Automatic Kinematic Calibration of Multi-Axis Gimbal Systems

Fu Zhang[1] and Minghui Zheng[2]

***Abstract*—This paper presents an automatic and efficient calibration method for multi-axis gimbal systems. The calibrated kinematic parameters consist of the axis orientation, the joint encoder nonlinearity map, and the endpoint gyro drift. The calibration of these parameters is conducted simultaneously by sequentially and independently moving one gimbal joint back and forth. With such straightforward intuition, the algorithm is formulated based on the product-of-exponential formula with an analytically derived error model. The key and unique contribution of this algorithm is that it produces a closed-form optimal solution which is computation efficient and easy for embedded implementation. Both numerical and experimental tests are performed to validate the effectiveness of the proposed algorithm in calibrating the kinematic parameters of gimbal systems and compensating the gyro bias.**

I. Introduction

Inertially stabilized multi-axis gimbal systems have been widely used on a variety of mobile platforms, such as unmanned aerial vehicles (UAVs), for payload (e.g., cameras) stabilization. As the host platform provides the necessary mobility in 3D space, it is required to stabilize the attitude of the payload, motivating the use of a 3-axis gimbal system as shown in Fig. 1. A gimbal system usually consists of three joints that are driven by direct-drive motors. The payload attitude relative to the inertial reference frame is measured by two MEMS inertial measurement units (IMUs), one on the payload (i.e., the endpoint IMU) and the other on the host vehicle (i.e., the vehicle IMU). In addition to these, other sensors such as magnetometer and GPS may also be available on the host vehicle. In this paper, we consider a type of compact and cost-effective gimbal systems, in which the endpoint IMU contains a rate gyro and an accelermeter only.

It is fundamentally important to accurately estimate the attitude of the gimbal system for its precise control. A commonly used attitude estimation approach integrates gyro rates and subsequently corrects the accumulated drift using an extended Kalman filter [3] or a complementary filter [4] based on the accelerometer measurements. However, a major problem with this method is that, due to accelerometer drift and the absence of yaw measurements, the attitude estimation is biased and incomplete. Besides the bias, the accelerometer is also susceptible to acceleration disturbances caused by propellers vibrations and airflow turbulence [5].

[1]Fu Zhang is with the Department of Mechanical Engineering, University of Hong Kong, Hong Kong, China. E-mails: fuzhang@hku.hk

[2]Minghui Zheng is with the Department of Mechanical and Aerospace Engineering, University at Buffalo, Buffalo, NY, 14260, USA. E-mail: mhzheng@buffalo.edu.

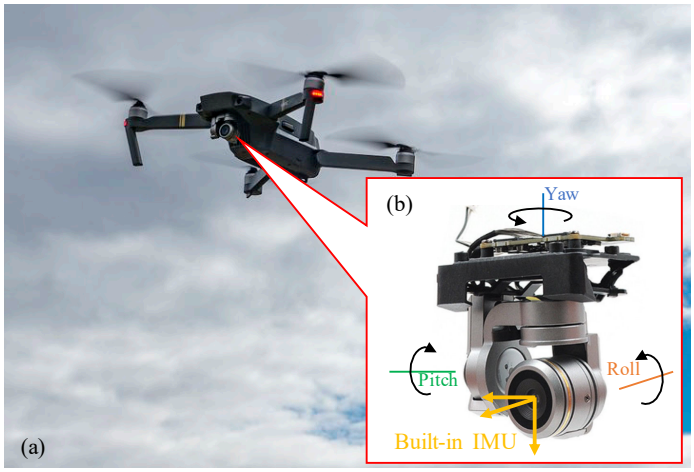


Fig. 1. An unmanned aerial vehicle (a) uses a three-axis miniaturized gimbal system (b) to stabilize its onboard camera. (Picture courtesy of [1] and [2]).

An alternative way to correct the gyro accumulated drift is based on the gimbal forward kinematics (i.e., the reference attitude). Two main issues need to be addressed in this approach: the joint axis orientation misalignment and the encoder nonlinearity. The encoder nonlinearity, i.e., deviation between the measured and actual joint angles, is usually caused by imperfect magnets in joint motors and location errors of hall sensors. Considering that the axis orientation and encoder nonlinearity constitute the kinematic parameters of a gimbal system, they need to be identified correctly in the calibration.

This paper investigates the kinematic calibration of cost-effective gimbal systems. The main contributions are summarized as follows. (1) We have proposed a calibration algorithm that can simultaneously and automatically calibrate the kinematic parameters (i.e., axis orientations and encoder nonlinearity) of a multi-axis gimbal system and the gyro bias of the endpoint IMU, without using any external measurement devices. The calibration algorithm is verified by both simulation and actual experiments on a 3-axis gimbal system. (2) We have developed an analytical error model for the calibration algorithm and numerically verified its effectiveness.

II. Related Work

Inertially stabilized gimbal systems are traditionally used to control the line-of-sight (LOS) of devices on moving platforms (e.g., cameras, televisions, and laser) [6]. Rue [7] analyzed the effect of axis orientation misalignment and transducer readout

error (i.e., the encoder nonlinearity) on the LOS of a two-axis gimbal system using the Pio's method [8]. The ground truth of the LOS direction are provided using a trunnion mirror and a two-axis auto-collimator, and the orientation misalignment and transducer errors are identified by solving a group of linear equations under the assumption that they are small enough. Wang *et al* [9] considered the calibration of a two-axis gimbal system in the context of enhancing the accuracy of an inertial navigation systems. In this context, the inertial sensors (i.e., gyroscope and accelerometer) are assumed un-calibrated and being subject to various unknown errors such as bias, scale factors, and misalignment. As to the gimbal system, only the non-orthogonal angle representing the axis orientation misalignment is assumed unknown while the encoder nonlinearity is assumed known or negligible. Wang *et al* formulated the error model of such inertial navigation systems into a state space model with vertical-channel damping and utilized an extended Kalman filter to estimate all unknown parameters (e.g., non-orthogonal angle). One feature of this calibration algorithm is that it does not require external sensors to measure the gimbal pose. Similar non-orthogonal angle calibration has been studied for three-axis gimbal systems as well (e.g., [10], [11]).

Since a gimbal system can be considered as a serial chain mechanism, some calibration methods for robotic manipulator can be borrowed. One approach is based on the Denavit-Hartenberg (DH) parameterization [12] and its variants (e.g., [13]–[16]). This approach is usually highly dependent on *ad hoc* kinematics representations to fix the singularity problem. Another approach is based on the product-of-exponential (POE) formula [17]. Okamura *et al* [18] employed the POE formula and derived an iterative least square algorithm for robot calibration. Particularly, the closed-form derivatives of robot kinematics with respect to the joint axes and angle are derived based on Lie theory, based on which several variants of calibration algorithms have been proposed (e.g., [19]–[22]). These variants address issues such as convergence rate but all within the iterative least square optimization frame as in [18]. Consequently, they all require iterative differentiation for the parameter-varying exponential maps, which may not be implementable in cost-effective platforms with limited computation resources. Moreover, these algorithms either need external devices such as laser trackers or calibration boards to measure the endpoint location and orientation (e.g., [23]–[31]), require the manipulator to form a closed chain (e.g., [32]), or are incapable of calibrating the encoder nonlinearity (e.g., [33]).

Our prior work [34] calibrated the gimbal kinematics automatically using an endpoint IMU, in which the gyro bias was not explicitly considered. This paper proposes a systematic method that calibrates the gimbal kinematics (i.e., the axis orientation and encoder nonlinearity) automatically using the endpoint gyro while explicitly compensating its gyro bias. In the proposed calibration method, each gimbal joint is moved back and forth individually, which not only decouples the calibration of joint axis orientation and encoder nonlinearity but also directly implies the gyro bias. The calibration algorithm with gyro bias compensation is derived under the rigorous assumption of single joint movement, and the error model of unwanted joint movements caused by joint control errors is developed and analyzed. Instead of only considering one non-orthogonal angle [9]–[11] or small linearized misalignment and nonlinearity [7], the method presented in this paper is readily extendable to multiple axes with both axis misalignment and encoder nonlinearity of large values. Furthermore, compared with the gimbal calibration in [7], [35] and robot kinematics calibration work reviewed previously, the proposed method has the following advantages: (1) it simultaneously calibrates the kinematic parameters (both axis orientations and encoder nonlinearity) of a multi-axis gimbal system and the gyro bias of the endpoint IMU without using any external devices such as the trunnion mirror or laser tracker, and thus the calibration can be fully automated; (2) it has a closed-form solution which only needs efficient algebraic computation and is easy for embedded implementation. The proposed calibration algorithm is validated by both numerical studies and experimental tests on a cost-effective 3-axis gimbal system.

The remainder of this paper is organized as follows. Section III introduces the kinematic model of a multi-axis gimbal system. Section IV presents the proposed calibration algorithm and its error model. Sections V and VI perform the numerical studies and experimental tests, respectively. Section VII concludes the paper.

III. Gimbal System Kinematic Model

By treating the multi-axis gimbal system as a serial manipulator, in this paper we use the POE formula to model its kinematics [36].

$$\mathbf{R}(\boldsymbol{\theta}(t)) = \mathbf{R}(\boldsymbol{\theta}(0))e^{\widehat{\boldsymbol{\xi}}_1 h_1(\theta_1(t))} \cdots e^{\widehat{\boldsymbol{\xi}}_n h_n(\theta_n(t))} \tag{1}$$

where $\mathbf{R}(\boldsymbol{\theta}(0))$ is the payload (i.e., endpoint IMU) attitude (when all joints are at their zero positions) relative to the host vehicle that is static during the calibration; $\boldsymbol{\xi}_i$ ($\|\boldsymbol{\xi}_i\|$=1, i=1, $\cdots$, n) are the axis orientation represented in the endpoint gyro frame; the hat operation $\widehat{\boldsymbol{\xi}}_i$ forms a skew-symmetric matrix; $\theta_i(t)$ is the measured angle of the i-th joint; and $h_i(\cdot)$ maps the measured joint angle to its actual value. It is usually problematic to identify an arbitrary nonlinear map $h_i(\theta)$in practice. Here we characterize it by a group of known, either local or global, basis functions.

$$h_i(\theta) = \mathbf{c}_i^T \mathbf{g}_i(\theta); \quad i = 1, 2, \cdots, n \tag{2}$$

where $\mathbf{g}_i(\theta)$= $\begin{bmatrix} g_i^1(\theta) & \cdots & g_i^m(\theta) \end{bmatrix}^T$ is the vector of basis function, and $\mathbf{c}_i$ is the corresponding coefficients vector. Some commonly used local basis functions include first order interpolation schemes and spline models. As no specific zero joint position is required in (1), we choose it to be the position where encoder measurement is zero (i.e., $h_i(\theta) = 0$ for $\theta = 0$). Finally, we assume that the nonlinearity map $h_i(\cdot)$ monotonically increases as moving each joint from the minimum position to the maximum always increases the encoder measurements in practice.

To simplify notations, we abbreviate $\mathbf{R}(\boldsymbol{\theta}(t))$ as $\mathbf{R}(t)$, $h_i(\theta_i(t))$ as $h_i(t)$, and $\mathbf{g}_i(\theta_i(t))$ as $\mathbf{g}_i(t)$ in the following discussion.

IV. Kinematic Calibration and Drift Compensation

With the kinematic model (1), the gimbal system kinematic calibration refers to determining the axis orientation $\boldsymbol{\xi}_i$ and encoder nonlinearity map $h_i(\cdot)$ (i.e., $\mathbf{c}_i$) of all joints. With calibrated kinematic parameters (i.e., $\boldsymbol{\xi}_i$ and $\mathbf{c}_i$), the payload attitude can be computed from the forward kinematic model (1) and fused with the endpoint gyro integration [3], [4] to produce a high-fidelity payload attitude estimate.

A. Algorithm Derivation

A straightforward calibration method is to minimize the residual error formed by (1) as presented in [18]–[22]. However, the extremely nonlinear constraints in (1) inevitably lead to an iterative procedure, which usually requires considerable memory and computation resources that are not available in cost-effective miniaturized gimbal systems. In this paper, the calibration of different joints and the calibration of axis orientation and encoder nonlinearity of each joint is decoupled by moving only one joint (e.g., the k-th joint) at a time while keeping the others at zero positions (e.g., $\theta_i = 0, h_i(\theta_i) = 0; i \neq k$). For example, if only moving the k-th joint, Equation (1) implies

$$e^{\widehat{\boldsymbol{\xi}}_k (h_k(t) - h_k(0))} = \mathbf{R}(0)^T \mathbf{R}(t) \tag{3}$$

Differentiating (3) leads to the body angular rate $\boldsymbol{\omega} \triangleq \mathbf{R}^T \dot{\mathbf{R}}$ as below

$$\boldsymbol{\omega} = \boldsymbol{\xi}_k \dot{h}_k \tag{4}$$

Then, integrating (4) yields

$$\int_0^t \boldsymbol{\omega} d\tau = \boldsymbol{\xi}_k \left(h_k(t) - h_k(0)\right) \tag{5}$$

To calibrate the kinematic parameters $\boldsymbol{\xi}_k$ and $h_k(\cdot)$, the integration $\int_0^t \boldsymbol{\omega} d\tau$ needs to be obtained, which can be achieved based on gyro measurements. However, MEMS gyros are usually susceptible to drifts, leading to biased measurements of $\int_0^t \boldsymbol{\omega} d\tau$. A commonly used model for MEMS gyros is

$$\boldsymbol{\omega}_m = \boldsymbol{\omega} + \mathbf{b} \tag{6}$$

where $\mathbf{b}$ is a constant bias vector. The gyro measurement noise was ignored in (6).

To compensate such gyro drift, the joint is moved back and forth as shown in Fig. 2. Then, two points (e.g., point t_A and t_B in Fig. 2) are taken respectively on the rising and falling edges which have the same joint measurements (i.e., $\boldsymbol{\theta}(t_A) = \boldsymbol{\theta}(t_B)$). As the encoder nonlinearity map is monotonically increasing, identical joint measurements implies identical joint positions (i.e., $h_k(t_A) = h_k(t_B)$), which leads to $\int_{t_A}^{t_B} \boldsymbol{\omega} d\tau = \mathbf{0}$ from (5). Hence,

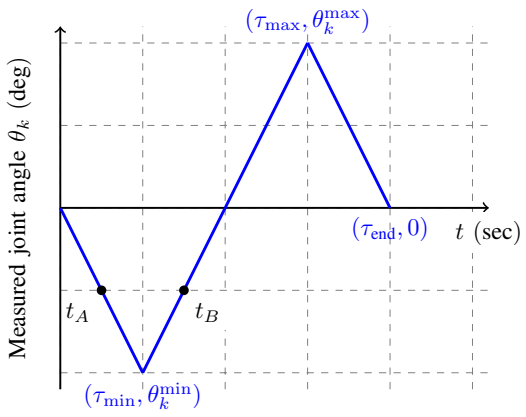


Fig. 2. Joint encoder readings during the calibration.

$$\begin{aligned} \int_{t_A}^{t_B} \boldsymbol{\omega}_m d\tau &= \int_{t_A}^{t_B} \boldsymbol{\omega} d\tau + \mathbf{b}(t_B - t_A) \\ &= \mathbf{b}(t_B - t_A) \end{aligned} \tag{7}$$

Applying (7) to many different sample pairs yields the optimal bias estimate

$$\bar{\mathbf{b}} = \frac{1}{M} \sum_{l=1}^{M} \frac{\int_{t_{A_l}}^{t_{B_l}} \boldsymbol{\omega}_m d\tau}{\Delta t_l} \tag{8}$$

where $\Delta t_l = t_{B_l} - t_{A_l}$ is the time difference and M is the number of sample pairs for bias estimation.

With the bias estimate, the angular rate integration $\int_0^t \boldsymbol{\omega} d\tau$ can be estimated as follows:

$$\int_0^t \bar{\boldsymbol{\omega}} d\tau = \int_0^t \boldsymbol{\omega}_m d\tau - \bar{\mathbf{b}} \cdot t \tag{9}$$

Let $\boldsymbol{\psi}_k(t) = \int_0^t \bar{\boldsymbol{\omega}} d\tau$ be the estimated angular rate integration, then (5) implies

$$\boldsymbol{\psi}_k(t) = \boldsymbol{\xi}_k \left(h_k(t) - h_k(0)\right) \tag{10}$$

which decomposes into the following two separate constraints: one for the axis orientation $\boldsymbol{\xi}_k$ and the other one for the encoder nonlinearity map $h_k(\cdot)$,

$$h_k(t) - h_k(0) = \phi_k(t) \tag{11a}$$

$$\boldsymbol{\xi}_k = \boldsymbol{\zeta}_k(t) \tag{11b}$$

where

$$\phi_k(t) = s \|\boldsymbol{\psi}_k(t)\| \tag{12a}$$

$$\boldsymbol{\zeta}_k(t) = s \frac{\boldsymbol{\psi}_k(t)}{\|\boldsymbol{\psi}_k(t)\|} \tag{12b}$$

respectively denote the angle and the orientation of the axis-angle vector $\boldsymbol{\psi}_k(t)$, and s denotes the direction of joint movements. Because $h_k(\cdot)$ is monotonically increasing, s can be determined as below

$$s = \begin{cases} 1 & \text{for} \quad \theta(t) \geq \theta(0) \\ -1 & \text{for} \quad \theta(t) < \theta(0) \end{cases} \tag{13}$$

Moving the k-th joint at many (i.e., N) positions yields a group of linear equations for $h_k(\cdot)$ and $\boldsymbol{\xi}_k$:

$$\underbrace{\begin{bmatrix} \mathbf{g}_k(t_1)^T - \mathbf{g}_k(0))^T \\ \vdots \\ \mathbf{g}_k(t_N)^T - \mathbf{g}_k(0))^T \end{bmatrix}}_{\mathbf{A}_k} \mathbf{c}_k = \underbrace{\begin{bmatrix} \phi_k(t_1) \\ \vdots \\ \phi_k(t_N) \end{bmatrix}}_{\mathbf{y}_k} \tag{14}$$

Then the optimal coefficients $\mathbf{c}_k$ can be analytically derived as

$$\mathbf{c}_k = \left(\mathbf{A}_k^T \mathbf{A}_k\right)^{-1} \mathbf{A}_k^T \mathbf{y}_k \tag{15}$$

To find the optimal joint axis $\boldsymbol{\xi}_k$ from $\boldsymbol{\zeta}_k(t_j), (j = 1, \cdots, N)$, where $\boldsymbol{\xi}_k$ and $\boldsymbol{\zeta}_k(t_j)$ are both on a unit sphere in $\mathbb{R}^3$, the following cosine distance is used to measure their distances:

$$D_c(\mathbf{u}, \mathbf{v}) = 1 - \mathbf{u}^T \mathbf{v}; \|\mathbf{u}\| = 1, \|\mathbf{v}\| = 1 \tag{16}$$

which essentially reflects the relative orientation of the two vectors. Minimizing the summed cosine distance between $\boldsymbol{\xi}_k$ and all $\boldsymbol{\zeta}_k(t_j), j = 1, \cdots, N$ yields

$$\max_{\|\boldsymbol{\xi}_k\|=1} \sum_{j=1}^{N} \boldsymbol{\xi}_k^T \boldsymbol{\zeta}_k(t_j) \tag{17}$$

whose solution can be found analytically

$$\boldsymbol{\xi}_k = \frac{\sum \boldsymbol{\zeta}_k(t_j)}{\| \sum \boldsymbol{\zeta}_k(t_j)\|} \tag{18}$$

It is noted that the gyro bias in (8), encoder nonlinearity in (15), and axis orientation in (18) are all estimated from the same data collected along the joint movements in Fig. 2. The complete algorithm for calibrating one joint (e.g., k-th joint)) is summarized in **Algorithm. 1**. In Line 2, θ_i^{sp} is the setpoint of the i-th joint angle that is fed to an underlying joint controller. In Line 5, θ_i is the current joint angle and needs to converge before collecting data (i.e., the error should be smaller than a prescribed threshold ϵ). In Line 6, t_c is the current time of the system, $\theta_k(t_c)$ is the current angle of the k-the joint, and $\mathbf{w}(t_c)$ is the gyro integration (three-channel) up to the current time. It should be mentioned that the gyro integration $\mathbf{w}$ is integrated at the rate of gyro measurements, which is usually much higher than the rate of joint angle samples. They are respectively saved in $\mathbf{T}, \boldsymbol{\Theta}$ and $\mathbf{W}$ and used to estimate the gyro bias in Lines 8-10, joint axis orientation and encoder nonlinearity map in Lines 11-17.

Remark 1: The assumption that the encoder nonlinear map $h(\cdot)$ is monotonically increasing can be relaxed. For non-monotonic $h(\cdot)$, the sign s in (12) can be determined in two ways. The first is to start moving a joint from its limit position, then s takes either positive or negative values. The second is to reformulate (17) as $\max_{\|\boldsymbol{\xi}_k\|=1} \sum_j^m \left(\boldsymbol{\xi}_k^T \boldsymbol{\zeta}_k(t_j)\right)^2$, whose solution is the principal singular vector of $\sum \boldsymbol{\zeta}_k(t_j) \left(\boldsymbol{\zeta}_k(t_j)\right)^T$ (or the negative of it). After obtaining $\boldsymbol{\xi}_k$, s is the sign of $\boldsymbol{\xi}_k^T \boldsymbol{\zeta}_k(t_j)$ and $h(\cdot)$ can be determined subsequently.

Remark 2: The proposed calibration algorithm can not calibrate the payload initial orientation (i.e., $\mathbf{R}(0)$ in (3)) because the endpoint gyro cannot measure the absolute orientations with respect to the host vehicle.

Algorithm 1 Calibration of the k-th joint of a gimbal

1: **procedure** GIMBALCAL(k)
2: Set $\theta_i^{\text{sp}} = 0, i = 1, \cdots, n$;
3: $\mathbf{T} = \text{NULL}$; $\boldsymbol{\Theta} = \text{NULL}$; $\mathbf{W} = \text{NULL}$; $\boldsymbol{\xi}_k = \mathbf{0}$;
4: **for** each setpoint θ_k^{sp} in Fig. 2 **do**
5: Wait until $|\theta_i - \theta_i^{\text{sp}}| < \epsilon$ for all i;
6: $\mathbf{T} = \mathbf{T} \bigcup t_c$; $\boldsymbol{\Theta} = \boldsymbol{\Theta} \bigcup \theta_k(t_c)$; $\mathbf{W} = \mathbf{W} \bigcup \mathbf{w}(t_c)$;
7: **end for**
8: Find all index pairs (A_l, B_l) in $\boldsymbol{\Theta}$ such that
9: $\theta_{A_l} = \theta_{B_l}$ for $l = 1, \cdots, M$;
10: Compute the gyro bias $\bar{\mathbf{b}}$ as in (8);
11: **for** each point t in $\mathbf{T}$ **do**
12: Compute $\boldsymbol{\psi}_k(t)$ as in (9);
13: Compute $\boldsymbol{\zeta}_k(t)$ and $\phi_k(t)$ from $\boldsymbol{\psi}_k(t)$ as in (12);
14: Append $\phi_k(t)$ to $\mathbf{y}_k$ as in (14); $\boldsymbol{\xi}_k = \boldsymbol{\xi}_k + \frac{\boldsymbol{\zeta}_k(t)}{\|\boldsymbol{\zeta}_k(t)\|}$;
15: Append $\mathbf{g}(\theta_k(t)) - \mathbf{g}(\theta_k(0))$ to $\mathbf{A}_k$ as in (14);
16: **end for**
17: $\boldsymbol{\xi}_k = \frac{\boldsymbol{\xi}_k}{\|\boldsymbol{\xi}_k\|}$; $\mathbf{c}_k = \left(\mathbf{A}_k^T \mathbf{A}_k\right)^{-1} \mathbf{A}_k^T \mathbf{y}_k$;
18: Return $\boldsymbol{\xi}_k, \mathbf{c}_k$;
19: **end procedure**

B. Error Analysis and Mitigation

The calibration and drift compensation algorithms derived in the previous section require that only one gimbal joint is moved at a time while other joints are kept at perfect zero positions. Due to the limited bandwidth of a practical joint controller, there could be residual control errors (i.e., nonzero joint positions). In this section, we analyze the effect of such joint control errors, especially of those non-rotating joint axes. The control errors on non-moving joints would degrade the calibration performance in two ways. First, due to the movement of these joints, equation (5) no longer holds and hence introduces errors to the kinematics calibration. Second, the movement of other joints would cause errors in gyro bias estimate, which in turn degrades the kinematics calibration.

For the first one, we analyze the deviation of $\int_0^t \boldsymbol{\omega} d\tau$ from its expectation $\boldsymbol{\xi}_k \left(h_k(t) - h_k(0)\right)$ in (5) when other non-moving joints are not ideally at their zero positions due to control errors. As shown in [36], the body angular rate $\boldsymbol{\omega}$ can be derived from (1) and takes the following form

$$\boldsymbol{\omega} \quad = \quad \log\left(\mathbf{R}^T \dot{\mathbf{R}}\right) = \mathbf{J}(\mathbf{h}) \dot{\mathbf{h}} \tag{19}$$

where $\mathbf{h} = \mathbf{h}(\boldsymbol{\theta}) = \begin{bmatrix} h_1(\theta_1) & \cdots & h_n(\theta_n) \end{bmatrix}^T$ abbreviates the encoder nonlinear map vector, and $\mathbf{J}(\mathbf{h})$ denotes the Jacobian matrix,

$$\mathbf{J}(\mathbf{h}) = \begin{bmatrix} e^{-\widehat{\boldsymbol{\xi}}_n h_n} \cdots e^{-\widehat{\boldsymbol{\xi}}_2 h_2} \boldsymbol{\xi}_1, & \cdots & , \boldsymbol{\xi}_n \end{bmatrix} \tag{20}$$

Then, integrating (19) yields the deviation of $\int_0^t \boldsymbol{\omega} d\tau$ from its expectation $\boldsymbol{\xi}_k \int_0^t \dot{h}_k d\tau$

$$\int_0^t \boldsymbol{\omega} d\tau - \boldsymbol{\xi}_k \int_0^t \dot{h}_k d\tau = \int_0^t \widetilde{\mathbf{J}}(\mathbf{h}) \dot{\mathbf{h}} d\tau \tag{21}$$

where

$$\widetilde{\mathbf{J}}(\mathbf{h}) = \begin{bmatrix} \widetilde{\mathbf{J}}_1, & \cdots & , \widetilde{\mathbf{J}}_k & , \cdots & , \widetilde{\mathbf{J}}_n \end{bmatrix} \tag{22}$$

and

$$\widetilde{\mathbf{J}}_i = \begin{cases} e^{-\widehat{\boldsymbol{\xi}}_n h_n} \cdots e^{-\widehat{\boldsymbol{\xi}}_{i+1} h_{i+1}} \boldsymbol{\xi}_i & ; \quad i \neq k \\ e^{-\widehat{\boldsymbol{\xi}}_n h_n} \cdots e^{-\widehat{\boldsymbol{\xi}}_{k+1} h_{k+1}} \boldsymbol{\xi}_k - \boldsymbol{\xi}_k & ; \quad i = k \end{cases} \tag{23}$$

For the second effect, let $\Delta\mathbf{b} = \mathbf{b} - \bar{\mathbf{b}}$ denote the bias estimation error caused by the control errors of non-moving joints. Then, the estimated angular rate integration $\int_0^t \bar{\boldsymbol{\omega}} d\tau$ in (9), which is used to calibrate the axis orientation and encoder nonlinearity map, becomes

$$\begin{aligned} \int_0^t \bar{\boldsymbol{\omega}} d\tau &= \int_0^t \boldsymbol{\omega}_m d\tau - \bar{\mathbf{b}} \cdot t \\ &= \int_0^t \boldsymbol{\omega} d\tau + \Delta\mathbf{b} \cdot t \end{aligned} \tag{24}$$

Substituting (21) into (24) yields the combined error

$$\int_0^t \bar{\boldsymbol{\omega}} d\tau = \boldsymbol{\xi}_k \int_0^t \dot{h}_k d\tau + \int_0^t \widetilde{\mathbf{J}}(\mathbf{h}) \dot{\mathbf{h}} d\tau + \Delta\mathbf{b} \cdot t \tag{25}$$

As a result, the error between the estimated angular rate integration and its expected value is

$$\mathbf{E}_k(t) = \int_0^t \widetilde{\mathbf{J}}(\mathbf{h}) \dot{\mathbf{h}} d\tau + \Delta\mathbf{b} \cdot t \tag{26}$$

Ideally, when the non-moving joints are at perfect zero positions (i.e., $\theta_i{=}\dot{\theta}_i{=}0, h_i{=}\dot{h}_i{=}0; i{\neq}k$) and the bias are accurately estimated (i.e., $\Delta\mathbf{b}{=}\mathbf{0}$), then $\widetilde{\mathbf{J}}(\mathbf{h})\dot{\mathbf{h}}{=}\mathbf{0}$ and $\mathbf{E}_k(t){=}\mathbf{0}$, which implies accurate estimate of the rate integration (i.e., $\int_0^t \bar{\boldsymbol{\omega}} d\tau{=}\boldsymbol{\xi}_k \left(h_k(t){-}h_k(0)\right)$)

In practice, with the control errors, the non-moving joints are usually not exactly at zero positions. Assume the residual positions of non-moving joints are upper bounded by ϵ and the moving joint (e.g. k-th joint) has its minimal angle at $h_k^{\min}$ and maximal angle at $h_k^{\max}$, then as proved in the Appendix, the error on the k-th joint ($k = 1, 2, ..., n$) caused by residual positions is bounded by the following quantities.

$$\begin{aligned} \left\| \int_0^t \widetilde{\mathbf{J}}(\mathbf{h}) \dot{\mathbf{h}} d\tau \right\| \leq (k-1) \big(2 \left|\sin(h_k^{\min})\right| + 2 \left|\sin(h_k^{\max})\right| \\ + 2 \left|1 - \cos(h_k^{\min})\right| + 2 \left|1 - \cos(h_k^{\max})\right| + 3 \big) \epsilon \\ + (n-k) \left(1 + 2|h_k^{\min}| + 2|h_k^{\max}|\right) \epsilon \end{aligned} \tag{27}$$

With the error bound given in (27), we come up with the following measures to mitigate the effect of joint control errors in practice: (1) improve joint control performance. For example, a joint controller with high disturbance attenuation (e.g., an integral action in the joint controller) can effectively compensate external disturbances or internal dynamics coupling effect (e.g., Coriolis force) and reduce the residual error of the non-moving joints. Compensating the dynamics coupling effect based on accurate serial chain mechanism dynamic model can also improve the joint control performance when the joint dynamic parameters are available; (2) verify that the encoder measurements of non-moving joints are near to zero before collecting a sample data for calibration; and (3) use the relative rotations between two successive measurements to prevent the error caused by gyro bias estimation error from growing linearly with respect to time (see (26)).

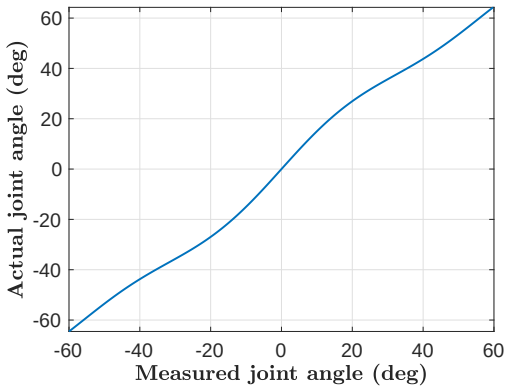


Fig. 3. An exemplary nonlinearity map.

V. Simulation Studies

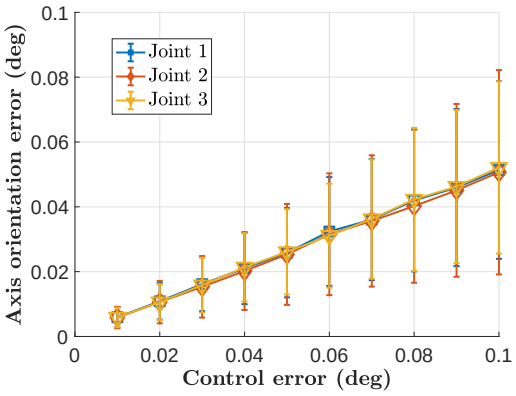
A. Simulation Setup

To verify the efficacy of the proposed calibration algorithm and the error model, 1000 trials of simulated calibration are conducted on a 3-axis gimbal system. In each trial, the ground-truth orientation of the three joints $\boldsymbol{\xi}_1, \boldsymbol{\xi}_2$ and $\boldsymbol{\xi}_3$ are randomly perturbed from their nominal design, $\mathbf{e}_x = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T, \mathbf{e}_y = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^T$ and $\mathbf{e}_z = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$, respectively. The magnitude perturbation is 10 degrees to simulate the manufacturing imperfections. Likewise, the encoder nonlinearity map is composed of five basis functions: $\mathbf{g}(\theta) = \begin{bmatrix} \theta & \sin(2\pi\theta) & \sin(\pi\theta) & \sin(\frac{2\pi}{3}\theta) & \sin(\frac{\pi}{2}\theta) \end{bmatrix}^T$ to simulate the joint encoder nonlinearity. The first element of the coefficient vector $\mathbf{c}$ for the basis function of each joint is always one, while the rest are uniformly sampled in the range of $[0, 0.05]$. This ensures that the nonlinearity map is dominated by the function $h(\theta) = \theta$ (i.e., the encoder nonlinearity is small). An exemplary nonlinearity map resulting from this setup is shown in Fig. 3. The moving range for all joints (as seen by encoders with nonlinearity map) is set to $\left[-\frac{\pi}{3}, \frac{\pi}{3}\right]$, or $\theta_i^{\min} = -\frac{\pi}{3}$ and $\theta_i^{\max} = \frac{\pi}{3}$ for $i = 1, 2, 3$.

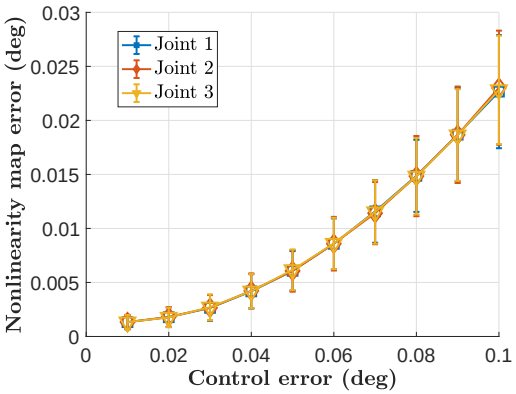
With the above configurations of axis orientations and encoder nonlinearities, each joint (as seen by encoders with nonlinearity map) is moved along the profile in Fig. 2, with the speed of $120°$/sec and joint sample interval of $3°$. This implies that a total of $N = 80$ sample triplets (i.e., the length of $\mathbf{T}, \boldsymbol{\Theta}$, and $\mathbf{W}$ in **Algorithm. 1**) will be collected and the calibration time for one joint will last 2 seconds. For the non-moving joints, a band-limited disturbance is added to their joint angles to simulate the control errors. The bound for the disturbance is ϵ and its effect will be evaluated in the simulation.

After the joint angles $\theta_i(t)$ (as seen by encoders with the effect of nonlinearity) are determined, we compute the actual joint angles $h_i(\theta_i(t))$, and hence the endpoint gyro angular rate from (19). The angular rate is computed at $2kHz$, a typical gyro sampling rate (with some pre-filtering). The computed payload angular rate is the actual (i.e., ground truth) angular rate, and is added with a bias equal to $0.01 rad/s$ to simulate the gyro bias.

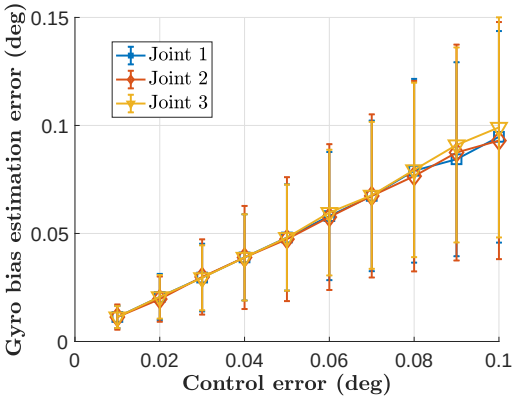
Finally, because encoder readings and gyro data are usually contaminated by measurements noise, a white noise in the



(a) Axis orientation error versus joint control error ϵ.

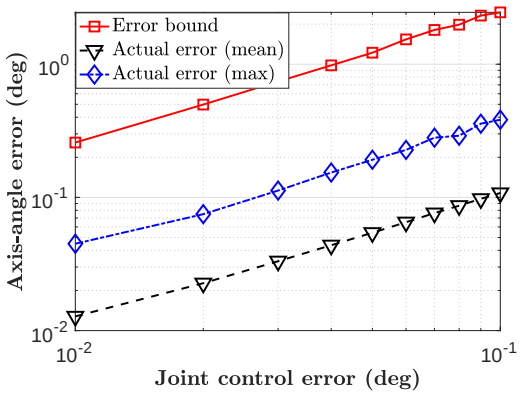


(b) Encoder nonlinearity map error versus joint control error ϵ.

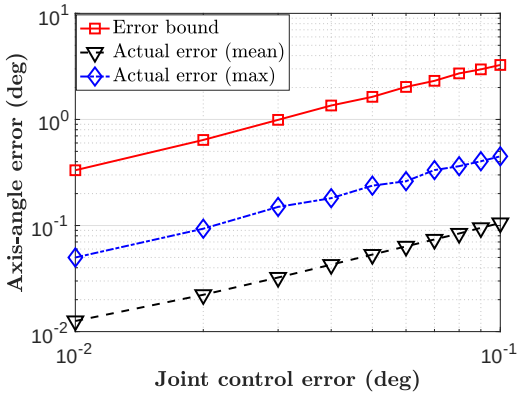


(c) Gyro bias estimation error versus joint control error ϵ.

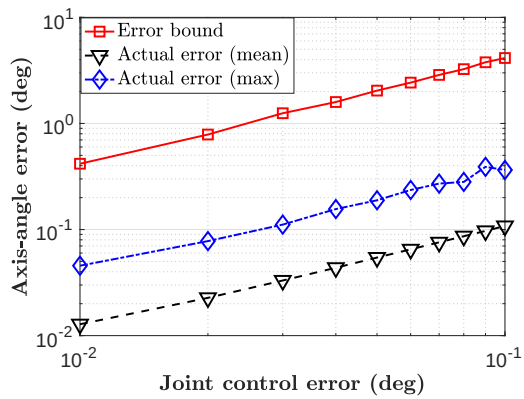
Fig. 4. Calibration error over different level of control error ϵ.



(a) Component X.



(b) Component Y.



(c) Component Z.

Fig. 5. Axis-angle error over different level of control error ϵ. For better visualization, the plots are shown in log scale.

range of $[-0.01^\circ, 0.01^\circ]$ is added to the ground truth joint angles $\theta_i(t)$ and the gyro integration.

B. Simulation Evaluation

To evaluate the proposed calibration algorithm and validate the error model, two performance metrics are examined. One is the calibration error of axis orientation, encoder nonlinearity map, and gyro bias at different levels of control errors ϵ over trials: the calibration error of k-th joint's axis orientation and encoder nonlinearity map for the j-th trial is defined as follows

$$\tilde{\boldsymbol{\xi}}_k^{(j)} \triangleq \cos^{-1}\left(\boldsymbol{\xi}_k^T \overline{\boldsymbol{\xi}}_k^{(j)}\right) \tag{28a}$$

$$\tilde{h}_k^{(j)} \triangleq \text{mean}\left(h_k(\theta) - \overline{h}_k^{(j)}(\theta)\right) \tag{28b}$$

where $\overline{\boldsymbol{\xi}}_k^{(j)}$ and $\overline{h}_k^{(j)}$ respectively denote the calibrated axis orientation and encoder nonlinearity map, and the mean in (28b) is conducted within the range of $\left[\theta_k^{\min}, \theta_k^{\max}\right]$ with the step size of 0.01°; The other performance metrics is the estimation error on axis-angle vector $\boldsymbol{\psi}_k(t)$ versus its error

bound given by (26). This is used to verify the effectiveness of our error analysis.

Fig. 4(a) shows the mean and standard deviation of axis orientation error for all 1000 trials over different level of control errors ϵ. Likewise, Fig. 4(b) and Fig. 4(c) respectively show the encoder nonlinearity map error and gyro bias error. It can be seen that the axis orientation estimation error and the gyro bias estimation error both grow linearly with respect to the joint control error ϵ. Though the nonlinearity map error seems quadratic to the joint control error, the overall error remains very small.

Furthermore, Fig. 5 shows the axis-angle error (i.e., $\mathbf{E}_k(t)$ in (26)) over different levels of control errors ϵ. Since for each trial the axis-angle error is slightly different (i.e., the $h_{\min}, h_{\max}$, and $\Delta\mathbf{b}$ are different), the error bound at each joint control error is the maximal value over all trials. For better visualization, both plots are shown in log scale. It can be seen that the actual axis-angle error remains well below our predicted error bound, which verifies our proposed error model. Additionally, both the actual axis-angle error and its error bound grow linearly (with slope equal to 1 in the log scale) with respect to the joint control error, which agrees with the prediction in (27).

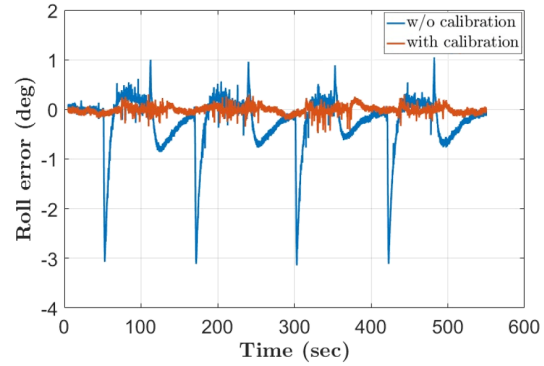
VI. Experimental Validation

In this section, the proposed calibration algorithm is implemented and verified on a DJI Mavic Pro onboard gimbal system (see Fig. 1). The X, Y and Z axes of the endpoint gyro are respectively aligned with the gimbal roll, pitch, and yaw directions. In the experiment, each to-be-calibrated joint axis moves along the profile provided in Fig. 2; the calibration data are collected every 1 degree. Instead of global basis functions, the encoder nonlinearity map is fitted using first-order interpolation due to its reduced requirements of memory and computation.

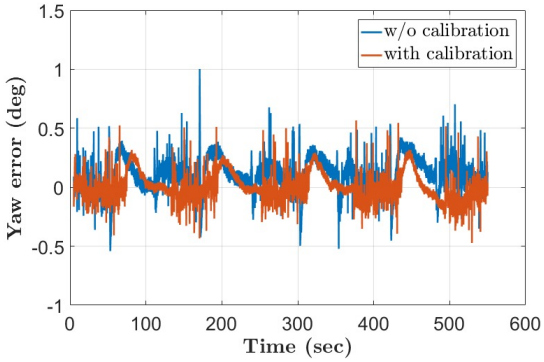
Using the calibrated axis orientation and encoder nonlinearity map, the reference attitude is computed by the gimbal forward kinematics in (1) and fused with the gyro measurements with a multiplicative Kalman filter [37]. The estimated attitude of the payload (i.e., camera) is then used as the feedback signal for the gimbal controller.

To evaluate the calibrated results, we examine the a-priori estimation error in the EKF. If the axis orientation and encoder nonlinearity are accurately calibrated, the reference attitude computed by the forward kinematics should be equal to the a-priori attitude estimate based on gyro integration. Otherwise, a residual a-priori estimation error will be present. To trigger the residual a-priori attitude error caused by imperfect calibrations, we move the pitch angle in $[0^\circ, -90^\circ]$ back and forth.

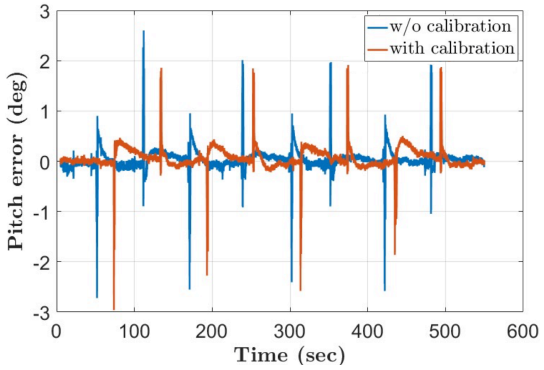
Fig. 6 shows the a-priori attitude error in all three directions. In this figure, we compare the a-priori errors between the case where the kinematic parameters are calibrated with the proposed calibration algorithm and the case where the nominal kinematic parameter values are used. It can be seen that without the calibration, the a-priori estimation error is quite large. For the roll direction, the error is up to 3° degrees at the moment the pitch angle moves from 0° to -90° (which occurs at approximately 50 sec, 170 sec, 305 sec, and 425 sec). At the moment the pitch angle moves from -90° back to 0° (which occurs at approximately 120 sec, 250 sec, 355 sec, and 495 sec), the error is excited again and up to 0.8 degrees. In contrast, with the calibration, the a-priori is much smaller and is below 0.5 degrees in all circumstances. For the yaw direction, the average error reduces from 0.4° to 0.2°. This reduction is much less than that in the roll direction because gimbal axis orientation error and encoder nonlinearity are already quite small as indicated by the blue line in Fig. 6(b). For the pitch direction, we observe that there are unusual large spikes in the a-priori error, with or without calibration, every time the pitch axis rotates from 0° to -90° or vice versa. Carefully inspecting the data we find that the spike only occurs when the pitch axis is fast rotating from 0° to -90° or back forth. Therefore, they are likely



(a) Attitude error in Roll direction.

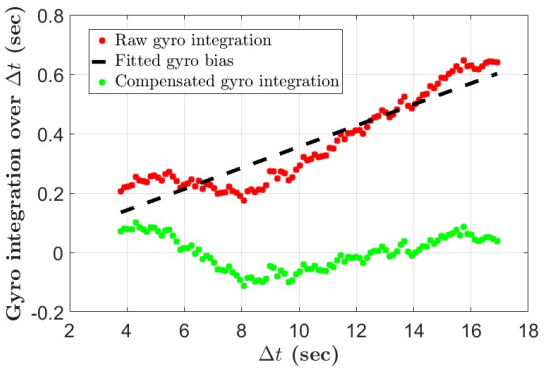


(b) Attitude error in Yaw direction.

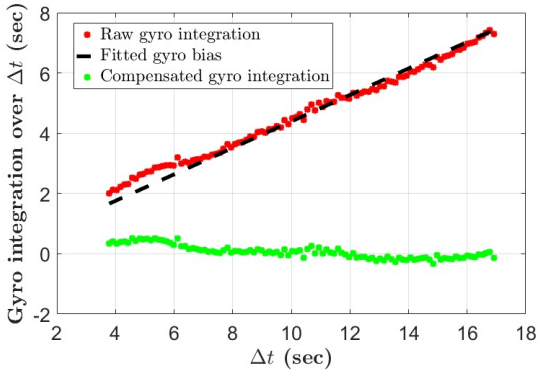


(c) Attitude error in Pitch direction.

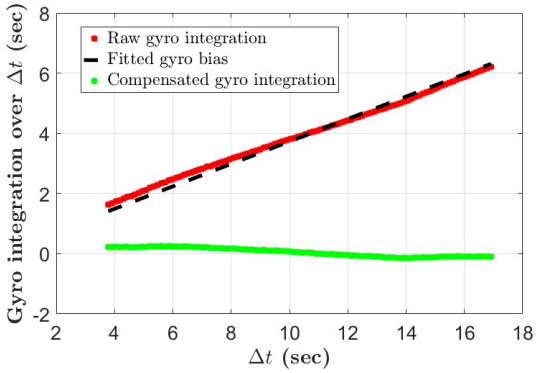
Fig. 6. Attitude estimation error



(a) Component X.



(b) Component Y.



(c) Component Z.

Fig. 7. Gyro integration between two time instants where the joint measurements are the same versus their time difference.

caused by the measurement delay in the encoder and should not be considered in evaluating the calibration performance. This also agrees with the fact that these spikes (1) only appeared in the pitch axis that is fast rotating; (2) reverse signs whenever the pitch axis reverses its rotating direction; (3) did not appear in the simulation that suffers from no encoder measurement delay. Excluding these spikes, we find that when the pitch axis rotates from 0° back to -90° (or vice versa), the a-priori error reduction with the calibration is quite significant, from 1° to be below 0.5°. The considerable improvement on attitude estimation justifies the necessity and efficacy of the proposed calibration algorithm. The achieved accuracy, although less than that in industry robots [24], is much sufficient for consumer-grade aerial photograph. Finally, it is worthwhile mentioning that the pitch movement in the two experiments are not accurately synchronized because they are separately conducted with manual pitch commands. This leads to a time offset between the spikes of these two experiments.

Fig. 7 shows the gyro integration (i.e., $\int_{t_A}^{t_B} \boldsymbol{\omega}_m d\tau$) over the time difference $\Delta t = t_B - t_A$ where the encoder readings are identical (i.e., $\boldsymbol{\theta}(t_B) = \boldsymbol{\theta}(t_A)$. It can be seen that all the three components of gyro integration vary linearly with the time difference and can be well interpolated by straight lines, whose slopes are the gyro bias. After subtracted from the fitted gyro bias, the gyro integration is around zero. This effectively verifies our assumptions for gyro bias estimation.

VII. Conclusion

This paper has presented a calibration method for cost-effective gimbal systems that have very limited computation and memory resources. The axis orientation and the encoder nonlinearity have been automatically calibrated with gyro drift compensation. An analytical error model of the algorithm (w.r.t. the joint control errors) has been rigorously derived and analyzed. The proposed algorithm is memory efficient with low computational cost. The effectiveness of the proposed calibration algorithm and its error model has been verified via both numerical and experimental studies.

Appendix

Lemma A.1: For a serial chain mechanism with n joints, axis orientation $\boldsymbol{\xi}_i$ and joint angle h_i for $i = 1, \cdots, n$, assume the k-th joint movement profile is shown in Fig. 8. It starts from zero and decreases to $h_k^{\min}$ at time $\tau_{\min}$, then increases until $h_k^{\max}$ at $\tau_{\max}$ passing zero at time t_{mid}, and finally decreases back to zero at time τ_{end}, the entire moving range joint is within $\left[-\frac{\pi}{2}, -\frac{\pi}{2}\right]$ (i.e., $-\frac{\pi}{2} \le h_k^{\min} \le 0 \le h_k^{\max} \le \frac{\pi}{2}$). Furthermore, the other joints are near zero with bound ϵ (i.e., $|h_i| \le \epsilon; i \ne k$). Then, the error Jacobian matrix defined as follow

$$\widetilde{\mathbf{J}}(\mathbf{h}) = \left[\begin{array}{cccccc} \widetilde{\mathbf{J}}_1, & \cdots & , \widetilde{\mathbf{J}}_k & , \cdots & , \widetilde{\mathbf{J}}_n \end{array} \right] \tag{29}$$

where

$$\widetilde{\mathbf{J}}_i = \begin{cases} e^{-\widehat{\boldsymbol{\xi}}_n h_n} \cdots e^{-\widehat{\boldsymbol{\xi}}_{i+1} h_{i+1}} \boldsymbol{\xi}_i & ; \quad i \ne k \\ e^{-\widehat{\boldsymbol{\xi}}_n h_n} \cdots e^{-\widehat{\boldsymbol{\xi}}_{k+1} h_{k+1}} \boldsymbol{\xi}_k - \boldsymbol{\xi}_k & ; \quad i = k \end{cases} \tag{30}$$

satisfies

$$\begin{aligned} \left\| \int_0^t \widetilde{\mathbf{J}}(\mathbf{h}) \dot{\mathbf{h}} d\tau \right\| \le (k-1) \left(2 \left| \sin(h_k^{\min}) \right| + 2 \left| \sin(h_k^{\max}) \right| \right. \\ \left. + 2 \left| 1 - \cos(h_k^{\min}) \right| + 2 \left| 1 - \cos(h_k^{\max}) \right| + 3 \right) \epsilon \\ + (n-k) \left(1 + 2|h_k^{\min}| + 2|h_k^{\max}| \right) \epsilon \end{aligned} \tag{31}$$

Proof. Since

$$\int_0^t \widetilde{\mathbf{J}}(\mathbf{h}) \dot{\mathbf{h}} d\tau = \sum_{i=1}^{n} \int_0^t \widetilde{\mathbf{J}}_i \dot{h}_i d\tau \tag{32}$$

we have the following three cases (In all cases, $\widetilde{\mathbf{J}}(\mathbf{h})$ is approximated by its first order Taylor expansion, which is justified by the non-moving joints being very small).

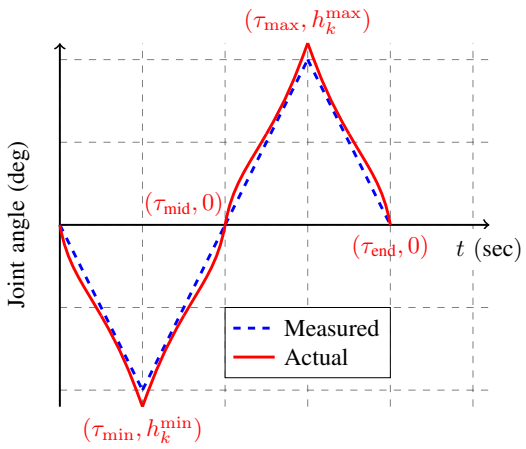


Fig. 8. Joint encoder readings and the actual joint angle during the calibration.

CASE I, $i > k$

Equation (30) implies

$$\widetilde{\mathbf{J}}_i = \boldsymbol{\xi}_i - \sum_{j=i+1}^{n} \widehat{\boldsymbol{\xi}}_j \boldsymbol{\xi}_i h_j \tag{33}$$

and therefore

$$\int_0^t \widetilde{\mathbf{J}}_i \dot{h}_i d\tau = \boldsymbol{\xi}_i h_i|_0^t - \sum_{j=i+1}^{n} \widehat{\boldsymbol{\xi}}_j \boldsymbol{\xi}_i \int_0^t h_j \dot{h}_i d\tau \tag{34}$$

Integrating $\int_0^t h_j \dot{h}_i d\tau$ by parts [38] yields

$$\int_0^t h_j \dot{h}_i d\tau = (h_j h_i)\,|_0^t - \int_0^t h_i \dot{h}_j d\tau \tag{35}$$

Assume that the non-rotating axis have the same control error distribution, i.e., $\int_0^t h_j \dot{h}_i d\tau = \int_0^t h_i \dot{h}_j d\tau; \forall i, j \neq k$, we have

$$\int_0^t h_j \dot{h}_i d\tau = \frac{1}{2} (h_j h_i)\,|_0^t \tag{36}$$

which is a high order term of small variables and can hence be ignored. As a result,

$$\int_0^t \widetilde{\mathbf{J}}_i \dot{h}_i d\tau = \boldsymbol{\xi}_i h_i|_0^t \tag{37}$$

The error bound is therefore given by

$$\left\| \int_0^t \widetilde{\mathbf{J}}_i \dot{h}_i d\tau \right\| \leq |h_i| \leq \epsilon \tag{38}$$

CASE II, $i = k$

$$\widetilde{\mathbf{J}}_k = - \sum_{j=k+1}^{n} \widehat{\boldsymbol{\xi}}_j \boldsymbol{\xi}_k h_j \tag{39}$$

and

$$\int_0^t \widetilde{\mathbf{J}}_k \dot{h}_k d\tau = - \sum_{j=k+1}^{n} \widehat{\boldsymbol{\xi}}_j \boldsymbol{\xi}_k \int_0^t h_j \dot{h}_k d\tau \tag{40}$$

As shown in Fig. 8, h_k first decreases until $h_k^{\min}$ at $\tau_{\min}$, then increases until $h_k^{\max}$ at $\tau_{\max}$, and finally decreases back to zero. In each time interval, $\dot{h}_k$ does not change its sign. Then with the mean value theorem [39], we have

$$\int_0^t h_j \dot{h}_k d\tau = h_j(\gamma_1) h_k|_0^{\tau_{\min}} + h_j(\gamma_2) h_k|_{\tau_{\min}}^{\tau_{\max}} + h_j(\gamma_2) h_k|_{\tau_{\max}}^{t} \tag{41}$$

where γ_1 is some time in between $[0, \tau_{\min}]$, γ_2 is in between $[\tau_{\min}, \tau_{\max}]$ and γ_3 is in between $[\tau_{\max}, t]$. As can be seen, we simplify the notation $h(\theta(t))$ used in the main text as $h(t)$. Equation (41) implies that

$$\int_0^t \widetilde{\mathbf{J}}_k \dot{h}_k d\tau = - \sum_{j=k+1}^{n} \widehat{\boldsymbol{\xi}}_j \boldsymbol{\xi}_k \left(h_j(\gamma_1) h_k|_0^{\tau_{\min}} + h_j(\gamma_2) h_k|_{\tau_{\min}}^{\tau_{\max}} + h_j(\gamma_3) h_k|_{\tau_{\max}}^{t} \right) \tag{42}$$

The error bound assuming $h_k(0) = 0; \forall i$ is therefore

$$\left\| \int_0^t \widetilde{\mathbf{J}}_k \dot{h}_k d\tau \right\| \leq \sum_{j=k+1}^{n} 2 \left(|h_k^{\min}| + |h_k^{\max}| \right) \epsilon \tag{43}$$

CASE III, $i < k$

$$\widetilde{\mathbf{J}}_i = e^{-\widehat{\boldsymbol{\xi}}_k h_k} \boldsymbol{\xi}_i - \sum_{j=k+1}^{n} \widehat{\boldsymbol{\xi}}_j e^{-\widehat{\boldsymbol{\xi}}_k h_k} \boldsymbol{\xi}_i h_j - \sum_{j=i+1}^{k-1} e^{-\widehat{\boldsymbol{\xi}}_k h_k} \widehat{\boldsymbol{\xi}}_j \boldsymbol{\xi}_i h_j \tag{44}$$

and

$$\int_0^t \widetilde{\mathbf{J}}_i \dot{h}_i d\tau = \int_0^t e^{-\widehat{\boldsymbol{\xi}}_k h_k} \boldsymbol{\xi}_i \dot{h}_i d\tau - \sum_{j=k+1}^{n} \int_0^t \widehat{\boldsymbol{\xi}}_j e^{-\widehat{\boldsymbol{\xi}}_k h_k} \boldsymbol{\xi}_i h_j \dot{h}_i d\tau - \sum_{j=i+1}^{k-1} \int_0^t e^{-\widehat{\boldsymbol{\xi}}_k h_k} \widehat{\boldsymbol{\xi}}_j \boldsymbol{\xi}_i h_j \dot{h}_i d\tau \tag{45}$$

Using the Rodrigue's formula [36], the first item in (45) can be rewritten as

$$\int_0^t e^{-\widehat{\boldsymbol{\xi}}_k h_k} \boldsymbol{\xi}_i \dot{h}_i d\tau = \boldsymbol{\xi}_k \boldsymbol{\xi}_k^T \boldsymbol{\xi}_i \int_0^t \dot{h}_i d\tau - \widehat{\boldsymbol{\xi}}_k \boldsymbol{\xi}_i \int_0^t \sin(h_k) \dot{h}_i d\tau - \widehat{\boldsymbol{\xi}}_k^2 \boldsymbol{\xi}_i \int_0^t \cos(h_k) \dot{h}_i d\tau \tag{46}$$

where

$$\int_0^t \sin(h_k)\,\dot{h}_i d\tau = (\sin(h_k)\,h_i)\,|_0^t - \int_0^t \cos(h_k)\,\dot{h}_k h_i d\tau \quad (47a)$$

$$\int_0^t \cos(h_k)\,\dot{h}_i d\tau = (\cos(h_k)\,h_i)\,|_0^t + \int_0^t \sin(h_k)\,\dot{h}_k h_i d\tau \quad (47b)$$

To estimate the error bound of (47a), we view $\cos(h_k)\,\dot{h}_k h_i$ as the multiplication of two separate functions: h_i and $\cos(h_k)\,\dot{h}_k$. The joint movement profile given in Fig. 8 shows that $\cos(h_k)\,\dot{h}_k$ does not change its sign in time intervals $[0, \tau_{\min}]$, $[\tau_{\min}, \tau_{\max}]$ and $[\tau_{\max}, \tau_{\text{end}}]$. With the mean value theorem [39], we have

$$\int_0^t \cos(h_k)\,\dot{h}_k h_i d\tau = h_i(\gamma_1)\sin(h_k)|_0^{h_k^{\min}} + h_i(\gamma_2)\sin(h_k)|_{h_k^{\min}}^{h_k^{\max}} + h_i(\gamma_3)\sin(h_k)|_{h_k^{\max}}^{h_k(t)} \quad (48)$$

where γ_1 is some time in between $[0, \tau_{\min}]$, γ_2 is in between $[\tau_{\min}, \tau_{\max}]$ and γ_3 is in between $[\tau_{\max}, t]$.

Similarly $\sin(h_k)\,\dot{h}_k$ does not change its sign in time intervals $[0, \tau_{\min}]$, $[\tau_{\min}, \tau_{\text{mid}}]$, $[\tau_{\text{mid}}, \tau_{\max}]$ and $[\tau_{\max}, \tau_{\text{end}}]$. Therefore,

$$\int_0^t \sin(h_k)\,\dot{h}_k h_i d\tau = -h_i(\gamma_1)\cos(h_k)|_0^{h_k^{\min}} - h_i(\gamma_2)\cos(h_k)|_{h_k^{\min}}^0 - h_i(\gamma_3)\cos(h_k)|_0^{h_k^{\max}} - h_i(\gamma_4)\cos(h_k)|_{h_k^{\max}}^{h_k(t)} \quad (49)$$

where $\gamma_1, \gamma_2, \gamma_3$ and γ_4 are respectively in the interval of $[0, \tau_{\min}]$, $[\tau_{\min}, \tau_{\text{mid}}]$, $[\tau_{\text{mid}}, \tau_{\max}]$ and $[\tau_{\max}, \tau_{\text{end}}]$.

Substituting equations (48) and (49) into (47) yields

$$\left|\int_0^t \sin(h_k)\,\dot{h}_i d\tau\right| \le \epsilon\left(1 + 2\left|\sin(h_k^{\min})\right| + 2\left|\sin(h_k^{\max})\right|\right) \quad (50a)$$

$$\left|\int_0^t \cos(h_k)\,\dot{h}_i d\tau\right| \le \epsilon\left(1 + 2\left|1 - \cos(h_k^{\min})\right| + 2\left|1 - \cos(h_k^{\max})\right|\right) \quad (50b)$$

Substituting (50) into (46) yields the error bound for the first item in (45)

$$\left\|\int_0^t e^{-\widehat{\boldsymbol{\xi}}_k h_k}\boldsymbol{\xi}_i \dot{h}_i d\tau\right\| \le \epsilon\left(2\left|\sin(h_k^{\min})\right| + 2\left|\sin(h_k^{\max})\right| + 2\left|1 - \cos(h_k^{\min})\right| + 2\left|1 - \cos(h_k^{\max})\right| + 3\right) \quad (51)$$

The second item in (45) can be rewritten as

$$\int_0^t \widehat{\boldsymbol{\xi}}_j e^{-\widehat{\boldsymbol{\xi}}_k h_k}\boldsymbol{\xi}_i h_j \dot{h}_i d\tau = \widehat{\boldsymbol{\xi}}_j\boldsymbol{\xi}_k\boldsymbol{\xi}_k^T\boldsymbol{\xi}_i\int_0^t h_j\dot{h}_i d\tau - \widehat{\boldsymbol{\xi}}_j\widehat{\boldsymbol{\xi}}_k\boldsymbol{\xi}_i\int_0^t \sin(h_k)\,h_j\dot{h}_i d\tau - \widehat{\boldsymbol{\xi}}_j\widehat{\boldsymbol{\xi}}_k^2\boldsymbol{\xi}_i\int_0^t \cos(h_k)\,h_j\dot{h}_i d\tau \quad (52)$$

where

$$\int_0^t \sin(h_k)\,h_j\dot{h}_i d\tau = (\sin(h_k)\,h_j h_i)\,|_0^t - \int_0^t \cos(h_k)\,h_j\dot{h}_k h_i d\tau - \int_0^t \sin(h_k)\,\dot{h}_j h_i d\tau \quad (53)$$

$$\int_0^t \cos(h_k)\,h_j\dot{h}_i d\tau = (\cos(h_k)\,h_j h_i)\,|_0^t + \int_0^t \sin(h_k)\,h_j\dot{h}_k h_i d\tau - \int_0^t \cos(h_k)\,\dot{h}_j h_i d\tau \quad (54)$$

Similar to (35), as $h_k, h_i, h_j, i \neq j \neq k$ are independent, we assume

$$\int_0^t \sin(h_k)\,h_j\dot{h}_i d\tau = \int_0^t \sin(h_k)\,\dot{h}_j h_i d\tau \quad (55a)$$

$$\int_0^t \cos(h_k)\,h_j\dot{h}_i d\tau = \int_0^t \cos(h_k)\,\dot{h}_j h_i d\tau \quad (55b)$$

Then,

$$\int_0^t \sin(h_k)\,h_j\dot{h}_i d\tau = \frac{1}{2}\sin(h_k)\,h_j h_i|_0^t - \frac{1}{2}\int_0^t \cos(h_k)\,h_j\dot{h}_k h_i d\tau \quad (56a)$$

$$\int_0^t \cos(h_k)\,h_j\dot{h}_i d\tau = \frac{1}{2}\cos(h_k)\,h_j h_i|_0^t + \frac{1}{2}\int_0^t \sin(h_k)\,h_j\dot{h}_k h_i d\tau \quad (56b)$$

Similar to the case II, the second term in (56) can separate to $h_j h_i$ and $\cos(h_k)\dot{h}_k$ or $\sin(h_k)\dot{h}_k$, which lead to terms in order of $o(h_j h_i)$. That is, (52) contains only higher oder terms (i.e., $o(h_j h_i)$) and its effect can be neglected. Similar derivation implies that the third term in (45) also vanishes. As a consequence, for the third case where $i < k$, the total error in (45) is dominated by its first term, i.e.,

$$\left\|\int_0^t \widetilde{\mathbf{J}}_i \dot{h}_i d\tau\right\| \le \left(2\left|\sin(h_k^{\min})\right| + 2\left|\sin(h_k^{\max})\right| + 2\left|1 - \cos(h_k^{\min})\right| + 2\left|1 - \cos(h_k^{\max})\right| + 3\right)\epsilon \quad (57)$$

Finally, from (32) we have

$$\left\|\int_0^t \widetilde{\mathbf{J}}(\mathbf{h})\dot{\mathbf{h}} d\tau\right\| = \left\|\sum_{i=1}^n \int_0^t \widetilde{\mathbf{J}}_i \dot{h}_i d\tau\right\| \le \sum_{i=1}^n \left\|\int_0^t \widetilde{\mathbf{J}}_i \dot{h}_i d\tau\right\| \quad (58)$$

Substituting (38), (43) and (57) into (58) yields

$$\left\|\int_0^t \widetilde{\mathbf{J}}(\mathbf{h})\dot{\mathbf{h}} d\tau\right\| \le (k-1)\left(2\left|\sin(h_k^{\min})\right| + 2\left|\sin(h_k^{\max})\right| + 2\left|1 - \cos(h_k^{\min})\right| + 2\left|1 - \cos(h_k^{\max})\right| + 3\right)\epsilon + (n-k)\left(1 + 2|h_k^{\min}| + 2|h_k^{\max}|\right)\epsilon \quad (59)$$

■

REFERENCES

[1] J. Goldman, "DJI mavic pro review," available at https://www.cnet.com/reviews/dji-mavic-pro-review/, November 2016.

[2] Global Flight, available at https://www.globe-flight.de/GF-Repaircenter-DJI-Mavic-Pro-Gimbal-Camera.

[3] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE robotics & automation magazine*, vol. 19, no. 3, pp. 20–32, 2012.

[4] R. Mahony, T. Hamel, and J.-M. Pflimlin, "Nonlinear complementary filters on the special orthogonal group," *IEEE Transactions on automatic control*, vol. 53, no. 5, pp. 1203–1218, 2008.

[5] G. Lu and F. Zhang, "Imu-based attitude estimation in the presence of narrow-band noise," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 2, pp. 841–852, 2019.

[6] A. Rue, "Stabilization of precision electrooptical pointing and tracking systems," *IEEE Transactions on Aerospace and Electronic Systems*, no. 5, pp. 805–819, 1969.

[7] A. K. Rue, "Calibration of precision gimbaled pointing systems," *IEEE Transactions on Aerospace and Electronic Systems*, no. 5, pp. 697–706, 1970.

[8] R. Pio, "Euler angle transformations," *IEEE Transactions on Automatic Control*, vol. 11, no. 4, pp. 707–715, 1966.

[9] B. Wang, Q. Ren, Z. Deng, and M. Fu, "A self-calibration method for nonorthogonal angles between gimbals of rotational inertial navigation system," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 4, pp. 2353–2362, 2015.

[10] C. Li, S. Zhang, and Y. Cao, "One new onboard calibration scheme for gimbaled imu," *Measurement*, vol. 46, no. 8, pp. 2359–2375, 2013.

[11] P. Gao, K. Li, L. Wang, and J. Gao, "A self-calibration method for non-orthogonal angles of gimbals in tri-axis rotational inertial navigation system," *IEEE sensors journal*, vol. 16, no. 24, pp. 8998–9005, 2016.

[12] R. P. Paul, *Robot manipulators: mathematics, programming, and control: the computer control of robot manipulators*. Richard Paul, 1981.

[13] W. Veitschegger and C.-H. Wu, "Robot accuracy analysis based on kinematics," *IEEE Journal on Robotics and Automation*, vol. 2, no. 3, pp. 171–179, 1986.

[14] S. A. Hayati, "Robot arm geometric link parameter estimation," in *Decision and Control, 1983. The 22nd IEEE Conference on*. IEEE, 1983, pp. 1477–1483.

[15] H. Zhuang, Z. S. Roth, and F. Hamano, "A complete and parametrically continuous kinematic model for robot manipulators," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 4, pp. 451–463, 1992.

[16] H. W. Stone, *Kinematic modeling, identification, and control of robotic manipulators*. Springer Science & Business Media, 2012, vol. 29.

[17] R. W. Brockett, "Robotic manipulators and the product of exponentials formula," in *Mathematical theory of networks and systems*. Springer, 1984, pp. 120–129.

[18] K. Okamura and F. C. Park, "Kinematic calibration using the product of exponentials formula," *Robotica*, vol. 14, no. 04, pp. 415–421, 1996.

[19] Y. Lou, T. Chen, Y. Wu, Z. Li, and S. Jiang, "Improved and modified geometric formulation of POE based kinematic calibration of serial robots," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2009, pp. 5261–5266.

[20] I.-M. Chen, G. Yang, C. T. Tan, and S. H. Yeo, "Local POE model for robot kinematic calibration," *Mechanism and Machine Theory*, vol. 36, no. 11, pp. 1215–1239, 2001.

[21] C. Li, Y. Wu, H. Löwe, and Z. Li, "POE-Based robot kinematic calibration using axis configuration space and the adjoint error model," *IEEE Transactions on Robotics*, vol. 32, no. 5, pp. 1264–1279, 2016.

[22] X. Yang, L. Wu, J. Li, and K. Chen, "A minimal kinematic model for serial robot calibration using POE formula," *Robotics and Computer-Integrated Manufacturing*, vol. 30, no. 3, pp. 326–334, 2014.

[23] T. Messay, R. Ordóñez, and E. Marcil, "Computationally efficient and robust kinematic calibration methodologies and their application to industrial robots," *Robotics and Computer-Integrated Manufacturing*, vol. 37, pp. 33–48, 2016.

[24] A. A. Hayat, R. A. Boby, and S. K. Saha, "A geometric approach for kinematic identification of an industrial robot using a monocular camera," *Robotics and Computer-Integrated Manufacturing*, vol. 57, pp. 329–346, 2019.

[25] Z. Jiang, W. Zhou, H. Li, Y. Mo, W. Ni, and Q. Huang, "A new kind of accurate calibration method for robotic kinematic parameters based on the extended kalman and particle filter algorithm," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 4, pp. 3337–3345, 2017.

[26] M. Gaudreault, A. Joubair, and I. Bonev, "Self-calibration of an industrial robot using a novel affordable 3d measuring device," *Sensors*, vol. 18, no. 10, p. 3380, 2018.

[27] C.-H. Rim, C.-M. Rim, J.-G. Kim, G. Chen, and J.-S. Pak, "A calibration method of portable coordinate measuring arms by using artifacts," *MAPAN*, vol. 34, no. 1, pp. 1–11, 2019.

[28] Y. Lin, H. Zhao, C. Ye, and H. Ding, "A computationally efficient and robust kinematic calibration model for industrial robots with kinematic parallelogram," in *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2017, pp. 1334–1339.

[29] Y. Guo, B. Song, X. Tang, X. Zhou, Y. Xie, and J. Jin, "Calibration for kinematic parameters of industrial robot by a laser displacement sensor," in *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*. IEEE, 2018, pp. 1829–1833.

[30] T. Sun, B. Lian, S. Yang, and Y. Song, "Kinematic calibration of serial and parallel robots based on finite and instantaneous screw theory," *IEEE Transactions on Robotics*, 2020.

[31] T. Messay-Kebede, G. Sutton, and O. Djaneye-Boundjou, "Geometry based self kinematic calibration method for industrial robots," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1–9.

[32] D. J. Bennett and J. M. Hollerbach, "Autonomous calibration of single-loop closed kinematic chains formed by manipulators with passive endpoint constraints," *IEEE transactions on Robotics and Automation*, vol. 7, no. 5, pp. 597–606, 1991.

[33] A. Filion, A. Joubair, A. S. Tahan, and I. A. Bonev, "Robot calibration using a portable photogrammetry system," *Robotics and Computer-Integrated Manufacturing*, vol. 49, pp. 77–87, 2018.

[34] F. Zhang, "Simultaneous self-calibration of nonorthogonality and nonlinearity of cost-effective multiaxis inertially stabilized gimbal systems," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 132–139, 2018.

[35] H. Zhuang and Z. S. Roth, "Modeling gimbal axis misalignments and mirror center offset in a single-beam laser tracking measurement system," *The International Journal of Robotics Research*, vol. 14, no. 3, pp. 211–224, 1995.

[36] R. M. Murray, Z. Li, S. S. Sastry, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[37] F. L. Markley and J. L. Crassidis, *Fundamentals of spacecraft attitude determination and control*. Springer, 2014, vol. 33.

[38] W. Rudin *et al.*, *Principles of mathematical analysis*. McGraw-hill New York, 1976, vol. 3, no. 4.2.

[39] M. Comenetz, *Calculus: the elements*. World Scientific Publishing Company, 2002.

Fu Zhang received the B.E. in Automation, in 2011, from the University of Science and Technology of China (USTC), China, and the Ph.D. degree in Mechanical Engineering, in 2015, from University of California, Berkeley, USA. He joined the University of Hong Kong, in 2018, where he is currently an assistant professor in Mechanical Engineering. His research interests include mechatronics and robotics, with focus on UAV design, control, and lidar-based navigation.

Minghui Zheng received the B.E. and M.E. degrees, in 2008 and 2011 respectively, from Beihang University, Beijing, China, and the Ph.D. degree in Mechanical Engineering, in 2017, from University of California, Berkeley, USA. She joined University at Buffalo, in 2017, where she is currently an assistant professor in Mechanical and Aerospace Engineering. Her research interests include advanced learning, estimation, and control with applications to high-precision and robotic systems.