

Direct Visual-Inertial Ego-Motion Estimation via Iterated Extended Kalman Filter

Shangkun Zhong and Pakpong Chirarattananon

Abstract—This letter proposes a reactive navigation strategy for recovering the altitude, translational velocity and orientation of Micro Aerial Vehicles. The main contribution lies in the direct and tight fusion of Inertial Measurement Unit (IMU) measurements with monocular feedback under an assumption of a single planar scene. An Iterated Extended Kalman Filter (IEKF) scheme is employed. The state prediction makes use of IMU readings while the state update relies directly on photometric feedback as measurements. Unlike feature-based methods, the photometric difference for the innovation term renders an inherent and robust data association process in a single step. The proposed approach is validated using real-world datasets. The results show that the proposed method offers better robustness, accuracy, and efficiency than a feature-based approach. Further investigation suggests that the accuracy of the flight velocity estimates from the proposed approach is comparable to those of two state-of-the-art Visual Inertial Systems (VINS) while the proposed framework is $\approx 15 - 30$ times faster thanks to the omission of reconstruction and mapping.

Index Terms—Aerial systems; perception and autonomy, sensor fusion.

I. INTRODUCTION

EFFICIENT and robust motion estimation plays a vital role in the operation of autonomous aerial robots. In recent years, several Visual-Inertial Systems have emerged as a framework for simultaneously recovering camera's motion and 3D map points by complementing visual sensors with IMU measurements. The visual-inertial motion estimation is one of the most intensively researched areas thanks to its accuracy, scalability and low cost. VINS, either optimization-based [1], [2], [3] or EKF-based [4], [5], [6], are capable of providing precise state estimation as environmental map points and camera poses are incrementally refined over a prolonged period. However, drawbacks of VINS exist. The refinement of a large number of poses and landmarks is of high computational complexity. While the sparse structure of the normal equation in the bundle adjustment and the incremental technique have been exploited to reduce the computational load [7], they are still unsuitable for real-time application on small robots with limited computational power. In addition, the robustness of VINS is influenced by the ability to continuously track features over a long period. This brings about

Manuscript received: September, 8, 2019; Revised December, 11, 2019; Accepted January, 2, 2020.

This paper was recommended for publication by Editor Eric Marchand upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the Research Grants Council of the Hong Kong Special Administrative Region of China (grant number CityU-11215117).

The authors are with the Department of Biomedical Engineering, City University of Hong Kong, Hong Kong SAR, China (emails: shanzhong4-c@my.cityu.edu.hk, pakpong.c@cityu.edu.hk).

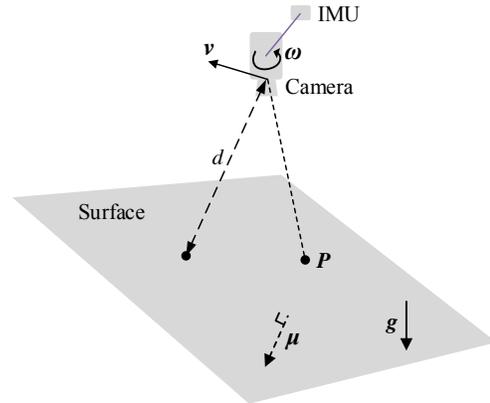


Fig. 1. Diagram of an IMU-camera rig movement. The moving IMU-camera setup observes a single non-horizontal plane. The linear velocity v , angular velocity ω , point P on the surface, the unit normal vector μ and the normalized gravity vector g are expressed in the camera frame.

some susceptibility to rapid motion, low-textured scenes, and varying light conditions.

Another family of motion estimation methods is the reactive navigation. As a less demanding approach, reactive navigation only relies on the processing of most recent frames of images and sensory data. Most prevalently, it employs optical flow from the visual sensor to track features between consecutive frames [8]. In contrast to the map-based navigation, the absence of landmarks' estimation significantly reduces the computational burden. Besides, optical flow-based methods offer more robust solutions as they are not required to maintain prolonged feature tracks.

There have been several developments related to optical flow-based navigation. Izzo *et al.* presented a safe landing strategy using ventral optical flow and time-to-contact [9]. Nevertheless, similar to VINS, uses of a monocular camera alone are unable to provide the metric scale. For this reason, Ho *et al.* proposed a distance and velocity estimator by taking into account the control inputs (instead of acceleration) to recover the metric scale [10]. In another example, Grabe *et al.* incorporated onboard IMU data to recover the linear velocity and orthogonal distance to planar scenes by the formulation of a nonlinear observer under a single plane assumption [11]. In the implementation, the plane's normal was assumed aligned with the gravity vector that is absolutely determined rather than estimated by the IMU measurements. To address the shortcoming, Hua *et al.* presented a nonlinear observer to estimate the depth, velocity, and gravity direction using the horizontal plane assumption [12]. The observer is unable to

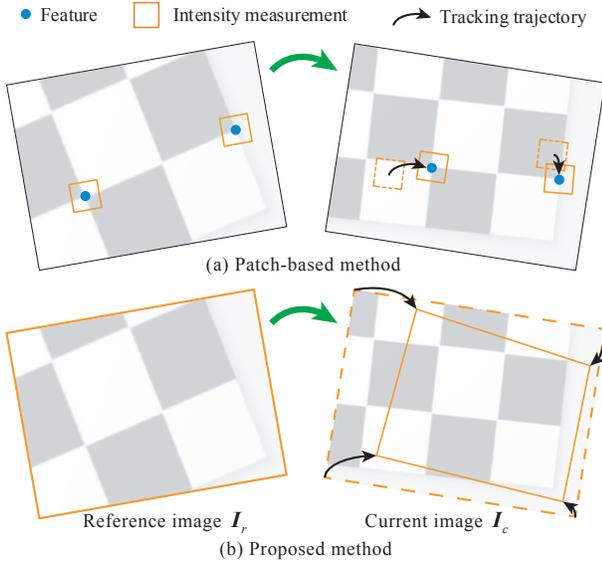


Fig. 2. A sketch comparing the measurement model of the patch-based approach [16] with the proposed method: (a) the patch-based method in [16] and (b) the proposed method. In [16], an intensity patch around each image feature in the reference frame I_r is warped into the current frame I_c according to the feature's depth and relative pose between two frames provided by the IMU prediction. The difference between the warped patch and the actual measurement constitutes the innovation term to update the state vector. The black arrows denote the trajectory during the iterated update. In contrast, the proposed method aligns the entire reference image I_r to the current image I_c using the homography model under the single plane assumption. The image deformation is predicted by the camera's motion through the IMU integration.

handle inclined planes.

The aforementioned map-based VINS and reactive navigation methods rely on feature extraction and association processes such as the well-established Lucas-Kanade (LK) tracker [8] to provide visual information. This feature identification and tracking process is relatively computationally demanding and may account for up to 99% of the total processing time in reactive navigation as found in [13]. Alternatively, the *direct* or *featureless* method, which eliminates the feature detection and tracking process, has been proposed to further reduce the computational complexity. Using image intensity, direct implementations also enhance the robustness against motion blur or in scenes with little texture. Nevertheless, for map-based methods, intensive computation is required for the generation of dense depth-map [14]. This issue is further remedied by the integration of feature tracking with patches of photometric feedback as a semi-direct approach [15], [16].

This paper presents a novel direct ego-motion estimation method for reactive navigation. Unlike previous direct methods for reactive navigation [17], [18], the proposed Iterated Extended Kalman Filter (IEKF) scheme efficiently estimates the inverse altitude, flight velocity, gravity direction, and plane's normal from photometric feedback in a single step. To achieve this, the single-plane assumption is employed. This is an attractive compromise when landmarks and the corresponding depth map is not considered. The assumption, also present in [11], [12], [17], [18], radically simplifies the computation, eliminating the preference to consider image patches to reduce the computational complexity as found in recent map-based strategies [15], [16]. Furthermore, the proposed approach has

no restriction on the motion pattern of the camera or the plane's inclination as present in [12], [17], [18].

The proposed framework takes motivation from the previous indirect reactive navigation method [11] and the semi-direct use of photometric feedback through IEKF [16]. That is, the photometric error from an entire image is directly integrated into the IMU measurements for the ego-motion estimation via the IEKF framework under the assumption of a single planar scene. As illustrated in Fig. 2, the proposed method differs from the work [16] owing to the single-plane assumption. The simplification means there exists only a low-dimension state vector associated with the image measurement model and the continuous homography equation. This makes the complex data association and mapping process unnecessary. To enhance the robustness, each pixel value on the image is integrated into Kalman update step instead of the multiple patches around the points of interest as in [16]. To deal with the large observation vector comprising of pixel intensities from the entire image, a Gauss-Newton Kalman gain [19] is used to substantially reduce the computation complexity. The direct implementation yields an implicit and robust tracking process. Meanwhile, the gravity direction and plane's normal are independently estimated. To do so, a compact parameterization of bearing vectors on manifolds is employed, avoiding the singularity. IMU biases are also estimated.

The downsides of the proposed method exist. Compared with the popular VINS, the proposed system is less versatile as it cannot be applied when multiple planes exist in the view. Without mapping, the 3D position is not formulated. Nevertheless, the efficiency, robustness, and precision of our system serve as a potential surrogate for computationally constrained platforms, such as small and insect-scale flying robots [20], [21].

The rest of this paper is structured as follows. Section II provides background on the continuous homography constraint. Section III presents the IEKF formulation for directly estimating the inverse altitude, ratio velocity, planar normal vector and gravity direction from photometric feedback. In Section IV, extensive flight experiments were performed to evaluate and benchmark the performance of the proposed method with respect to two state-of-the-art VINS [2], [16]. Lastly, conclusion and future directions are provided.

II. CONTINUOUS HOMOGRAPHY CONSTRAINT AND OPTICAL FLOW

In this section, we briefly recall the derivation of the continuous homography constraint. Throughout the manuscript, vectors and matrices are represented by bold letters. Vectors are expressed with respect to the camera's frame unless stated otherwise.

As illustrated in Fig. 1, suppose a point $P \in \mathbb{R}^3$ associated with a flat surface stationary in the inertia frame is observed by a moving camera with the linear velocity $v \in \mathbb{R}^3$ and angular velocity $\omega \in \mathbb{R}^3$. The motion of point P resulting from the camera movement is

$$\dot{P} = -[\omega]_{\times} P - v, \quad (1)$$

where $[\boldsymbol{\omega}]_{\times} \in \mathbb{R}^{3 \times 3}$ denotes the skew-symmetric matrix associated with $\boldsymbol{\omega}$. Let $\boldsymbol{\mu} \in \mathbb{S}^2$ denote a unit vector normal to the plane, not necessarily parallel to the gravity direction \mathbf{g} , and $d = \boldsymbol{\mu}^T \mathbf{P}$ denote the orthogonal distance from camera center to the plane as illustrated in Fig. 1. Eq. (1) becomes

$$\dot{\mathbf{P}} = -([\boldsymbol{\omega}]_{\times} + \frac{1}{d} \mathbf{v} \boldsymbol{\mu}^T) \mathbf{P}. \quad (2)$$

Let $\mathbf{p} = [u, v, 1]^T$ be the projection of point \mathbf{P} on the image plane and λ denote depth of the point \mathbf{P} in the camera frame ($\lambda > 0$), this yields

$$\mathbf{P} = \lambda \mathbf{M}^{-1} \mathbf{p}, \quad \dot{\mathbf{P}} = \dot{\lambda} \mathbf{M}^{-1} \mathbf{p} + \lambda \mathbf{M}^{-1} \dot{\mathbf{p}}, \quad (3)$$

where $\mathbf{M} \in \mathbb{R}^{3 \times 3}$ is the pinhole camera intrinsic matrix. Substituting Eq. (3) into (2) provides

$$\dot{\mathbf{p}} = -\mathbf{H} \mathbf{p} - \frac{\dot{\lambda}}{\lambda} \mathbf{p}, \quad (4)$$

where $\mathbf{H} \in \mathbb{R}^{3 \times 3}$ is known as the continuous homography matrix relating the optical flow $\dot{\mathbf{p}}$ to its coordinates \mathbf{p} [22]:

$$\mathbf{H} = \mathbf{M}([\boldsymbol{\omega}]_{\times} + \frac{1}{d} \mathbf{v} \boldsymbol{\mu}^T) \mathbf{M}^{-1}. \quad (5)$$

In Eq. (4), since the third element of $\dot{\mathbf{p}}$ is always zero, we obtain

$$\dot{\lambda}/\lambda = -\mathbf{e}_z^T \mathbf{H} \mathbf{p}, \quad (6)$$

where $\mathbf{e}_z = [0, 0, 1]^T$. Substituting the result into Eq. (4) yields

$$\dot{\mathbf{p}} = -(\mathbf{1} - \mathbf{p} \mathbf{e}_z^T) \mathbf{H} \mathbf{p}, \quad (7)$$

where $\mathbf{1}$ is a 3×3 identity matrix. Eq. (7) relates the camera motion and orientation with respect to the plane to the optical flow $\dot{\mathbf{p}}$.

III. IEKF ESTIMATION FRAMEWORK

In this section, we present the IEKF framework for estimating i) the distance to a flat plane, ii) the camera's translational velocity, iii) the plane's normal vector, and iv) the gravity direction. To achieve this, the IMU measurements are employed for propagation of the state and covariance estimates, while the photometric feedback from the camera is directly used for the correction of the predicted state.

A. State Definition

The state vector consists of the following elements:

$$\mathbf{x} := (\alpha, \boldsymbol{\vartheta}, \boldsymbol{\mu}_s, \mathbf{g}_s, \mathbf{b}_a, \mathbf{b}_\omega), \quad (8)$$

where α is the inverse orthogonal distance ($\alpha = d^{-1}$) from the camera center to the surface. The inverse parameterization has been shown to produce superior accuracy in [16]. Similar to [18], $\boldsymbol{\vartheta} := \mathbf{v}/d \in \mathbb{R}^3$ is defined as the ratio of flight velocity to the distance. The unit normal vector $\boldsymbol{\mu}$ of the plane and the normalized gravity vector \mathbf{g} are represented as members of manifolds on \mathbb{S}^2 [16]. They can be obtained by rotating the basis vector \mathbf{e}_z via rotations $\boldsymbol{\mu}_s, \mathbf{g}_s \in SO(3)$, such as $\boldsymbol{\mu} = \boldsymbol{\mu}_s(\mathbf{e}_z)$ and $\mathbf{g} = \mathbf{g}_s(\mathbf{e}_z)$. Compared to other parametrization

method, such as azimuth and elevation, this implementation does not suffer from the singularity issue and it is relatively simple to derive their differentials. The separate treatment of $\boldsymbol{\mu}_s$ and \mathbf{g}_s allows the estimation to deal with non-horizontal ground. The terms \mathbf{b}_i 's represent IMU biases as defined below.

B. State Prediction

The state propagation begins with the discretization of the continuous dynamic model.

1) *State Dynamics*: The dynamics of the state is dependent on the specific acceleration $\hat{\mathbf{a}}$ and the angular rate $\hat{\boldsymbol{\omega}}$ of the camera frame. These quantities are related to the measurements from the accelerometer (\mathbf{a}_m) and gyroscope ($\boldsymbol{\omega}_m$). For simplicity, the IMU frame is assumed to be aligned with the camera's frame. In addition, the IMU readings are assumed to be corrupted by a bias \mathbf{b} and white noise \mathbf{w} such that

$$\mathbf{a}_m = \hat{\mathbf{a}} + \mathbf{b}_a + \mathbf{w}_a, \quad \boldsymbol{\omega}_m = \hat{\boldsymbol{\omega}} + \mathbf{b}_\omega + \mathbf{w}_\omega. \quad (9)$$

Consequently, the state dynamics ($\dot{\mathbf{x}}$) are:

$$\dot{\alpha} = \alpha \boldsymbol{\mu}^T \boldsymbol{\vartheta} + w_\alpha, \quad (10)$$

$$\dot{\boldsymbol{\vartheta}} = \alpha(\hat{\mathbf{a}} - g_0 \mathbf{g}) + (\boldsymbol{\mu}^T \boldsymbol{\vartheta} \mathbf{1} - [\hat{\boldsymbol{\omega}}]_{\times}) \boldsymbol{\vartheta} + \mathbf{w}_\vartheta, \quad (11)$$

$$\dot{\boldsymbol{\mu}}_s = \mathbf{N}(\boldsymbol{\mu}_s)^T \hat{\boldsymbol{\omega}} + \mathbf{w}_\mu, \quad (12)$$

$$\dot{\mathbf{g}}_s = \mathbf{N}(\mathbf{g}_s)^T \hat{\boldsymbol{\omega}} + \mathbf{w}_g, \quad (13)$$

$$\dot{\mathbf{b}}_a = \mathbf{w}_{b_a}, \quad \dot{\mathbf{b}}_\omega = \mathbf{w}_{b_\omega}, \quad (14)$$

where $g_0 = 9.8 \text{ ms}^{-2}$ is the gravitational acceleration. The terms \mathbf{w}_i 's are zero-mean Gaussian white noise. The operator $\mathbf{N}(\cdot)$ linearly projects a 3×1 unit vector into the tangent space of a unit vector in \mathbb{R}^2 such that

$$\mathbf{N}(\boldsymbol{\mu}_s) = [\boldsymbol{\mu}_s(\mathbf{e}_x), \boldsymbol{\mu}_s(\mathbf{e}_y)], \quad (15)$$

where $\mathbf{e}_x = [1, 0, 0]^T$ and $\mathbf{e}_y = [0, 1, 0]^T$ such that $\boldsymbol{\mu}_s(\mathbf{e}_i)$'s are the basis vectors of the coordinate system.

2) *Discretization*: The dynamics of the state described by the Eq. (10)-(14) are nonlinear in nature. To leverage the IEKF, they are discretized using the forward Euler method:

$$\mathbf{x}_k^- \approx \mathbf{x}_{k-1}^+ \boxplus \Delta T \dot{\mathbf{x}}_{k-1}^+. \quad (16)$$

where k denotes the time index at instant t_k and ΔT denotes the IMU sample time. \mathbf{x}_{k-1}^+ is an a-posteriori estimate at time t_{k-1} and \mathbf{x}_k^- an a-priori estimate at time t_k . The boxplus (or boxminus) operator in Eq. (16) behaves as a regular addition (or subtraction) in the Euclidean space. The exception is when it is applied to unit vectors defined on 2-manifolds (\mathbb{S}^2). Readers are referred to [16] for the detailed definition of these operators.

The propagation of the covariance matrix of the state uncertainty follows

$$\boldsymbol{\Sigma}_k^- = \mathbf{F}_{k-1} \boldsymbol{\Sigma}_{k-1}^+ \mathbf{F}_{k-1}^T + \mathbf{G}_{k-1} \mathbf{Q}_{k-1} \mathbf{G}_{k-1}^T, \quad (17)$$

with $\boldsymbol{\Sigma}_{k-1}^+$ denoting an a-posteriori covariance at time t_{k-1} and $\boldsymbol{\Sigma}_k^-$ an a-priori covariance at time t_k . \mathbf{F}_{k-1} and \mathbf{G}_{k-1} are the Jacobians of the propagated states with respect to the previous state \mathbf{x}_{k-1}^- and process noise \mathbf{w}_{k-1} . \mathbf{Q}_{k-1} is the covariance matrix of the additive process noise \mathbf{w}_i 's at t_{k-1} .

The state prediction is performed according to Eq. (16) and (17) at the rate determined by the frequency of the IMU measurements (ΔT^{-1}), independent of the observation or visual feedback.

C. Photometric Measurements

We directly use pixel intensities from the whole image as measurements for the state update. This constitutes one key contribution of our work.

1) *Image-based Measurement Model*: At time t_r , a point on the surface projected onto the reference image plane at \mathbf{p}_r has the corresponding pixel intensities $\mathbf{I}_r(\mathbf{p}_r)$ with $\mathbf{I} \in \mathbb{R}^{m \times n}$ denoting the 2D image domain. After a time period (δT), each spot displaces to a new location on the current image plane \mathbf{I}_c according to the current state (\mathbf{x}_k). This motion can be described using a homography projective transformation mapping $\mathcal{H}_k(\mathbf{p}_r|\mathbf{x}_k)$. The corresponding pixel intensity remains identical under the constant brightness assumption:

$$\mathbf{I}_r(\mathbf{p}_r) = \mathbf{I}_c(\mathcal{H}_k(\mathbf{p}_r|\mathbf{x}_k)), \quad (18)$$

where the mapping $\mathcal{H}_k(\mathbf{p}_r|\mathbf{x}_k)$ is derived from Eq. (7),

$$\mathcal{H}_k(\mathbf{p}_r|\mathbf{x}_k) \approx \mathbf{p}_r - \delta T(\mathbf{1} - \mathbf{p}_r \mathbf{e}_z^T) \mathbf{H}_k(\mathbf{x}_k) \mathbf{p}_r. \quad (19)$$

From here, we define an observation vector $\mathbf{z}_k \in \mathbb{R}^{mn}$ obtained by stacking the elements of $\mathbf{I}_c(\mathcal{H}_k(\mathbf{p}_r|\mathbf{x}_k))$ from the entire image. Let $\mathbf{h}(\mathbf{x}_k)$ be an observation model derived from Eq. (18) and (19). Subsequently, the measurement of pixel intensities over the entire image is modeled as

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \boldsymbol{\eta}_k, \quad (20)$$

where $\boldsymbol{\eta}_k$ is the observation noise assumed to be zero-mean Gaussian white noise with covariance \mathbf{R}_k . This means the entire image is used as feedback for the state \mathbf{x}_k .

2) *Iterated State Update*: The update step is executed once a new image is available. Instead of relying on identified image features [2] or using the sparse patch-based intensity measurements as in [16], pixel intensities from the entire image are used as a measurement vector as outlined by Eq. (20). In addition, the use of IEKF reduces the susceptibility to the inaccuracy of the initial estimate of the standard EKF.

The update step is designed to find a Kalman gain that provides an approximate maximum a-posteriori probability estimate. This is equivalent to finding the a-posteriori estimate that minimizes the cost function

$$\arg \min_{\mathbf{x}_k^+} \left\| \mathbf{x}_k^+ \boxminus \mathbf{x}_k^- \right\|_{\Sigma_k^-}^2 + \left\| \mathbf{z}_k - \mathbf{h}(\mathbf{x}_k^+) \right\|_{\mathbf{R}_k}^2. \quad (21)$$

Since the dynamic model and measurement model are nonlinear, Eq. (21) is solved recursively via the IEKF framework. That is, each update step contains several iterative steps (denoted by a subscript j). Starting from $j = 0$, the a-posteriori estimate of the state at the j^{th} iteration is

$$\mathbf{x}_{k,j+1}^+ = \mathbf{x}_{k,j}^+ \boxplus \Delta \mathbf{x}_{k,j}. \quad (22)$$

$$\begin{aligned} \Delta \mathbf{x}_{k,j} = & \mathbf{K}_{k,j} \left(\left(\mathbf{z}_k - \mathbf{h}(\mathbf{x}_{k,j}^+) \right) + \mathbf{S}_{k,j} \mathbf{L}_{k,j} \left(\mathbf{x}_{k,j}^+ \boxminus \mathbf{x}_k^- \right) \right) \\ & - \mathbf{L}_{k,j} \left(\mathbf{x}_{k,j}^+ \boxminus \mathbf{x}_k^- \right), \end{aligned} \quad (23)$$

where matrices $\mathbf{L}_{k,j}$ and $\mathbf{S}_{k,j}$ are Jacobians [16]:

$$\mathbf{L}_{k,j} = \frac{\partial \mathbf{x}_k^- \boxplus \Delta \mathbf{x}}{\partial \Delta \mathbf{x}} \left(\mathbf{x}_{k,j}^+ \boxminus \mathbf{x}_k^- \right), \quad \mathbf{S}_{k,j} = \frac{\partial \mathbf{h}(\mathbf{x}_{k,j}^+)}{\partial \mathbf{x}_{k,j}^+}. \quad (24)$$

Rather than using a regular Kalman gain for Eq. (23), we use the Gauss-Newton (GN) Kalman gain [19]:

$$\mathbf{K}_{k,j} = \left((\mathbf{L}_{k,j}^T \Sigma_k^- \mathbf{L}_{k,j})^{-1} + \mathbf{S}_{k,j}^T \mathbf{R}_k^{-1} \mathbf{S}_{k,j} \right)^{-1} \mathbf{S}_{k,j}^T \mathbf{R}_k^{-1}. \quad (25)$$

The use of GN gain dramatically improves the computational efficiency. The calculation of a standard Kalman gain is dominated by an inverse operation of an $mn \times mn$ matrix, which is overwhelmingly large for $m \times n$ image feedback. The inverse operation in Eq. (25) is performed on a square matrix of which the dimension is determined by the length of \mathbf{x} or 14×14 . The complexity of Eq. (25) is prevailed by the multiplication $\mathbf{S}_{k,j}$ or $O\{(mn)^2\}$ operations only.

Finally, the iteration is terminated when the 2-norm of $\Delta \mathbf{x}_{k,j}$ is below a certain threshold or the iteration reaches the maximum steps. The state covariance is updated only once with the Jacobians at the last u^{th} iteration step according to

$$\Sigma_k^+ = \Sigma_k^- - \mathbf{K}_{k,u} \mathbf{S}_{k,u} \mathbf{L}_{k,u}^T \Sigma_k^- \mathbf{L}_{k,u}. \quad (26)$$

Iterated updates effectively prevent the accumulation of errors and improve the convergence and accuracy, especially in the initialization with large uncertainty. On the other hand, the pre-defined termination conditions limit the iteration to a few steps, mitigating the extra computational burden.

IV. EXPERIMENTAL EVALUATION

This section presents the results from several experiments to illustrate and assess our approach with various real-world datasets in terms of accuracy and computational cost. Root Mean Squared Errors (RMSE) of the estimated states with respect to the ground-truth are used for evaluation. In section IV-B1, we first compare the proposed direct method with the traditional feature-based or LK method. Then a comparison between the proposed method and two state-of-the-art VINS is provided in section IV-B2. Lastly, the proposed method is tested when the robot flies over planes with different angles of inclination.

A. Experimental Setup

For the experiments, we collected real-world datasets with an IMU-camera setup (MYNT AI, MYNT EYE) mounted on an AscTec Hummingbird quadrotor (Ascending Technologies) as shown in Fig. 3(a). A motion tracking system (NaturalPoint, OptiTrack) was used to provide the ground-truth position and orientation, allowing the true state to be evaluated. For a horizontal surface, the true distance is the robot's altitude. The true velocity in the body frame is computed from the position and then transformed into the body frame.

The visual-inertial sensor contains an ICM 2060 IMU from InvenSense and a MT9V034 camera from ON Semiconductor, both of which operate under hardware synchronization. Both intrinsics and extrinsics between these two sensors were calibrated beforehand. The IMU provides the measurements of

TABLE I
COMPARISON OF THE ESTIMATION RESULTS FROM THE D-IEKF, D-EKF AND LK METHODS.

| Flight | Speed | Pattern | $\ v\ _a, \ v\ _m^1$ (cm/s) | Altitude RMSE (cm) | | | Velocity RMSE (cm/s) | | | Average time cost (ms) | | |
|--------|-------|---------|--------------------------------|--------------------|----------------|------|----------------------|------------|------------|------------------------|-------------|------|
| | | | | D-IEKF | D-EKF | LK | D-IEKF | D-EKF | LK | D-IEKF | D-EKF | LK |
| ① | Low | CKB | 24,83 | 2.2 | 2.3 | 4.9 | 2.0 | 2.0 | 1.4 | 1.36 | 1.33 | 6.97 |
| ② | | CKB | 21,113 | 3.7 | 3.7 | 7.3 | 1.2 | 1.2 | 1.8 | 1.35 | 1.34 | 7.01 |
| ③ | | CKB | 24,94 | 3.2 | 3.2 | 73.6 | 2.2 | 2.2 | 118.5 | 1.37 | 1.34 | 6.91 |
| ④ | | VEG | 21,64 | 2.9 | 3.6 | 5.5 | 1.4 | 1.4 | 1.4 | 1.38 | 1.33 | 7.11 |
| ⑤ | | VEG | 22,86 | 3.2 | 5.7 | 29.7 | 2.0 | 2.2 | 35.4 | 1.37 | 1.34 | 7.13 |
| ⑥ | High | VEG | 44,194 | 5.9 | 6.1 | 85.1 | 6.1 | 6.3 | 145.0 | 1.43 | 1.31 | 7.25 |
| ⑦ | | VEG | 57,275 | 5.8 | * ² | 75.3 | 7.0 | * | 146.2 | 1.44 | 0.63 | 7.22 |

¹ $\|v\|_a$ and $\|v\|_m$ are the root mean squared velocity and maximum velocity magnitude computed from the motion capture system used to describe the flight characteristics.

² The * symbol denotes a divergent estimate.

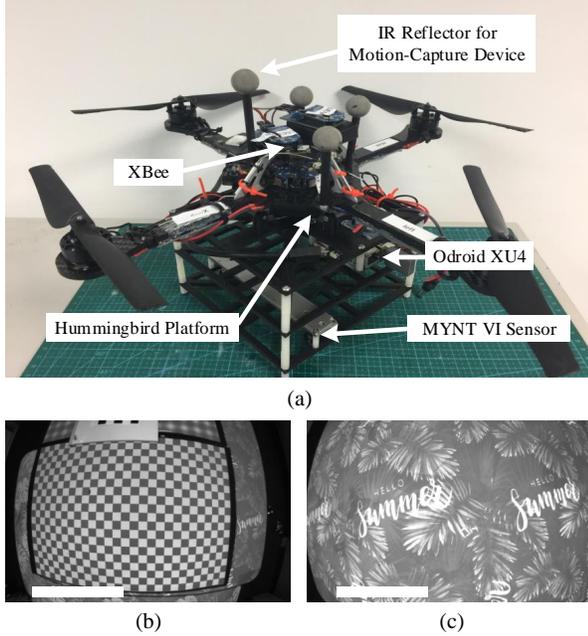


Fig. 3. (a) An AscTec Hummingbird quadrotor with a downward-facing MYNT VI sensor. (b), (c) two textures for experimental validation: Checkerboard (b) and Vegetation (c). The motion capture system is used for ground-truth measurements. Real-time control commands are transmitted from the ground station to the onboard controller via a pair of XBees. The white scale bars in (b) and (c) are 0.5 m and 0.3 m.

specific accelerations and angular rates at 500 Hz. To attenuate the disturbance from vibration, a low-pass filter was employed. The IMU data were then downsampled to 100 Hz for the state prediction ($\Delta T = 0.01$ s). Grayscale images of size 752×480 px were acquired at 30 frames per second. Both IMU measurements and images were recorded on the Odroid XU4 board and post-processed offline on a laptop with the Intel Core i5-8250U CPU at 1.6GHz. The offline implementation allows several estimation strategies to be compared using the same datasets. To verify the proposed estimation strategy, the algorithm was implemented in C++[†]. Consecutive image frames are taken as the reference I_r and current image I_c ($\delta T^{-1} = 30$ Hz). All estimates were obtained with the same set of parameters unless specified. Assuming the state and measurement noises are statistically uncorrelated from one another and time independent, Q_k and R_k become diagonal

and constant. The maximum iteration steps during the update stage (Eq. (21)) was set to three, the termination threshold of 2-norm of iteration change $\Delta x_{k,j}$ to 0.05, the initial inverse altitude α_0 to 10.0 m^{-1} , ratio velocity ϑ_0 to 0.0 s^{-1} . The initial normal vector μ_0 was chosen as $[0.2, -0.1, 0.97]^T$ to make the task more challenging and the initial gravity direction was set to $[0, 0, 1]^T$.

B. Flights over Horizontal Ground

For validation, we initially performed seven flights over two patterns on the horizontal ground and recorded the measurements. These patterns, Checkerboard (CKB) and Vegetation (VEG), shown in Fig. 3(b)-(c), were selected as they feature salient corners and edges. During each flight, the robot was remotely controlled to follow an arbitrary trajectory covering an approximate $0.8 \times 0.8 \times 0.8$ m volume for over 60 s. Among seven flights, five are low-speed flights with the RMS velocities of $\approx 0.2 \text{ ms}^{-1}$ and the other two are high-speed flights with the RMS velocities over 0.4 ms^{-1} . These two flight regimes were tested to inspect the performance of different estimation methods in different real-world scenarios. Outdoor flights are excluded due to the difficulty in obtaining ground-truth measurements and the single plane assumption may not hold in a complex landscape.

1) *Comparison of the Proposed Direct Method with the LK Method:* The use of the photometric difference between the consecutive frames as IEKF innovation term is one key feature of the proposed method. The featureless approach has the potential to be more robust as it is not susceptible to feature tracking errors. To verify this, the proposed direct method is compared with the traditional LK method. In the implementation of the estimation algorithms, for the proposed direct method (D-IEKF), original images were downsampled to 90×58 px. For the LK method, the pipeline is similar to [2]. That is, 50 Harris corners [23] were extracted from original images (752×480 px). These corners were then tracked by the pyramidal LK method over consecutive images with 20×20 patch size and three image levels. The innovation term in the proposed method is replaced with the difference between the predicted and measured feature coordinates. Our preliminary findings reveal that for the LK method, the update step is vulnerable to tracking outliers, resulting in occasional divergence. To resolve this, the objective function in Eq. (21) is robustified with the Huber loss [24] for the LK method. In addition to D-IEKF and LK, we also performed the estimation

[†] Available at <https://github.com/ris-lab/direct-vi-iekf/>

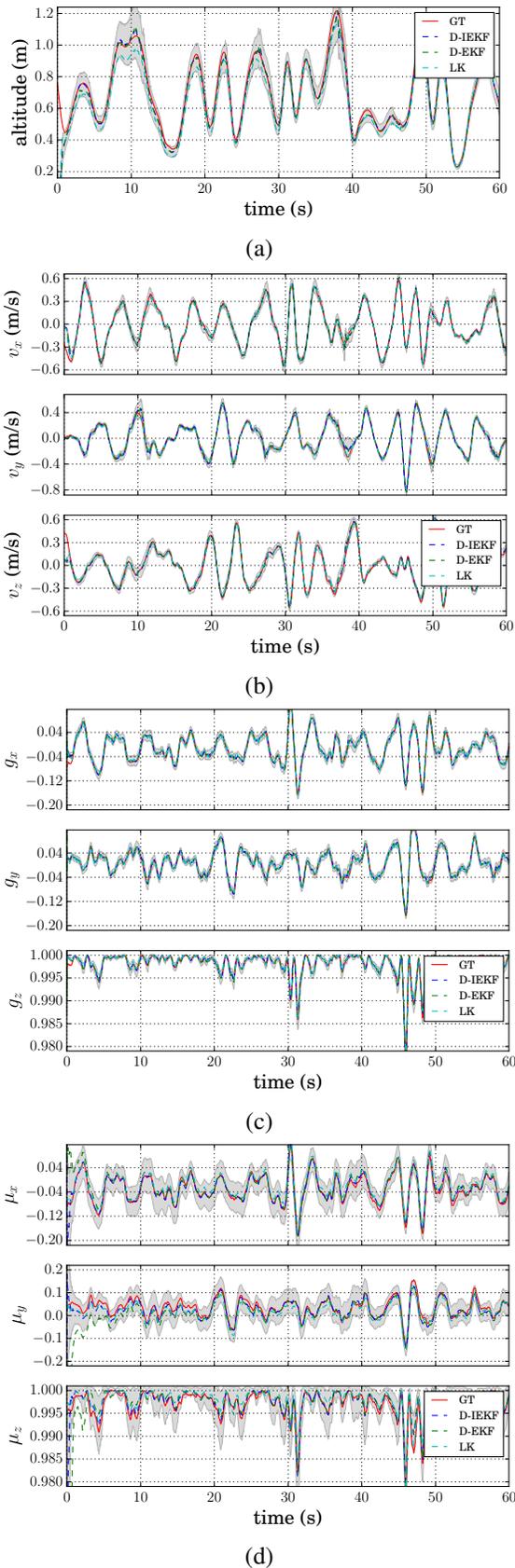


Fig. 4. Comparison of the estimates from the proposed and benchmark methods using dataset ①. The estimates of (a) altitude, (b) vVelocity (c) gravity vector and (d) normal vector from the four approaches are plotted against the ground-truth values (GT). Gray shaded areas indicate $2\text{-}\sigma$ bounds of the D-IEKF estimates.

using photometric feedback with a standard EKF (this is equivalent to setting the maximum iteration step of the IEKF to one), notated as D-EKF. For all cases, the weights between the prediction and the image measurements were tuned to obtain the best results for all methods and retained the same for all experiments.

For assessment, all estimation errors are computed after the estimates converge to the ground-truth, this corresponds to three seconds after the first image update is performed. Table I shows the RMSEs of the estimated altitude, linear velocity with respect to the ground-truth, and the average time consumption per frame from all three implementations. It can be seen that both direct methods produce lower RMSEs in the linear velocity and altitude than LK while they are approximately five times faster. It can be concluded that the direct methods outperform the LK method in terms of accuracy and efficiency. This is because, in many circumstances, the quality of the LK estimates suffers from the unreliability caused by incorrect feature correspondences in certain frames. The LK feature association fails to handle repetitive textures such as the CKB pattern, despite the use of Huber loss function, and subsequently corrupts the estimation. This issue could be further ameliorated with an additional outlier rejection strategy such as an application of the epipolar constraint between image correspondences [2]. On the other hand, in direct methods, the consistent homography projective constraint is imposed (Eq. (19)), yielding an inherent outlier rejection. As a result, by exploiting the single plane assumption, the proposed strategy is more robust than the LK method.

The results from D-EKF exhibit marginally larger RMSEs in the distance and linear velocity compared to that of D-IEKF, consistent with the outcomes in [16]. As anticipated, the deprivation of the iterated update defers the convergence of the estimates towards the ground-truth only at the beginning in most cases. The comparable average time cost between these two methods similarly indicates that multiple iterative steps occur almost exclusively at the inception phase of all sequences. After convergence, only one iterative step is required to meet the termination criterion. IEKF essentially accelerates the convergence at a slight increase in computational cost. In addition to the favorable speed-up of the convergence, the result from flight ⑦, in which the robot maneuvered at relatively high speed, highlights the exceptional robustness of D-IEKF. The high flight speed renders the LK method to be extremely inaccurate and causes the D-EKF method to diverge since a single update iteration was not sufficient for the estimation to reduce the initial photometric error between consecutive images and the prediction. The failure to properly find the maximum a posteriori probability estimate (Eq. (21)) leads to an accumulation of errors and the divergence of the estimates. This demonstrates that multiple iteration steps enhance the robustness compared to a single iterative step.

Fig. 4 depicts the estimation results from flight ① in detail. The estimated altitudes from all three approaches converge close to the ground truth at around $t = 3$ s (Fig. 4(a)). The results from D-IEKF and D-EKF are nearly identical after the convergence. Fig. 4(a)-(c) reveals that the estimates of the altitude, velocity and gravity direction from all methods are

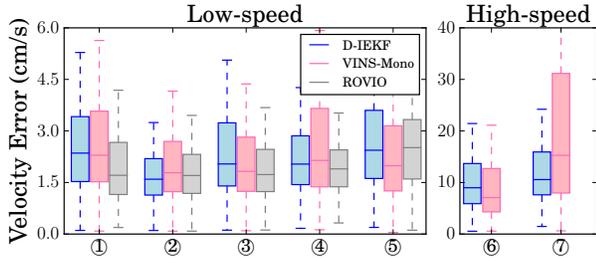


Fig. 5. Boxplots comparing D-IEKF, VINS-Mono, and ROVIO in terms of velocity errors. The left sub-figure shows the results from five low-speed flights and the right sub-figure corresponds to two high-speed flights. The increased flight speed deteriorates the estimation performance. ROVIO estimates failed to converge in the case of high-speed flights.

only slightly different whereas the estimates of the plane’s normal vector from D-IEKF and D-EKF in Fig. 4(d) display a noticeable distinction at the beginning. This corroborates the claim that the iterative update expedites the convergence. Furthermore, it can be observed that the uncertainties of the D-IEKF estimates, and likewise estimation errors, are more pronounced at the extreme points of the velocity plots (Fig. 4(b)). These points coincide with the periods where the camera’s acceleration approaches zero. This is consistent with the fact that the scale ambiguity of the monocular vision cannot be resolved in the absence of acceleration.

2) *Comparison of the Proposed Direct Method with the State-of-the-art VINS*: We further compare the D-IEKF method against two state-of-the-art VINS: VINS-Mono [2] and ROVIO [16] using their published C++ codes. Unlike the proposed estimator for reactive navigation, VINS-Mono and ROVIO are map-based VINS suitable for autonomous navigation. Since these map-based VINS do not assume the camera to be pointing toward a single flat terrain, the estimate of the distance to a flat terrain or flight altitude is not immediately available for comparison. Despite substantial differences in assumptions and computational complexity, both regimes provide the estimated flight velocity that can be directly compared. This serves as a surrogate measure for comparison of distance estimation owing to the tightly coupled dynamics of velocity and distance.

VINS-Mono is a variant of a visual-inertial SLAM system rather than a front-end. It features an accurate joint optimization of visual inertial information, loop closure, and map merging and reuse [2]. For comparison, the estimates of flight velocity from the sliding window estimator were logged out. In contrast to VINS-Mono, ROVIO is characterized as a robust and fast visual-inertial front-end. It leverages an IEKF framework by tightly integrating patch-based photometric feedback as its Kalman innovation term. For comparison with the D-IEKF estimates, we used the default ROVIO parameter configuration, which has been well-tuned to achieve a balanced trade-off between accuracy and efficiency. The number of tracked features per frame is set to 25 and the patch size to 6×6 . The second and third levels are employed for tracking the multiple level features.

Seven datasets collected from the previous section were used to obtain the velocity estimates from both ROVIO and

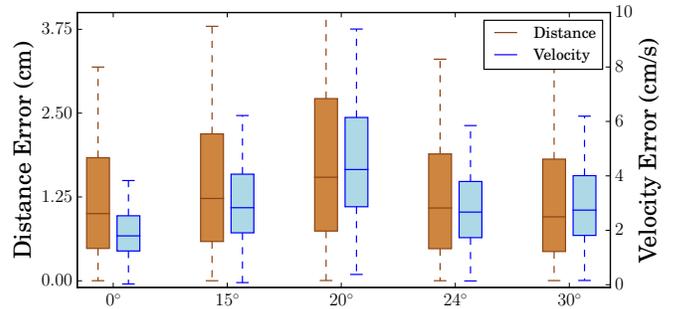


Fig. 6. Distance and velocity errors of the D-IEKF estimates from flights over planes with different angles of inclination.

VINS-Mono as outlined. Fig. 5 presents the velocity estimation errors using boxplots, depicting the medians and quartiles of the errors. According to the plot, the results demonstrate no overall significant distinction between the three methods in the case of five low-speed maneuvers. Nevertheless, for the two high-speed flights, ROVIO failed the initialization and subsequent tracking as a result. VINS-Mono, on the other hand, has a robust and complex initialization procedure that provides relatively accurate initial estimates. For D-IEKF, the adoption of entire images and iterated updates improve the robustness to deal with the high-speed flights. From the obtained results, it can be concluded that D-IEKF has comparable performance to that of two state-of-the-art VINS when it comes to the flight velocity estimation.

In terms of the efficiency, the time consumption per frame averaged from all sequences from all three methods are D-IEKF-1.42 ms, VINS-Mono-41.52 ms, and ROVIO-23.5 ms. Note that for VINS-Mono three threads operate in parallel and only the time cost of the sliding optimization is counted. While the proposed estimator is approximately 15-30 times faster than VINS-Mono and ROVIO, the exceptional computational efficiency is compromised by the lack of mapping and the requirement of a single observed flat surface.

C. Flights over Tilted Planes

Different from [11], [12], [18], our strategy to separately estimate the gravity direction and the plane’s normal allows the proposed method to relax the assumption that camera observes horizontal ground. In other words, it is applicable to flights above an inclined plane. To verify this, additional flight experiments were carried out over surfaces covered by the VEG texture with the angles of inclination up to 30° using identical flight configurations and estimation parameters to the experiments on horizontal ground.

Fig. 6 shows the errors of the estimated distance to the inclined planes and the translational velocity. The data belonging to flight ④ in Table I is employed as the flight above a 0° -inclined plane. The plot shows that the accuracy of the estimates is not visibly affected by the plane’s inclination. Only marginal variation is seen across different angles. A closer inspection is given in Fig. 7. The plot demonstrates the angle between the estimated normal vector and the vertical for different cases. The estimated angles evidently oscillate within

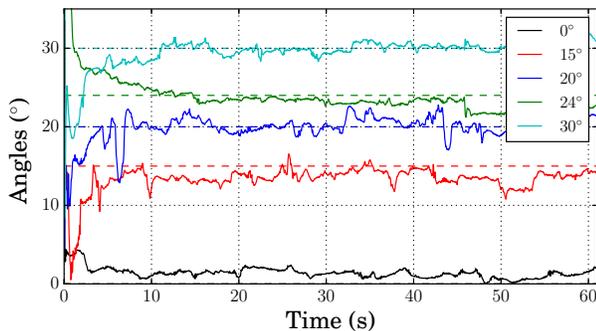


Fig. 7. The angle between the estimated normal vector and gravity vector on different incline planes. The dash lines are the ground-truth angles of the corresponding plane obtained from the motion capture system.

the vicinity of the true planes' inclination angles. The result from the 20° case exhibits the largest deviation among the four experiments. That is related to the deteriorated accuracy of the velocity and distance estimates shown in Fig. 6, consistent with the relationship predicted by Eq. (10) and (11).

V. CONCLUSION

In this paper, we have proposed a computationally efficient framework to estimate the inverse altitude, velocity and the surface's orientation for MAVs from a monocular vision and IMU measurements. The key contribution of our framework lies in the direct use of photometric feedback as the Kalman innovation term. This renders a robust, efficient and inherent data association in a single step. Extensive flight experiments were conducted to demonstrate the effectiveness of our approach. The results prove that the direct use of entire images for feedback offers better accuracy, robustness, and efficiency than the existing feature-based (LK) method. The iterated update scheme improves the estimation with a minimal increase in computation power. Further analysis comparing the proposed method against two state-of-the-art VINS reveals that the accuracy of the velocity estimates calculated from the proposed method is comparable with the two benchmark VINS. It should be highlighted that the exclusion of mapping (and therefore, comprehensive odometry information) and the single plane assumption in the proposed estimator permits it to be ≈ 15 -30 times faster than the two VINS. Finally, additional flights were performed to showcase the estimator's ability to determine the plane's normal vector. The results suggest that the achieved estimation performance is not adversely affected when the robot flies over non-horizontal surfaces. Overall, this work offers an attractive lightweight navigation solution for aerial robots with limited computational power.

Possible future directions include the extension of the framework to be applicable with the observation of multiple planes by dealing with the planar area segmentation and ego-motion estimation in one step.

REFERENCES

- [1] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visualinertial odometry using nonlinear optimization," *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2015.
- [2] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.
- [3] R. Mur-Artal and J. D. Tardós, "Visual-inertial monocular slam with map reuse," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 796–803, 2017.
- [4] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint kalman filter for vision-aided inertial navigation," in *IEEE International Conference on Robotics and Automation*, 2007, pp. 3565–3572.
- [5] S. Weiss, M. W. Achtelik, S. Lynen, M. C. Achtelik, L. Kneip, M. Chli, and R. Siegwart, "Monocular vision for long-term micro aerial vehicle state estimation: A compendium," *Journal of Field Robotics*, vol. 30, no. 5, pp. 803–831, 2013.
- [6] Z. Huai and G. Huang, "Robocentric visualinertial odometry," *The International Journal of Robotics Research*, 2019.
- [7] H. Liu, M. Chen, G. Zhang, H. Bao, and Y. Bao, "Ice-ba: Incremental, consistent and efficient bundle adjustment for visual-inertial slam," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [8] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework," *International Journal of Computer Vision*, vol. 56, no. 3, pp. 221–255, 2004.
- [9] D. Izzo and Croon, "Landing with time-to-contact and ventral optic flow estimates," *Journal of Guidance Control & Dynamics*, vol. 35, no. 4, p. 1362, 2012.
- [10] H. W. Ho, G. C. de Croon, and Q. Chu, "Distance and velocity estimation using optical flow from a monocular camera," *International Journal of Micro Air Vehicles*, vol. 9, no. 3, pp. 198–208, 2017.
- [11] V. Grabe, H. H. Blthoff, D. Scaramuzza, and P. R. Giordano, "Nonlinear ego-motion estimation from optical flow for online control of a quadrotor uav," *International Journal of Robotics Research*, vol. 34, no. 8, pp. 1114–1135, 2015.
- [12] M.-D. Hua, N. Manerikar, T. Hamel, and C. Samson, "Attitude, linear velocity and depth estimation of a camera observing a planar target using continuous homography and inertial data," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1429–1435.
- [13] H. W. Ho, G. C. H. E. de Croon, E. van Kampen, Q. P. Chu, and M. Mulder, "Adaptive gain control strategy for constant optical flow divergence landing," *IEEE Transactions on Robotics*, vol. 34, no. 2, pp. 508–516, April 2018.
- [14] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "Dtm: Dense tracking and mapping in real-time," in *2011 international conference on computer vision*. IEEE, 2011, pp. 2320–2327.
- [15] H. Jin, P. Favaro, and S. Soatto, "A semi-direct approach to structure from motion," *The Visual Computer*, vol. 19, no. 6, pp. 377–394, 2003.
- [16] M. Bloesch, M. Burri, S. Omari, M. Hutter, and R. Siegwart, "Iterated extended kalman filter based visual-inertial odometry using direct photometric feedback," *The International Journal of Robotics Research*, vol. 36, no. 10, pp. 1053–1072, 2017.
- [17] H. Zhang and J. Zhao, "Bio-inspired vision based robot control using featureless estimations of time-to-contact," *Bioinspiration & Biomimetics*, vol. 12, no. 2, p. 025001, 2016.
- [18] P. Chirarattananon, "A direct optic flow-based strategy for inverse flight altitude estimation with monocular vision and IMU measurements," *Bioinspiration & biomimetics*, vol. 13, no. 3, p. 036004, 2018.
- [19] B. M. Bell and F. W. Cathey, "The iterated kalman filter update as a gauss-newton method," *IEEE Transactions on Automatic Control*, vol. 38, no. 2, pp. 294–297, 1993.
- [20] J. Shu and P. Chirarattananon, "A quadrotor with an origami-inspired protective mechanism," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3820–3827, 2019.
- [21] Y. Chen, H. Zhao, J. Mao, P. Chirarattananon, E. F. Helbling, N.-s. P. Hyun, D. R. Clarke, and R. J. Wood, "Controlled flight of a microrobot powered by soft artificial muscles," *Nature*, vol. 575, no. 7782, pp. 324–329, 2019.
- [22] Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry, *An invitation to 3-d vision: from images to geometric models*. Springer Science & Business Media, 2012, vol. 26.
- [23] J. Shi and C. Tomasi, "Good features to track," in *IEEE Conference on Computer Vision & Pattern Recognition*, 2002.
- [24] D. Sibley, C. Mei, I. D. Reid, and P. Newman, "Adaptive relative bundle adjustment," in *Robotics: science and systems*, vol. 32, 2009, p. 33.