t_k}^{t_{k+1}} \mathbf{F}_c(\tau)\mathrm{d}\tau\right) \tag{6}$$

$$\mathbf{Q}_k = \int_{t_k}^{t_{k+1}} \mathbf{\Phi}(t_{k+1}, \tau)\mathbf{G}_c\mathbf{Q}_c\mathbf{G}_c^T\mathbf{\Phi}^T(t_{k+1}, \tau)\mathrm{d}\tau \tag{7}$$

Then, the propagated covariance can be determined as

$$\mathbf{P}_{k+1|k} = \mathbf{\Phi}_k\mathbf{P}_{k|k}\mathbf{\Phi}_k^T + \mathbf{Q}_k. \tag{8}$$

B. Measurement Model for Plane Features

For simplicity, we assume the IMU frame $\{I\}$ and the RGBD-camera frame $\{C\}$ coincide[2]. Let ${}^G\mathbf{n}$ denote the normal vector to a plane, whose direction is assumed known in the global frame of reference, and thus we need not include it in the state vector. We fit planes in the 3D point cloud provided by the RGBD camera, and use its normal vector, $\mathbf{z}_{plane}$, as the plane feature measurement:

$$\mathbf{z}_{plane} = \mathbf{C}(\boldsymbol{\eta}_\theta){}^I\mathbf{n} = \mathbf{C}(\boldsymbol{\eta}_\theta)\mathbf{C}({}^I\mathbf{q}_G){}^G\mathbf{n} \tag{9}$$

where $\boldsymbol{\eta}_{\boldsymbol{\theta}} = \alpha\mathbf{k}$ is the measurement noise representing a rotation by an angle α around the unit vector $\mathbf{k}$. Since ${}^G\mathbf{n}$ is a unit norm vector, the measurement noise is modelled as an extra rotation of the plane's normal vector. Moreover, in order to avoid a singular representation of the noise covariance when processing this observation in the EKF, we introduce the following modified measurement model:

$$\mathbf{z}'_{plane} = \begin{bmatrix} \mathbf{z}_{plane}(1)/\mathbf{z}_{plane}(3) \\ \mathbf{z}_{plane}(2)/\mathbf{z}_{plane}(3) \end{bmatrix} \tag{10}$$

and the linearized error model is computed as:

$$\tilde{\mathbf{z}}'_{plane} = \mathbf{z}'_{plane} - \hat{\mathbf{z}}'_{plane} \simeq \mathbf{H}_{plane}\tilde{\mathbf{x}} + \boldsymbol{\eta}_{plane} \tag{11}$$

where $\hat{\mathbf{z}}'_{plane}$ is the expected measurement computed by evaluating (10) at the current state estimate and $\boldsymbol{\eta}_{\boldsymbol{\theta}} = \mathbf{0}$, $\boldsymbol{\eta}_{plane}$

[1] For the IMU-RGBD camera position, velocity, biases, and the point feature position, an additive error model is utilized (i.e., $\widetilde{\mathbf{y}} = \mathbf{y} - \hat{\mathbf{y}}$ is the error in the estimate $\hat{\mathbf{y}}$ of a quantity $\mathbf{y}$). For the quaternion $\mathbf{q}$ we employ a multiplicative error model $\delta\bar{q} = \bar{q} \otimes \hat{\bar{q}}^{-1} \simeq \begin{bmatrix} \frac{1}{2}\delta\boldsymbol{\theta}^T & 1 \end{bmatrix}^T$, where $\delta\boldsymbol{\theta}$ is a minimal representation of the attitude error.

[2] In practice, we perform IMU-RGBD camera extrinsic calibration following the approach of [3].

is the measurement noise, and the measurement Jacobian, $\mathbf{H}_{plane}$, is computed using the chain rule as:

$$\mathbf{H}_{plane} = \mathbf{H}_c \begin{bmatrix} \mathbf{H}_{\boldsymbol{\theta}_1} & \mathbf{0}_{3\times 15} \end{bmatrix} \tag{12}$$

where

$$\begin{aligned} \mathbf{H}_c &= \frac{\partial \mathbf{z}'_{plane}}{\partial {}^{\mathbf{I}}\mathbf{n}} \\ &= \frac{1}{\hat{\mathbf{z}}_{plane}(3)^2} \begin{bmatrix} \hat{\mathbf{z}}_{plane}(3) & \mathbf{0} & -\hat{\mathbf{z}}_{plane}(1) \\ \mathbf{0} & \hat{\mathbf{z}}_{plane}(3) & -\hat{\mathbf{z}}_{plane}(2) \end{bmatrix} \\ \mathbf{H}_{\boldsymbol{\theta}_1} &= \frac{\partial {}^{\mathbf{I}}\mathbf{n}}{\partial {}^{I}\hat{\boldsymbol{\theta}}_G} = \lfloor \mathbf{C}({}^{I}\hat{\mathbf{q}}_G)\, {}^{G}\mathbf{n} \rfloor \end{aligned} \tag{13}$$

C. Measurement Model for Point Features

The RGBD camera can directly measure the 3D position of a point feature ${}^{I}\mathbf{p}_f$ in the IMU frame $\{I\}$ as:

$$\mathbf{z}_{point} = {}^{I}\mathbf{p}_f + \boldsymbol{\eta}_{point} = \mathbf{C}({}^{I}\mathbf{q}_G)({}^{G}\mathbf{p}_f - {}^{G}\mathbf{p}_I) + \boldsymbol{\eta}_{point} \tag{14}$$

The linearized error model is computed as:

$$\tilde{\mathbf{z}}_{point} = \mathbf{z}_{point} - \hat{\mathbf{z}}_{point} \simeq \mathbf{H}_{point}\tilde{\mathbf{x}} + \boldsymbol{\eta}_{point} \tag{15}$$

where $\hat{\mathbf{z}}_{point}$ is the expected measurement computed by evaluating (14) at the current state estimate and $\boldsymbol{\eta}_{point} = \mathbf{0}$, while the measurement Jacobian, $\mathbf{H}_{point}$, is

$$\mathbf{H}_{point} = \begin{bmatrix} \mathbf{H}_{\boldsymbol{\theta}_2} & \mathbf{0}_{3\times 3} & \mathbf{H}_{\mathbf{p}} & \mathbf{H}_{\mathbf{p}_f} & \mathbf{0}_{3\times 6} \end{bmatrix} \tag{16}$$

where

$$\mathbf{H}_{\boldsymbol{\theta}_2} = \frac{\partial \mathbf{z}_{point}}{\partial {}^{I}\boldsymbol{\theta}_G} = \lfloor \mathbf{C}({}^{I}\hat{\mathbf{q}}_G)({}^{G}\hat{\mathbf{p}}_f - {}^{G}\hat{\mathbf{p}}_I) \rfloor$$

$$\mathbf{H}_{\mathbf{p}} = \frac{\partial \mathbf{z}_{point}}{\partial {}^{G}\mathbf{p}} = -\mathbf{C}({}^{I}\hat{\mathbf{q}}_G), \quad \mathbf{H}_{\mathbf{p}_f} = \frac{\partial \mathbf{z}_{point}}{\partial {}^{G}\mathbf{p}_f} = \mathbf{C}({}^{I}\hat{\mathbf{q}}_G)$$

III. Observability Analysis

In this section, we first provide a brief overview of the method proposed by Hermann and Krener [11] for analyzing the observability of nonlinear systems, and then present our extension [3] for determining the unobservable directions of nonlinear systems.

A. Observability Analysis with Lie Derivatives

Consider a nonlinear, continuous-time system:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{f}_0(\mathbf{x}) + \sum_{i=1}^{l} \mathbf{f}_i(\mathbf{x}) u_i \\ \mathbf{y} = \mathbf{h}(\mathbf{x}) \end{cases} \tag{17}$$

where $\mathbf{u} = \begin{bmatrix} u_1 & \dots & u_l \end{bmatrix}^T$ is its control input, $\mathbf{x} = \begin{bmatrix} x_1 & \dots & x_m \end{bmatrix}^T$ is the system's state vector, $\mathbf{y}$ is the system output, and $\mathbf{f}_i,\ i = 0, \dots, l$ are the process functions. The zeroth-order Lie derivative of a measurement function $\mathbf{h}$ is defined as the function itself [11]:

$$\mathfrak{L}^0\mathbf{h} = \mathbf{h}(\mathbf{x}) \tag{18}$$

and the span of the ith order Lie derivative is defined as:

$$\nabla \mathfrak{L}^i\mathbf{h} = \begin{bmatrix} \frac{\partial \mathfrak{L}^i\mathbf{h}}{\partial x_1} & \frac{\partial \mathfrak{L}^i\mathbf{h}}{\partial x_2} & \dots & \frac{\partial \mathfrak{L}^i\mathbf{h}}{\partial x_m} \end{bmatrix} \tag{19}$$

For any ith order Lie derivative, $\mathfrak{L}^i\mathbf{h}$, the $i+1$th order Lie derivative $\mathfrak{L}^{i+1}_{\mathbf{f}_j}\mathbf{h}$ with respect to any process function $\mathbf{f}_j$ can be computed as:

$$\mathfrak{L}^{i+1}_{\mathbf{f}_j}\mathbf{h} = \nabla \mathfrak{L}^i\mathbf{h} \cdot \mathbf{f}_j \tag{20}$$

Finally, the observability matrix $\mathscr{O}$ of system (17) is defined as a matrix with block rows the span of the Lie derivatives of (17), i.e.,

$$\mathscr{O} = \begin{bmatrix} \nabla \mathfrak{L}^0\mathbf{h} \\ \nabla \mathfrak{L}^1_{\mathbf{f}_i}\mathbf{h} \\ \nabla \mathfrak{L}^2_{\mathbf{f}_i\mathbf{f}_j}\mathbf{h} \\ \nabla \mathfrak{L}^3_{\mathbf{f}_i\mathbf{f}_j\mathbf{f}_k}\mathbf{h} \\ \vdots \end{bmatrix} \tag{21}$$

where $i, j, k = 0, \dots, l$. Based on [11], to prove that a system is observable, it suffices to show that any submatrix of $\mathscr{O}$ comprising a subset of its rows is of full column rank. In contrast, to prove that a system is unobservable and find its unobservable directions, we need to: (i) show that the infinitely many block rows of $\mathscr{O}$ can be written as a linear combination of a subset of its block rows, which form a submatrix $\mathscr{O}'$; and (ii) find the nullspace of $\mathscr{O}'$ in order to determine the system's unobservable directions. Although accomplishing (ii) is fairly straightforward, achieving (i) is extremely challenging especially for high-dimensional systems, such as the IMU-RGBD camera navigation system.

B. Observability Analysis with Basis Functions

To address this issue, we leverage the methodology in [12] and [3] in our observability analysis, which relies on change of variables for proving that a system is unobservable and finding its unobservable directions.

Theorem 1: Assume that there exists a nonlinear transformation $\boldsymbol{\beta}(\mathbf{x}) = \begin{bmatrix} \boldsymbol{\beta}_1(\mathbf{x})^T & \dots & \boldsymbol{\beta}_t(\mathbf{x})^T \end{bmatrix}^T$ (i.e., a set of basis functions) of the variable $\mathbf{x}$ in (17), such that:

(A1) $\mathbf{h}(\mathbf{x}) = \mathbf{h}'(\boldsymbol{\beta})$ is a function of $\boldsymbol{\beta}$.

(A2) $\frac{\partial \boldsymbol{\beta}}{\partial \mathbf{x}} \cdot \mathbf{f}_i,\ i = 0, \dots, l$, are functions of $\boldsymbol{\beta}$;

(A3) $\boldsymbol{\beta}$ is a function of the variables of a set $\mathbf{S}$ comprising Lie derivatives of system (17) from order zero up to order p, with $p < \infty$.

Then:

(i) The observability matrix of (17) can be factorized as: $\mathscr{O} = \Xi \cdot \mathbf{B}$, where $\mathbf{B} \triangleq \frac{\partial \boldsymbol{\beta}}{\partial \mathbf{x}}$ and Ξ is the observability matrix of the following system:

$$\begin{cases} \dot{\boldsymbol{\beta}} = \mathbf{g}_0(\boldsymbol{\beta}) + \sum_{i=1}^{l} \mathbf{g}_i(\boldsymbol{\beta}) u_i \\ \mathbf{y} = \mathbf{h}'(\boldsymbol{\beta}) \end{cases} \tag{22}$$

where $\mathbf{g}_i(\boldsymbol{\beta}) \triangleq \frac{\partial \boldsymbol{\beta}}{\partial \mathbf{x}} \mathbf{f}_i(\mathbf{x}),\ i = 0, \dots, l$.

(ii) System (22) is observable.

(iii) $null(\mathscr{O}) = null(\mathbf{B})$.

Proof: The proof is given in [3].

Based on *Theorem 1*, the unobservable directions can be determined with significantly less effort. To find a system's unobservable directions, we first need to define the basis functions that satisfy conditions *(A1)* and *(A3)*, and verify that condition *(A2)* is satisfied, or equivalently that the

basis function set is complete. Once all the conditions are fulfilled, the unobservable directions of (17) correspond to the nullspace of matrix $\mathbf{B}$, which has finite dimensions, and thus it is easy to analyze.

IV. Observability Analysis of the IMU-RGBD Camera Navigation System

In this section, we leverage *Theorem 1* to study the observability of the IMU-RGBD camera navigation system when using plane and point feature observations. To do this, in Section IV-A we find the system's basis functions, which are also the observable modes, using only a single plane feature. Then, in Section IV-B we complete the basis function set for the IMU-RGBD camera navigation system using both plane and point features. Finally, in Section IV-C, we find the unobservable directions of the IMU-RGBD camera navigation system when using only plane observations, and when using both plane and point feature measurements.

A. Basis Functions when using Plane Features

For simplicity, we express the orientation between the IMU frame $\{I\}$ and the global frame $\{G\}$ using the Cayley-Gibbs-Rodriguez parameters [13], ${}^{I}\mathbf{s}_{G}$. Furthermore, we retain only a few of the subscripts and superscripts in the state vector which is expressed as:

$$\mathbf{x} = \begin{bmatrix}\mathbf{s}^T & \mathbf{v}^T & \mathbf{p}^T & \mathbf{p}_f^T & \mathbf{b}_a^T & \mathbf{b}_g^T\end{bmatrix}^T$$

Employing the propagation model in [3], the IMU-RGBD camera navigation system using only plane features can be written as:

$$\begin{bmatrix}\dot{\mathbf{s}}\\ \dot{\mathbf{v}}\\ \dot{\mathbf{p}}\\ \dot{\mathbf{p}}_f\\ \dot{\mathbf{b}}_a\\ \dot{\mathbf{b}}_g\end{bmatrix} = \underbrace{\begin{bmatrix}-\frac{1}{2}\mathbf{D}\mathbf{b}_g\\ \mathbf{g}-\mathbf{C}^T\mathbf{b}_a\\ \mathbf{v}\\ \mathbf{0}\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}}_{f_0} + \underbrace{\begin{bmatrix}\frac{1}{2}\mathbf{D}\\ \mathbf{0}\\ \mathbf{0}\\ \mathbf{0}\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}}_{\mathbf{f}_1}\boldsymbol{\omega} + \underbrace{\begin{bmatrix}\mathbf{0}\\ \mathbf{C}^T\\ \mathbf{0}\\ \mathbf{0}\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}}_{\mathbf{f}_2}\mathbf{a}$$
$$\mathbf{z}_{plane} = \mathbf{C}^G\mathbf{n} \tag{23}$$

where $\mathbf{C} \triangleq \mathbf{C}(\mathbf{s})$ represents the rotation matrix corresponding to $\mathbf{s}$, and $\mathbf{D} \triangleq 2\frac{\partial \mathbf{s}}{\partial \boldsymbol{\theta}} = \mathbf{I} + \lfloor \mathbf{s}\times \rfloor + \mathbf{s}\mathbf{s}^T$. Note that f_0 is a 18×1 vector, while $\mathbf{f}_1$ and $\mathbf{f}_2$ are both 18×3 matrices which is a compact way for representing three process functions:

$$\begin{aligned}\mathbf{f}_1\boldsymbol{\omega} &= f_{11}\omega_1 + f_{12}\omega_2 + f_{13}\omega_3\\ \mathbf{f}_2\mathbf{a} &= f_{21}a_1 + f_{22}a_2 + f_{23}a_3\end{aligned} \tag{24}$$

To define the basis functions for this system, we follow the conditions of *Theorem 1*: (i) Select basis functions so that the measurement function $\mathbf{z}_{plane}$ can be expressed as a function of $\boldsymbol{\beta}$; (ii) Select the remaining basis functions as functions of the system's Lie derivatives, until condition *(A2)*, (i.e., $\frac{\partial \boldsymbol{\beta}}{\partial \mathbf{x}}\cdot\mathbf{f}_i$ is a function of $\boldsymbol{\beta}$ for any i), is satisfied by all the basis functions.

For this particular problem, we define the first set of basis functions directly as the measurement function:

$$\boldsymbol{\beta}_1 \triangleq \mathbf{z}_{plane} = \mathbf{C}^G\mathbf{n} \tag{25}$$

where $\boldsymbol{\beta}_1$ is a 3×1 vector representing in a compact form 3 basis functions. To check if condition *(A2)* of *Theorem 1* is fulfilled, we compute the span of $\boldsymbol{\beta}_1$ with respect to $\mathbf{x}$

$$\begin{aligned}\frac{\partial \boldsymbol{\beta}_1}{\partial \mathbf{x}} &= \begin{bmatrix}\frac{\partial \boldsymbol{\beta}_1}{\partial \boldsymbol{\theta}}\frac{\partial \boldsymbol{\theta}}{\partial \mathbf{s}} & \frac{\partial \boldsymbol{\beta}_1}{\partial \mathbf{v}} & \frac{\partial \boldsymbol{\beta}_1}{\partial \mathbf{p}} & \frac{\partial \boldsymbol{\beta}_1}{\partial \mathbf{p}_f} & \frac{\partial \boldsymbol{\beta}_1}{\partial \mathbf{b}_a} & \frac{\partial \boldsymbol{\beta}_1}{\partial \mathbf{b}_g}\end{bmatrix}\\ &= \begin{bmatrix}\lfloor \mathbf{C}^G\mathbf{n}\rfloor\frac{\partial \boldsymbol{\theta}}{\partial \mathbf{s}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\end{bmatrix}\end{aligned}$$

and project it onto all the process functions:

$$\frac{\partial \boldsymbol{\beta}_1}{\partial \mathbf{x}}\cdot f_0 = -\lfloor \mathbf{C}^G\mathbf{n}\rfloor\mathbf{b}_g \tag{26}$$

$$\frac{\partial \boldsymbol{\beta}_1}{\partial \mathbf{x}}\cdot f_{1i} = \lfloor \mathbf{C}^G\mathbf{n}\rfloor\mathbf{e}_i = -\lfloor \mathbf{e}_i\rfloor\mathbf{C}^G\mathbf{n} \tag{27}$$

$$\frac{\partial \boldsymbol{\beta}_{1j}}{\partial \mathbf{x}}\cdot f_{2i} = \mathbf{0} \tag{28}$$

where $i = 1,2,3$, $\mathbf{e}_1 = \begin{bmatrix}1 & 0 & 0\end{bmatrix}^T$, $\mathbf{e}_2 = \begin{bmatrix}0 & 1 & 0\end{bmatrix}^T$, $\mathbf{e}_3 = \begin{bmatrix}0 & 0 & 1\end{bmatrix}^T$, $\frac{\partial \boldsymbol{\theta}}{\partial \mathbf{s}}\frac{1}{2}\mathbf{D} = \frac{\partial \boldsymbol{\theta}}{\partial \mathbf{s}}\frac{\partial \mathbf{s}}{\partial \boldsymbol{\theta}} = \mathbf{I}_3$.

Obviously, $\frac{\partial \boldsymbol{\beta}_{1j}}{\partial \mathbf{x}}\cdot f_0$ contains $\mathbf{b}_g$, and thus is not a function of the previously defined basis function $\boldsymbol{\beta}_1$. To proceed, we will employ condition *(A3)* of *Theorem 1* to define additional basis functions as nonlinear combinations of the system's Lie derivatives.

Since the basis function $\boldsymbol{\beta}_1$ is the zeroth-order Lie derivative of the measurement $\mathbf{h} = \mathbf{z}_{plane}$, then by definition, (26) is one of the first-order Lie derivatives:

$$\frac{\partial \boldsymbol{\beta}_1}{\partial \mathbf{x}}\cdot f_0 = \mathfrak{L}^1_{\mathbf{f}_0}\mathbf{h}$$

Hereafter, we will make use of this fact to define more basis functions. By definition, the second-order Lie derivative $\mathfrak{L}^2_{\mathbf{f}_0\mathbf{f}_{1i}}\mathbf{h}$ can be computed as

$$\mathfrak{L}^2_{\mathbf{f}_0\mathbf{f}_{1i}}\mathbf{h} = \nabla\mathfrak{L}^1_{\mathbf{f}_0}\mathbf{h}\cdot f_{1i} = -\lfloor\lfloor \mathbf{C}^G\mathbf{n}\rfloor\mathbf{e}_i\rfloor\mathbf{b}_g \tag{29}$$

If we stack equation (29), for $i = 1,2,3$, into a matrix form

$$\begin{bmatrix}\mathfrak{L}^2_{\mathbf{f}_0\mathbf{f}_{11}}\mathbf{h}\\ \mathfrak{L}^2_{\mathbf{f}_0\mathbf{f}_{12}}\mathbf{h}\\ \mathfrak{L}^2_{\mathbf{f}_0\mathbf{f}_{13}}\mathbf{h}\end{bmatrix} = -\underbrace{\begin{bmatrix}\lfloor\lfloor \mathbf{C}^G\mathbf{n}\rfloor\mathbf{e}_1\rfloor\\ \lfloor\lfloor \mathbf{C}^G\mathbf{n}\rfloor\mathbf{e}_2\rfloor\\ \lfloor\lfloor \mathbf{C}^G\mathbf{n}\rfloor\mathbf{e}_3\rfloor\end{bmatrix}}_{\mathbf{Y}}\mathbf{b}_g \tag{30}$$

since $\mathbf{Y}$ is a 9×3 matrix of full column rank, $\mathbf{b}_g$ can be determined in terms of the Lie derivatives $\mathfrak{L}^2_{\mathbf{f}_0\mathbf{f}_{1i}}\mathbf{h}$ and $\boldsymbol{\beta}_1 = \mathbf{C}^G\mathbf{n}$. Therefore, $\mathbf{b}_g$ is a function of the Lie derivatives, and we define it as a new basis function:

$$\boldsymbol{\beta}_2 \triangleq \mathbf{b}_g \tag{31}$$

Then, if we compute the span of $\boldsymbol{\beta}_2$, and project it onto the process functions, we have

$$\frac{\partial \boldsymbol{\beta}_2}{\partial \mathbf{x}}\cdot f_0 = \mathbf{0} \quad \frac{\partial \boldsymbol{\beta}_2}{\partial \mathbf{x}}\cdot f_{1i} = \mathbf{0} \quad \frac{\partial \boldsymbol{\beta}_2}{\partial \mathbf{x}}\cdot f_{2i} = \mathbf{0} \tag{32}$$

which are all zeros, and thus do not contain any term not belonging to the previously defined basis functions. Therefore, we have found a complete basis function set of the IMU-RGBD camera navigation system using a single plane feature.

B. Basis Functions when using both Plane and Point Features

As shown in [3], the basis functions of the IMU-RGBD camera navigation system using only a single point feature are

$$\begin{bmatrix}\boldsymbol{\beta}_3 & \boldsymbol{\beta}_4 & \boldsymbol{\beta}_5 & \boldsymbol{\beta}_6 & \boldsymbol{\beta}_7\end{bmatrix} = \begin{bmatrix}C(\mathbf{p}_f-\mathbf{p}) & \mathbf{b}_g & \mathbf{C}\mathbf{v} & \mathbf{C}\mathbf{g} & \mathbf{b}_a\end{bmatrix} \tag{33}$$

Since the basis functions are also the system's observable modes, the complete basis function set of the IMU-RGBD camera navigation system when using *both* plane and point features is the union of the basis function sets, $\{\boldsymbol{\beta}_1,\boldsymbol{\beta}_2\}$ (resulting from measurements of the plane feature), and $\{\boldsymbol{\beta}_3,\boldsymbol{\beta}_4,\boldsymbol{\beta}_5,\boldsymbol{\beta}_6,\boldsymbol{\beta}_7\}$ (computed for observations of the point feature). Hereafter, we will determine the union of these two basis function sets after removing redundant elements.

First, since $\boldsymbol{\beta}_2=\boldsymbol{\beta}_4=\mathbf{b}_g$, we have $\boldsymbol{\beta}_2\cap\boldsymbol{\beta}_4=\mathbf{b}_g$. Then, under the assumption that the normal vector of the observed plane is not parallel to gravity, $\mathbf{C}=\mathbf{C}(\mathbf{s})$ can be expressed in terms of ${}^G\mathbf{n}$, $\mathbf{g}$, $\boldsymbol{\beta}_1=\mathbf{C}{}^G\mathbf{n}$ and $\boldsymbol{\beta}_6=\mathbf{C}\mathbf{g}$ using, e.g., the method introduced in [14]. Since both ${}^G\mathbf{n}$ and $\mathbf{g}$ are known quantities, we have $\boldsymbol{\beta}_1\cap\boldsymbol{\beta}_6=\mathbf{s}$. Therefore, the basis functions of the IMU-RGBD camera navigation system using both plane and point features are:

$$\boldsymbol{\beta}'=\begin{bmatrix}\boldsymbol{\beta}'_1\\ \boldsymbol{\beta}'_2\\ \boldsymbol{\beta}'_3\\ \boldsymbol{\beta}'_4\\ \boldsymbol{\beta}'_5\end{bmatrix}=\begin{bmatrix}C(\mathbf{p}_f-\mathbf{p})\\ \mathbf{s}\\ \mathbf{C}\mathbf{v}\\ \mathbf{b}_g\\ \mathbf{b}_a\end{bmatrix} \tag{34}$$

with which, we leverage result *(i)* of *Theorem 1* to construct the observable system in terms of the basis functions as:

$$\begin{aligned}\dot{\boldsymbol{\beta}}'&=\frac{\partial\boldsymbol{\beta}'}{\partial\mathbf{x}}\dot{\mathbf{x}}=\frac{\partial\boldsymbol{\beta}'}{\partial\mathbf{x}}\left(\mathbf{f}_0(\mathbf{x})+\sum_{i=1}^{l}\mathbf{f}_i(\mathbf{x})u_i\right)\\ &=\begin{bmatrix}-\lfloor\mathbf{C}(\mathbf{p}_f-\mathbf{p})\rfloor\mathbf{b}_g-\mathbf{C}\mathbf{v}\\ -\frac{1}{2}\mathbf{D}\mathbf{b}_g\\ -\lfloor\mathbf{C}\mathbf{v}\rfloor\mathbf{b}_g+\mathbf{C}\mathbf{g}-\mathbf{b}_a\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}+\begin{bmatrix}\lfloor\mathbf{C}(\mathbf{p}_f-\mathbf{p})\rfloor\\ \frac{1}{2}\mathbf{D}\\ \lfloor\mathbf{C}\mathbf{v}\rfloor\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}\boldsymbol{\omega}+\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \mathbf{I}\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}\mathbf{a}\\ &=\begin{bmatrix}-\lfloor\boldsymbol{\beta}'_1\rfloor\boldsymbol{\beta}'_4-\boldsymbol{\beta}'_3\\ -\frac{1}{2}\mathbf{D}(\boldsymbol{\beta}'_2)\boldsymbol{\beta}'_4\\ -\lfloor\boldsymbol{\beta}'_3\rfloor\boldsymbol{\beta}'_4+\mathbf{C}(\boldsymbol{\beta}'_2)\mathbf{g}-\boldsymbol{\beta}'_5\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}+\begin{bmatrix}\lfloor\boldsymbol{\beta}'_1\rfloor\\ \frac{1}{2}\mathbf{D}(\boldsymbol{\beta}'_2)\\ \lfloor\boldsymbol{\beta}'_3\rfloor\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}\boldsymbol{\omega}+\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \mathbf{I}\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}\mathbf{a}\end{aligned} \tag{35}$$

where $\mathbf{D}(\boldsymbol{\beta}'_2)\triangleq\mathbf{I}+\lfloor\boldsymbol{\beta}'_2\rfloor+\boldsymbol{\beta}'_2\boldsymbol{\beta}'^T_2$. System (35) is actually a minimal representation of the IMU-RGBD camera navigation system using both plane and point features. Hereafter, we will show how to find the unobservable directions of the IMU-RGBD camera navigation system leveraging result *(iii)* of *Theorem 1*.

C. Determining the System's Unobservable Directions

In this section, we first determine the unobservable directions of the IMU-RGBD camera navigation system when observing only a single plane feature by computing the nullspace of the basis functions' span, $\mathbf{B}_1\triangleq\begin{bmatrix}\frac{\partial\boldsymbol{\beta}_1}{\partial\mathbf{x}}^T & \frac{\partial\boldsymbol{\beta}_2}{\partial\mathbf{x}}^T\end{bmatrix}^T$. Then, we will find the unobservable directions of the IMU-RGBD camera navigation system when observing both a single plane feature and a single point feature by computing the nullspace of $\mathbf{B}_2\triangleq\frac{\partial\boldsymbol{\beta}'}{\partial\mathbf{x}}$.

Theorem 2: The IMU-RGBD camera navigation system observing a single plane feature is unobservable, and its unobservable directions are spanned by the IMU-RGBD camera orientation around the plane's normal vector and the accelerometer bias in the IMU frame $\{I\}$, as well as the IMU-RGBD camera position, velocity, and the point feature position in the global frame $\{G\}$.

Proof: In the previous section, we have shown that the basis function set $\{\boldsymbol{\beta}_1,\boldsymbol{\beta}_2\}$ satisfies all three conditions of *Theorem 1*. Therefore, the system's unobservable directions span the nullspace of matrix $\mathbf{B}_1$, which is formed by stacking the spans of the basis functions $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$ as:

$$\begin{bmatrix}\lfloor\mathbf{C}{}^G\mathbf{n}\rfloor\frac{\partial\boldsymbol{\theta}}{\partial\mathbf{s}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}\end{bmatrix} \tag{36}$$

It is easy to see that the nullspace of $\mathbf{B}_1$ is spanned by[3]

$$\mathbf{N}_{plane}\triangleq\begin{bmatrix}\frac{\partial\mathbf{s}}{\partial\boldsymbol{\theta}}\mathbf{C}{}^G\mathbf{n} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\end{bmatrix}=\begin{bmatrix}\mathbf{N}^g_{plane} & \mathbf{N}^p_{plane}\end{bmatrix} \tag{37}$$

where $\mathbf{N}^g_{plane}$ (the first column of $\mathbf{N}_{plane}$) corresponds to the IMU-RGBD camera's rotation around the plane feature's normal vector, and $\mathbf{N}^p_{plane}$ (the remaining 12 columns of $\mathbf{N}_{plane}$) denotes the unobservable directions in the IMU-RGBD camera velocity, position, the point feature position, and the accelerometer bias. ■

In contrast, when both point and plane feature measurements are available, we have

Theorem 3: The IMU-RGBD camera navigation system using a single point feature and a single plane feature (of known direction which is not parallel to gravity) is unobservable, and its unobservable subspace is spanned by 3 directions corresponding to the IMU-RGBD camera position in the global frame $\{G\}$.

Proof: Employing result *(iii)* of *Theorem 1*, the system's unobservable directions can be determined by computing the nullspace of the span $\mathbf{B}_2$ of the corresponding basis functions $\boldsymbol{\beta}'$, where

$$\mathbf{B}_2=\begin{bmatrix}\lfloor\mathbf{C}(\mathbf{p}_f-\mathbf{p})\rfloor\frac{\partial\boldsymbol{\theta}}{\partial\mathbf{s}} & \mathbf{0} & -\mathbf{C} & \mathbf{C} & \mathbf{0} & \mathbf{0}\\ \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\ \lfloor\mathbf{C}\mathbf{v}\rfloor\frac{\partial\boldsymbol{\theta}}{\partial\mathbf{s}} & \mathbf{C} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0}\end{bmatrix} \tag{38}$$

Let $\mathbf{N}=\begin{bmatrix}\mathbf{N}_1^T & \mathbf{N}_2^T & \mathbf{N}_3^T & \mathbf{N}_4^T & \mathbf{N}_5^T & \mathbf{N}_6^T\end{bmatrix}^T$ be the right nullspace of matrix $\mathbf{B}_2$. Hereafter, we employ the relation

[3] $\frac{\partial\mathbf{s}}{\partial\boldsymbol{\theta}}\mathbf{C}{}^G\mathbf{n}$ is the perturbing term with respect to $\mathbf{s}$, while the corresponding term for $\boldsymbol{\theta}$ is $\mathbf{C}{}^G\mathbf{n}$.

$\mathbf{B}_2\mathbf{N} = \mathbf{0}$ to determine the elements of $\mathbf{N}$. Specifically, from the second, fourth, and fifth block rows of the product $\mathbf{B}_2\mathbf{N}$, we have:

$$\mathbf{N}_1 = \mathbf{N}_5 = \mathbf{N}_6 = \mathbf{0}_3 \tag{39}$$

Then, from the first and third block rows of $\mathbf{B}_2\mathbf{N}$, we have $\mathbf{N}_3 = \mathbf{N}_4 = \mathbf{I}_3$, and $\mathbf{N}_2 = \mathbf{0}_3$. Using Gaussian elimination, it is easy to show that the rank of matrix $\mathbf{B}_2$ is 15. Thus, the dimension of its right nullspace is exactly three, and the system's unobservable directions are spanned by:

$$\mathbf{N} \triangleq \begin{bmatrix} \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix}^T \tag{40}$$

which corresponds to the global position of the IMU-RGBD camera and the point feature. Intuitively, this means that translating the IMU-RGBD camera and the point feature positions concurrently has no impact on the system's measurements. ■

In [3], it was shown that the unobservable directions of the IMU-RGBD camera navigation system using only a single point feature are spanned by:

$$\mathbf{N}_{point} \triangleq \begin{bmatrix} \frac{\partial \mathbf{s}}{\partial \boldsymbol{\theta}}\mathbf{C}\mathbf{g} & \mathbf{0}_3 \\ -\lfloor \mathbf{v} \rfloor \mathbf{g} & \mathbf{0}_3 \\ -\lfloor \mathbf{p} \rfloor \mathbf{g} & \mathbf{I}_3 \\ -\lfloor \mathbf{p}_f \rfloor \mathbf{g} & \mathbf{I}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{N}^g_{point} & \mathbf{N}^p_{point} \end{bmatrix} \tag{41}$$

Note that $\mathbf{N} = \mathbf{N}_{plane} \cap \mathbf{N}_{point}$, which makes sense because any unobservable quantity of the IMU-RGBD camera navigation system using both point and plane feature observations, must be unobservable when the system uses either plane or point feature measurements.

V. Algorithm Description

In this section, we present our IMU-RGBD camera navigation algorithm employing the observability constrained (OC)-EKF, which seeks to maintain the original system's observability properties in the linearized implementation (EKF). In particular, we first describe the implementation of the OC-EKF for processing point feature measurements. Then, we prove that once the OC-EKF is employed for point feature measurements, the observability constraint is automatically satisfied for the plane feature measurements.

A system's observability Gramian [10], $\mathbf{M}$, is defined as

$$\mathbf{M} = \begin{bmatrix} \mathbf{H}^1 \\ \mathbf{H}^2\boldsymbol{\Phi}^{2,1} \\ \vdots \\ \mathbf{H}^k\boldsymbol{\Phi}^{k,1} \end{bmatrix} \tag{42}$$

where $\boldsymbol{\Phi}^{k,1} \triangleq \boldsymbol{\Phi}^{k-1}\cdots\boldsymbol{\Phi}^1$ is the state transition matrix from time step 1 to k, and $\mathbf{H}^k$ is the measurement Jacobian at time step k. As described in [10], a system's unobservable directions, $\mathbf{N}$, are supposed to span the observability Gramian's nullspace

$$\mathbf{M}\mathbf{N} = \mathbf{0} \tag{43}$$

However, in [8] and [15], the authors show that (43) does not hold when a nonlinear system is linearized using the current state estimate. As a consequence, the EKF gains spurious information along unobservable directions, which results in smaller uncertainty (that causes the filter to be inconsistent) and larger estimation errors. To address this issue, the OC-EKF modifies the state transition and measurement Jacobian matrices in such a way so that the resulting linearized system adheres to the observability properties of the original nonlinear system. In particular, in [8] it was shown that (43) can be satisfied by enforcing the following two constraints:

$$\mathbf{N}^{k+1} = \boldsymbol{\Phi}^k\mathbf{N}^k \tag{44}$$

$$\mathbf{H}^k\mathbf{N}^k = \mathbf{0}, \quad \forall k > 0 \tag{45}$$

where $\mathbf{N}^k$ and $\mathbf{N}^{k+1}$ are the unobservable directions evaluated at time-steps k and $k+1$. Hereafter, we briefly describe the implementation of our algorithm, while the interested reader is referred to [16] for more details.

A. Observability Constraint for Point Feature Measurements

In this section, we present the implementation of the OC-EKF for the IMU-RGBD camera navigation system using point feature measurements.

(1) Modification of the State Transition Matrix $\boldsymbol{\Phi}^k$: We start by modifying the state transition matrix, $\boldsymbol{\Phi}^k$, according to the observability constraint (44)

$$\mathbf{N}^{k+1}_{point} = \boldsymbol{\Phi}^k\mathbf{N}^k_{point} \tag{46}$$

where $\mathbf{N}^k_{point}$ and $\mathbf{N}^{k+1}_{point}$, defined in (41), are the unobservable directions when using only point features, at time-steps k and $k+1$ respectively, and $\boldsymbol{\Phi}^k$ has the following structure:

$$\begin{bmatrix} \boldsymbol{\Phi}_{11} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \boldsymbol{\Phi}_{16} \\ \boldsymbol{\Phi}_{21} & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \boldsymbol{\Phi}_{25} & \boldsymbol{\Phi}_{26} \\ \boldsymbol{\Phi}_{31} & \delta t\mathbf{I}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \boldsymbol{\Phi}_{35} & \boldsymbol{\Phi}_{36} \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \tag{47}$$

In [3], it was shown that the observability constraint (46) is equivalent to the following three constraints:

$$\boldsymbol{\Phi}_{11}\mathbf{C}^k\mathbf{g} = \mathbf{C}^{k+1}\mathbf{g} \tag{48}$$

which is satisfied by modifying $\boldsymbol{\Phi}^*_{11} = \mathbf{C}^{k+1}\mathbf{C}^{k^T}$, and

$$\boldsymbol{\Phi}_{21}\mathbf{C}^k\mathbf{g} = \lfloor \mathbf{v}^k \rfloor \mathbf{g} - \lfloor \mathbf{v}^{k+1} \rfloor \mathbf{g} \tag{49}$$

$$\boldsymbol{\Phi}_{31}\mathbf{C}^k\mathbf{g} = \delta t \lfloor \mathbf{v}^k \rfloor \mathbf{g} + \lfloor \mathbf{p}^k \rfloor \mathbf{g} - \lfloor \mathbf{p}^{k+1} \rfloor \mathbf{g} \tag{50}$$

which can be formulated and solved analytically as a constrained optimization problem where we seek to find the closest, in the Frobenius norm, $\boldsymbol{\Phi}^*_{21}$ and $\boldsymbol{\Phi}^*_{31}$ that satisfy constraints (49) and (50).

(2) Modification of the Measurement Jacobian $\mathbf{H}^k_{point}$: During the update, we seek to modify the Jacobian matrix $\mathbf{H}^k_{point}$ so as to fulfill constraint (45), i.e.,

$$\mathbf{H}^k_{point}\mathbf{N}^k_{point} = 0 \tag{51}$$

Substituting $\mathbf{H}^k_{point}$ and $\mathbf{N}^k_{point}$, as defined in (16) and (41) respectively, into (51), it can be shown that (51) is equivalent to the following two constraints

$$\begin{bmatrix}\mathbf{H}^k_{\boldsymbol{\theta}_2} & \mathbf{H}^k_{\mathbf{p}}\end{bmatrix}\begin{bmatrix}\mathbf{C}^k\mathbf{g} \\ \left(\lfloor\mathbf{p}^k_f\rfloor - \lfloor\mathbf{p}^k\rfloor\right)\mathbf{g}\end{bmatrix} = \mathbf{0} \tag{52}$$

$$\mathbf{H}^k_{\mathbf{p}_f} = \mathbf{H}^k_{\mathbf{p}} \tag{53}$$

As before, we can analytically determine ${\mathbf{H}^k_{\boldsymbol{\theta}_2}}^*$ and $\mathbf{H}^{k\,*}_{\mathbf{p}}$ that are closest to $\mathbf{H}^k_{\boldsymbol{\theta}_2}$ and $\mathbf{H}^k_{\mathbf{p}}$ in the Frobenius norm, which also satisfy the constraint (52), and select ${\mathbf{H}^k_{\mathbf{p}_f}}^* = \mathbf{H}^{k\,*}_{\mathbf{p}}$ (see [3] for details).

B. Observability Constraint for Plane Feature Measurements

In this section, we will prove that once the OC-EKF is applied to the IMU-RGBD camera navigation system using point feature measurements, the observability constraint (43) is automatically satisfied for the plane feature measurements.

Substituting $\boldsymbol{\Phi}^k$ and $\mathbf{H}^k_{plane}$ into the observability Gramian $\mathbf{M}_{plane}$ [see (42)] for plane feature measurements, the first block row of $\mathbf{M}_{plane}$ is just the measurement Jacobian matrix

$$\mathbf{M}_{plane}(1) = \mathbf{H}^1_{plane} = \mathbf{H}^1_c\begin{bmatrix}\lfloor\mathbf{C}({}^I\mathbf{q}^1_G)^G\mathbf{n}\rfloor & \mathbf{0}_{3\times 15}\end{bmatrix} \tag{54}$$

while the kth block row is computed as

$$\begin{aligned}\mathbf{M}_{plane}(k) &= \mathbf{H}^k_{plane}\boldsymbol{\Phi}^{k,1} = \mathbf{H}^k_{plane}\boldsymbol{\Phi}^{k-1}\cdots\boldsymbol{\Phi}^1 \\ &= \mathbf{H}^k_c\begin{bmatrix}\lfloor\mathbf{C}({}^I\mathbf{q}^k_G)^G\mathbf{n}\rfloor\boldsymbol{\Pi}^k & \mathbf{0}_{3\times 12} & \boldsymbol{\Psi}^k\end{bmatrix}\end{aligned}$$

where $\boldsymbol{\Pi}^k = \boldsymbol{\Phi}^{k-1}_{11}\cdots\boldsymbol{\Phi}^1_{11}$, and $\boldsymbol{\Psi}^k$ is a time-varying matrix that does not affect the current analysis. When applying the OC-EKF to point features, we have modified $\boldsymbol{\Phi}^k_{11} = \mathbf{C}^{k+1}\mathbf{C}^{k\,T}$. Therefore, multiplying $\mathbf{N}^g_{plane}$ and $\mathbf{N}^p_{plane}$ to the right hand side of $\mathbf{M}_{plane}$, we have:

$$\begin{aligned}\mathbf{M}_{plane}(1)\mathbf{N}^g_{plane} &= \mathbf{H}^1_c\lfloor\mathbf{C}({}^I\mathbf{q}^1_G)^G\mathbf{n}\rfloor\mathbf{C}({}^I\mathbf{q}^1_G)^G\mathbf{n} = \mathbf{0} \\ \mathbf{M}_{plane}(k)\mathbf{N}^g_{plane} &= \mathbf{H}^k_c\lfloor\mathbf{C}({}^I\mathbf{q}^k_G)^G\mathbf{n}\rfloor\boldsymbol{\Phi}^{k-1}_{11}\cdots\boldsymbol{\Phi}^1_{11}\mathbf{C}({}^I\mathbf{q}^1_G)^G\mathbf{n} \\ &= \mathbf{H}^k_c\lfloor\mathbf{C}({}^I\mathbf{q}^k_G)^G\mathbf{n}\rfloor\mathbf{C}^k\mathbf{C}^{k-1\,T}\cdots\mathbf{C}^2\mathbf{C}^{1\,T}\mathbf{C}({}^I\mathbf{q}^1_G)^G\mathbf{n} \\ &= \mathbf{H}^k_c\lfloor\mathbf{C}({}^I\mathbf{q}^k_G)^G\mathbf{n}\rfloor\mathbf{C}({}^I\mathbf{q}^k_G)^G\mathbf{n} = \mathbf{0} \\ \mathbf{M}_{plane}(1)\mathbf{N}^p_{plane} &= \mathbf{M}_{plane}(k)\mathbf{N}^p_{plane} = \mathbf{0}\end{aligned}$$

Thus, $\mathbf{M}_{plane}\mathbf{N}_{plane} = \mathbf{0}$ is automatically satisfied. In summary, after enforcing the observability constraint for point feature measurements on the state transition matrix, $\boldsymbol{\Phi}^k$, the observability constraint (43) is automatically satisfied for the plane feature measurements.

VI. Experimental Results

Our experimental setup comprised an InterSense NavChip IMU and a Kinect, which contained an RGB camera and an infrared (IR) depth-finding camera. The intrinsic parameters of the Kinect RGB camera and IR camera, as well as the transformation between them, were determined offline using the algorithm described in [17]. The IMU signals were sampled at 100 Hz, and the Kinect provided RGBD images at a frequency of 6.3 Hz. The plane features are extracted from the RGBD images using the method proposed in [5].

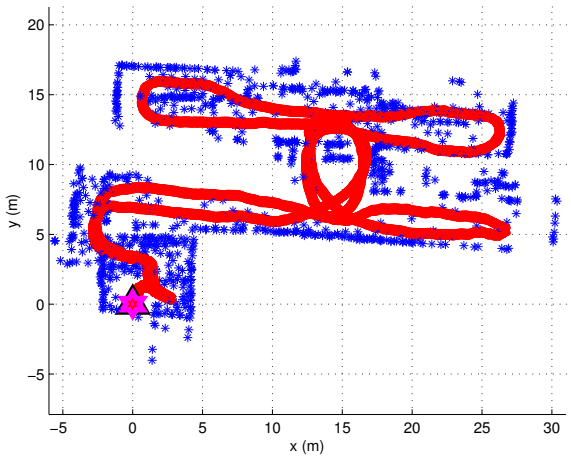

Fig. 1. The overhead x-y view of the IMU-Kinect 3D trajectory and the point features estimated by the *OC-MSC-KF SLAM*.

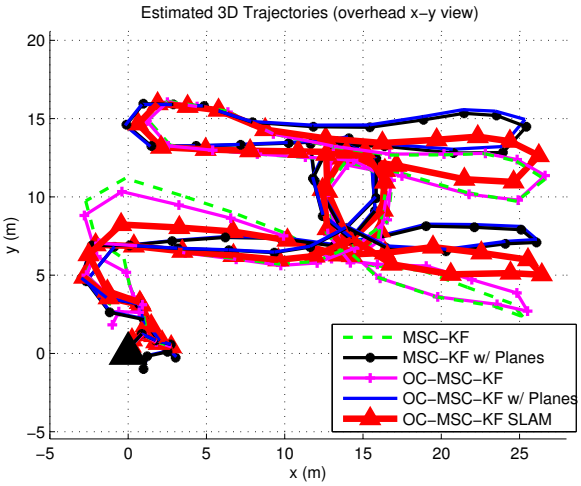

Fig. 2. IMU-Kinect trajectory estimated by the compared algorithms. The black triangle denotes the initial position of the IMU-Kinect pair.

In our experiment, which took place in an office environment, a person holding the IMU-Kinect pair traversed for about 185 meters in two floors of a building, and returned to the initial position. Using the data collected, we examine the final position error of the following five algorithms:

- *MSC-KF*: The Multi-state constraint Kailman filter of [7] using only point feature measurements.
- *MSC-KF w/ Planes*: The MSC-KF that processes both point and plane feature measurements.
- *OC-MSC-KF*: The observability-constrained *MSC-KF* using only point feature measurements [15].
- *OC-MSC-KF w/ Planes*: This is our proposed algorithm, in which we process both point and plane feature measurements with the *OC-MSC-KF*.
- *OC-MSC-KF SLAM*: In this algorithm, we process most of the point features as in the *OC-MSC-KF*, while also simultaneously building a map with a small portion of

TABLE I
CLOSE-LOOP ERRORS

Estimation Algorithm	Final Error (m)	Pct. (%)
MSC-KF	3.22	1.74
MSC-KF w/ Planes	1.57	0.85
OC-MSC-KF	2.37	1.28
OC-MSC-KF w/ Planes	1.46	0.79
OC-MSC-KF SLAM	0.093	0.05

the observed point features using SLAM. This algorithm is supposed to be much more accurate compared to the previous algorithms (as shown in our experiment), because the filter can close loops when observing the same point features again. The main issue of concern for SLAM is that its computational cost is much higher (quadratic in terms of the number of estimated point features), compared to the *MSC-KF* (linear in the number of estimated point features). In our experiment, we use it as a benchmark to examine the performance of the other algorithms.

The 3D trajectory of the IMU-Kinect pair and the point features, estimated by *OC-MSC-KF SLAM*, are plotted in Fig. 1. The 3D trajectories estimated by the algorithms considered are shown in Fig. 2, and their final errors are reported in Table I. As expected, *OC-MSC-KF SLAM* has the lowest final error, and our proposed algorithm, *OC-MSC-KF w/ Planes*, outperforms the other four algorithms. Additionally, the algorithms using both point and plane feature measurements (*MSC-KF w/ Planes* and *OC-MSC-KF w/ Planes*), have much smaller final error and perform closer to the *OC-MSC-KF SLAM*. This is because the plane features provide periodic corrections to the IMU-RGBD camera pair's orientation, thus also improving its position estimation accuracy. Finally, we note that enforcing the observability constraints (*OC-MSC-KF* and *OC-MSC-KF w/ Planes*) results in better accuracy since the filters do not process spurious information.

VII. CONCLUSION AND FUTURE WORK

In this paper, we present an algorithm for fusing inertial measurements, as well as point and plane feature observations in an IMU-RGBD camera navigation system. Specifically, we first prove that by observing only a single plane feature of know direction, only the plane's direction in the IMU frame and the gyroscope bias are observable. Then, we show that by observing a single point feature *and* a single plane feature, of known direction other than the gravity, all the estimated quantities in the IMU-RGBD camera navigation system become observable, except the IMU-RGBD camera position in the global frame. Based on the observability analysis, we design an OC-EKF that significantly improves the estimation accuracy and consistency by removing spurious information along unobservable directions from the estimator. Finally, we experimentally validate our algorithm, and show that it outperforms alternative methods.

As part of our future work, we are currently investigating OC-EKF-based algorithms for processing observations of unknown plane features.

REFERENCES

[1] L. Heng, G. H. Lee, F. Fraundorfer, and M. Pollefeys, "Real-time photo-realistic 3d mapping for micro aerial vehicles," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Francisco, CA, Sept. 25–30 2011, pp. 4012–4019.

[2] B. D. Bouvrie, "Improving RGBD indoor mapping with IMU data," Master's thesis, Delft University of Technology, 2011.

[3] C. X. Guo and S. I. Roumeliotis, "IMU-RGBD camera 3d pose estimation and extrinsic calibration: Observability analysis and consistency improvement," in *Proc. of the IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, May 6–10 2013, pp. 2920–2927.

[4] A. S. Huang, A. Bachrach, P. Henry, M. Krainin, D. Maturana, D. Fox, and N. Roy, "Visual odometry and mapping for autonomous flight using an RGB-D camera," in *Proc. of the International Symposium on Robotics Research*, Flagstaff, AZ, Aug. 28 – Sept. 1 2011.

[5] C. Erdogan, M. Paluri, and F. Dellaert, "Planar segmentation of RGBD images using fast linear fitting and markov chain monte carlo," in *Proc. of the IEEE International Conference on Computer and Robot Vision*, Toronto, Canada, May 27–30 2012, pp. 32–39.

[6] F. Servant, P. Houlier, and E. Marchand, "Improving monocular plane-based SLAM with inertial measurements," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, Oct. 18-22 2010, pp. 3810–3815.

[7] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint kalman filter for vision-aided inertial navigation," in *Proc. of the IEEE International Conference on Robotics and Automation*, Rome, Italy, Apr. 10–14 2007, pp. 3482–3489.

[8] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Observability-constrained vision-aided inertial navigation," University of Minnesota, Dept. of Comp. Sci. & Eng., MARS Lab, Tech. Rep., Feb. 2012. [Online]. Available: http://www-users.cs.umn.edu/~joel/index.php?page=publications

[9] A. I. Mourikis, N. Trawny, S. I. Roumeliotis, A. E. Johson, A. Ansar, and L. Matthies, "Vision-aided inertial navigation for spacecraft entry, descent, and landing," *IEEE Trans. on Robotics*, vol. 25, no. 2, pp. 264–280, Apr. 2009.

[10] P. S. Maybeck, *Stochastic models, estimation and control*. New York, NY: Academic Press, 1979, vol. 1.

[11] R. Hermann and A. J. Krener, "Nonlinear controllability and observability," *IEEE Trans. on Automatic Control*, vol. 22, no. 5, pp. 728–740, Oct. 1977.

[12] J. A. Hesch, "Toward consistent vision-aided inertial navigation," Ph.D. dissertation, University of Minnesota, 2013.

[13] M. D. Shuster, "A survey of attitude representations," *the Journal of Astronautical Sciences*, vol. 41, no. 4, pp. 439–517, Oct.-Dec. 1993.

[14] X. S. Zhou and S. I. Roumeliotis, "Determining 3d relative transformations for any combination of range and bearing measurements," *IEEE Trans. on Robotics*, vol. 29, no. 2, pp. 458 – 474, Apr. 2013.

[15] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Towards consistent vision-aided inertial navigation," in *Proc. of the 10th International Workshop on the Algorithmic Foundations of Robotics*, Cambridge, Massachusetts, June 13–15 2012.

[16] C. X. Guo and S. I. Roumeliotis, "Observability-constrained EKF implementation of the IMU-RGBD camera navigation using point and plane features," University of Minnesota, Tech. Rep., Mar. 2013. [Online]. Available: http://www-users.cs.umn.edu/~chaguo/

[17] D. Herrera, J. Kannala, and J. Heikkila, "Joint depth and color camera calibration with distortion correction," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 34, no. 10, pp. 2058–2064, Oct. 2012.