Robotics and Autonomous Systems 107 (2018) 262–279

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Visual–inertial estimation of velocity for multicopters based on vision motion constraint☆

Heng Deng [a,*], Usman Arif [a], Qiang Fu [a,b], Zhiyu Xi [a], Quan Quan [a], Kai-Yuan Cai [a]

[a] School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China
[b] School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China

HIGHLIGHTS

- A visual–inertial method to estimate velocity for multicopters.
- A Kalman filter which integrates monocular camera-based estimations with IMU data.
- No need for any prior knowledge of the environment or any external sensors.
- An efficient approach proposed based on MS algorithm to detect outliers.
- Observability analysis performed to verify the feasibility of the proposed method.

ARTICLE INFO

Article history:
Received 6 November 2017
Received in revised form 6 May 2018
Accepted 23 June 2018
Available online 4 July 2018

Keywords:
Visual–inertial
Velocity estimation
Multicopters
Observability analysis
Mean shift
Kalman filter

ABSTRACT

Velocity estimation is essential for multicopters to guarantee flight stability and maneuverability. For such a purpose, this paper proposes a new method for multicopter velocity estimation based on visual and inertial information in GPS-denied or confined environments. In this method, no map, artificial landmark of the environment is required, and only the off-the-shelf onboard sensors in a multicopter including a low-cost Inertial Measurement Unit (IMU), a downward-looking monocular camera and an ultrasonic range finder facing downwards are exploited to constitute the vision motion constraint. This constraint connects metric velocity with the point correspondences between successive images in which an efficient approach based on Mean Shift (MS) algorithm is developed to detect outliers and select optimal matching points. Then, it is theoretically verified that the estimation system is observable based on observability analysis. Furthermore, combined with the vision motion constraint and a multicopter dynamic model, the metric velocity is estimated using a standard Linear Kalman Filter (LKF). Finally, the proposed method is tested with a collection of synthetic data from simulation as well as flight experiments using real data from DJI Matrice 100 and Guidance. The simulation and experimental results indicate that the proposed method can accurately estimate the velocity of the multicopter in GPS-denied or confined environments.

1. Introduction

In recent years, multicopters, a kind of unmanned aerial vehicles (UAVs), have attracted increasing attention in the field of both academic research and industrial applications. Since multicopters are highly maneuverable and enable relatively safe and low-cost experimentation in navigation, mapping, and control strategies [1], they are widely used in a range of mission scenarios, such as search and rescue [2], load transportation [3], aerial manipulation [4], surveillance [5] and agricultural application [6]. However, there exist some scientific and technological challenges. Environment sensing and autonomous navigation, which is crucial to guarantee stability and safety for multicopters, remains an open and challenging issue. As stated in [7], accurate state estimation is always a fundamental necessity to implement fully autonomous manipulation in complex environments. Besides, accurate velocity estimation is required for multicopters since velocity feedback will increase damping to improve the stability and in return make multicopters more tractable.

As for velocity estimation of UAVs, a massive amount of research at the beginning has focused on indoor research using external motion capture systems such as Vicon [8,9] and outdoor applications using GPS signals [10–12]. However, these approaches rely mainly on external positioning systems, restricting UAVs to be used in a wide range of applications. While Vicon exhibits

☆ This work is supported by the National Key Project of Research and Development Plan under Grant 2016YFC1402500.
* Corresponding author.
E-mail address: dengheng@buaa.edu.cn (H. Deng).

https://doi.org/10.1016/j.robot.2018.06.010
0921-8890/

excellency with multiple high-resolution external cameras to track the pose of multicopters with submillimeter accuracy, it demands a complicated installation and calibration process, and it is infeasible for the vast outdoor environment. Besides, the operation range is limited within the field of view of the vision system. GPS signals may not be available or sufficiently reliable in some limited or confined areas without high-quality satellite signals, such as forests and buildings. Therefore, onboard sensing, especially the onboard vision has been a promising sensor modality for small autonomous multicopters since it does not require energy to interrogate the environment, and it can provide rich information and span wide field of view [13]. In general, visual–inertial fusion using onboard vision is a widely-used approach to provide more accurate velocity estimation for multicopters in limited environments [14,15]. Some attempts are using artificial landmarks or user-specified points of known position and appearance [16], but they can be reasonably accurate only if the target is detected successfully and quickly. Thus, they are limited to some scenarios with individual targets and relies pretty much on known features. As the work [17] shows, the automatic landing of a Micro Aerial Vehicle (MAV) on a moving vehicle was implemented with successful flight tests at up to 50 km/h. They fused together measurements from the MAV's onboard integrated navigation system, from multiple cameras tracking a visual fiducial marker and from the IMU and GPS data on the ground vehicle.

While the methods above require some knowledge or modifications to the environment, it is a better choice to develop a more general approach for the unknown scene. Simultaneous Localization And Mapping (SLAM) is a conventional technique for multicopter navigation in unknown environments. Weiss et al. in [18] enabled the Micro Aerial Vehicle (MAV) to determine its position autonomously and consequently stabilize itself. The work has been viewed as the first implementation for a MAV to navigate autonomously through an unknown environment, in which a monocular camera is used as the only exteroceptive sensor independent of any external aid like GPS or artificial landmarks. Following these results, Weiss has made more subsequent improvements to deal with practical issues such as scale drift, time delays, online estimation in [19,20]. Shen et al. in [21] proposed a method to estimate the velocity through a SLAM algorithm and an unscented Kalman filter which fused information from stereo cameras and inertial measurements. However, the data association and loop closure in SLAM required hundreds of points to be stored demanding more computational power and such process was quite complicated.

Compared to SLAM, optical flow is an alternative approach. Herissé et al. in [22] proposed a system to implement hovering and landing on a moving platform based on optical flow. However, it only provided the scaled linear velocity. Parrot AR.Drone [23] was the first commercial product to use an onboard downward-looking monocular camera and an ultrasonic range finder to measure metric velocity to stabilize itself based on optical flow , but the hardware design and software implementation were closed source. Similarly, PX4FLOW [24] also used a monocular camera to compute velocity with an ultrasonic sensor for scaling. However, it could only deal with a small resolution of 64 × 64 pixels limiting the measurement range and accuracy. Besides, the velocity measurements are only available when the multicopter flies over at most five meters above the ground and the ground is restricted to be relatively flat as discussed in the literature [25]. Grabe et al. in [26] proposed and experimentally verified an onboard velocity estimation and closed-loop control using the observed optical flow based on the continuous homography constraint. Homography, which contains rich information between two successive images, has been successfully applied to vision-based navigation missions. Zhao et al. in [27] designed a homography-based vision-aided inertial navigation system to provide drift-free velocity and attitude estimation for UAV stabilization. The work in [28] claimed that the attitude, velocity, and IMU measurement biases are observable during a time interval based on the assumption that multiple salient and repeatable feature points can be extracted and matched between two successive images according to their similarity. However, in practice, the matching points are usually contaminated by outliers. Therefore, it is necessary to detect and reject outliers to guarantee an accurate and robust estimation.

In contrast to existing visual estimation methods, we do not require any prior knowledge of the scene, nor do we need any external sensors like GPS or motion capture systems. It is assumed that the states of the multicopter flying in GPS-denied or confined spaces only come from the onboard sensors including an IMU, an ultrasonic range finder and a downward-looking monocular camera without any other exteroceptive sensor. For the monocular vision system, there is not any map or artificial landmark in the environment with the scene supposed as a flat plane. For the inertial system, unknown constant biases corrupt the measurements of the low-cost IMU so that they must be estimated and then compensated online.

To this end, this paper proposes a new visual–inertial estimation of metric velocity for multicopters based on vision motion constraint. The constraint is related to the corresponding features between two successive images and contains the velocity information directly. The mismatching of the features may indeed cause an error in the estimation. Therefore, an efficient approach based on Mean Shift (MS) algorithm is developed to detect outliers and select optimal matching points. It is proved that only one matching point is required to obtain the estimate based on observability analysis. More specially, combined with the vision motion constraint, the metric velocity is estimated using a standard Linear Kalman Filter (LKF) with a unique multicopter dynamic model. In contrast to existing studies, the major contributions of this paper are: (1) no need for any prior knowledge of the environment or any external sensors with only one feature correspondence required, (2) an efficient approach proposed based on MS algorithm to detect outliers and select optimal matching points between two successive images, and (3) an observability analysis performed to verify the feasibility of the proposed visual–inertial estimation.

The remainder of the paper is organized as follows. The problem formulation is given in Section 2. Section 3 presents the design of the proposed visual–inertial estimation system. In Section 4, vision motion constraint and optimal matching points selection based on MS algorithm are described and proved. In Section 5, the proposed method is theoretically verified through observability analysis, and then the procedure of discrete LKF is given. Section 6 shows the simulation and experimental results to validate the proposed estimation method and Section 7 gives the conclusions and future research plan.

2. Problem formulation

2.1. Preliminaries

2.1.1. Notations and definitions

Note that the notations and definitions in this paper are consistent with that in [29]. Let $\mathbb{R}^{m\times n}$ denote a real matrix with m rows and n columns while $\mathbb{R}^n$ an n-dimensional real column vector. Define $\mathbf{A}^{\mathrm{T}}$ and $\mathbf{A}^{-1}$ as transpose and inverse of the corresponding matrix $\mathbf{A}$, respectively. Let $\mathbf{I}_n$ denote an n-dimensional identity matrix and $\mathbf{0}_{m\times n}$ is a null matrix of dimension $m \times n$. The symbol $\mathbf{e}_3$ denotes a unit vector $[0 \quad 0 \quad 1]^{\mathrm{T}}$. For an arbitrary vector $\mathbf{a} = [a_1 \quad a_2 \quad a_3]^{\mathrm{T}} \in \mathbb{R}^3$, define the corresponding skew symmetric matrix

$$[\mathbf{a}]_\times = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \in \mathbb{R}^{3\times 3}$$

such that $\mathbf{a}\times\mathbf{b} = [\mathbf{a}]_\times\mathbf{b}$ stands for the cross product with the vector $\mathbf{b} = [b_1 \quad b_2 \quad b_3]^{\mathrm{T}} \in \mathbb{R}^3$.

As shown in Fig. 1, there are three coordinate frames defined in the estimation system: Earth-Fixed Coordinate Frame (EFCF), Aircraft-Body Coordinate Frame (ABCF), and Camera Coordinate Frame (CCF). The EFCF $\{\mathrm{e}\} = \{o_\mathrm{e}x_\mathrm{e}y_\mathrm{e}z_\mathrm{e}\}$ denotes a local North-East-Down (NED) frame with the coordinate origin o_e located on the ground plane or on the initial position where the multicopter takes off. Since the ground plane is assumed to be horizontal, the $o_\mathrm{e}x_\mathrm{e}y_\mathrm{e}$ plane coincides with the ground plane, and the third component of any ground point remains zero. The ABCF $\{\mathrm{b}\} = \{o_\mathrm{b}x_\mathrm{b}y_\mathrm{b}z_\mathrm{b}\}$ is a right-hand frame fixed to a multicopter with a height h above the ground plane. The Center of Gravity (CoG) of the multicopter is chosen as the origin o_b of frame {b}. The IMU modules are mounted in the CoG of the multicopter and the measurements of IMU are w.r.t. frame {b}. A monocular camera is attached to the bottom of the multicopter in a downward-looking direction with a small offset to the CoG. We use $\mathbf{R}_\mathrm{b}^\mathrm{c}$ and $\mathbf{T}_{o_\mathrm{b}}^\mathrm{c}$ to describe the rotation and translation from frame {b} to frame {c} which can be calibrated in advance. For any ground point $\mathbf{p}$, denote ${}^\mathrm{e}\mathbf{p}$ and ${}^\mathrm{b}\mathbf{p}$ as the coordinates of $\mathbf{p}$ in frame {e} and {b}, respectively. They satisfy

$$ {}^\mathrm{e}\mathbf{p} = \mathbf{R}_\mathrm{b}^\mathrm{e}\cdot\ {}^\mathrm{b}\mathbf{p} + \mathbf{T}_{o_\mathrm{b}}^\mathrm{e}, \tag{1} $$

where $\mathbf{R}_\mathrm{b}^\mathrm{e}$ and $\mathbf{T}_{o_\mathrm{b}}^\mathrm{e}$ represent the rotation matrix and translation vector from frame {b} to frame {e}, respectively.

Remark 1. In general, the monocular camera is not precisely mounted in the CoG of the multicopter because of the limited installation space and installation error. Thus, there are subsistent rotation and translation matrices between frame {c} and frame {b}, which can be roughly calibrated before real experiment. In this paper, we account for the difference between the two frames and unify the measurements in the same frame such as the body frame. However, without loss of generality, the camera frame is assumed to be the same as the body frame in the derivation of the proposed estimation method.

2.1.2. Inertial measurement model

Inertial measurements come from a three-axis accelerometer, a three-axis gyroscope, and an ultrasonic range finder. The measurement model of each sensor is separately built as follows.

Accelerometers are fixed to the ABCF, which can measure specific forces, i.e., the nongravitational acceleration along different body axes. ${}^\mathrm{b}\mathbf{a}_\mathrm{m} \in \mathbb{R}^3$ denote the reading of accelerometers w.r.t. the ABCF, and the accelerometer model is built as

$$ \begin{aligned} {}^\mathrm{b}\mathbf{a}_\mathrm{m} &= {}^\mathrm{b}\mathbf{a} + \mathbf{b}_\mathrm{a} + \mathbf{n}_\mathrm{a} \\ \dot{\mathbf{b}}_\mathrm{a} &= \mathbf{n}_{\mathbf{b}_\mathrm{a}}, \end{aligned} \tag{2} $$

where ${}^\mathrm{b}\mathbf{a}$ denotes the true value of the specific force, $\mathbf{b}_\mathrm{a}$ is the bias of acceleration, and noises $\mathbf{n}_\mathrm{a}$, $\mathbf{n}_{\mathbf{b}_\mathrm{a}}$ are often considered to be Gaussian White Noises (GWNs).

Gyroscopes measure angular velocity along different axes. ${}^\mathrm{b}\boldsymbol{\omega}_\mathrm{m} \in \mathbb{R}^3$ denote the reading of gyroscopes w.r.t. the ABCF, and the gyroscope model is built as

$$ \begin{aligned} {}^\mathrm{b}\boldsymbol{\omega}_\mathrm{m} &= {}^\mathrm{b}\boldsymbol{\omega} + \mathbf{b}_\mathrm{g} + \mathbf{n}_\mathrm{g} \\ \dot{\mathbf{b}}_\mathrm{g} &= \mathbf{n}_{\mathbf{b}_\mathrm{g}}, \end{aligned} \tag{3} $$

where ${}^\mathrm{b}\boldsymbol{\omega}$ denotes the true value of angular velocity, $\mathbf{b}_\mathrm{g}$ is the corresponding bias, and noises $\mathbf{n}_\mathrm{g}$, $\mathbf{n}_{\mathbf{b}_\mathrm{g}}$ are often considered to be GWNs.

Remark 2. In general, the gyroscopes are reasonably robust to noise and adequately reliable so that the bias $\mathbf{b}_\mathrm{g}$ can be relatively small. Furthermore, considering the de-biased angular velocity is directly output by a nonlinear complementary filter for a mature autopilot, the bias $\mathbf{b}_\mathrm{g}$ is measured and assumed to be known in this paper.

The height of a multicopter h can be obtained by an ultrasonic range finder and Euler angles that

$$ h = d_\mathrm{sonar}\cos\theta\cos\phi, \tag{4} $$

where $d_\mathrm{sonar} \in \mathbb{R}_+ \cup \{0\}$ is the measurement of ultrasonic range finder, and the angles θ and ϕ are pitch and roll angles, respectively.

2.1.3. Multicopter dynamic model

For convenience, the multicopter is assumed as a rigid body with constant mass and moments of inertia. Note that forces acting on the multicopter are the gravity and propeller thrust. Concretely, the gravity acts along the positive direction of the $o_\mathrm{e}z_\mathrm{e}$ axis while the propeller thrust acts along the negative direction of the $o_\mathrm{b}z_\mathrm{b}$ axis. Thus, one has

$$ {}^\mathrm{e}\dot{\mathbf{v}} = g\mathbf{e}_3 - \frac{f}{m}\mathbf{R}_\mathrm{b}^\mathrm{e}\mathbf{e}_3, \tag{5} $$

where $f \in \mathbb{R}_+ \cup \{0\}$ denotes the total propeller thrust, and $g \in \mathbb{R}_+$ is the acceleration of gravity. Intuitively, the direction of the thrust points upwards. And the rotation matrix $\mathbf{R}_\mathrm{b}^\mathrm{e}$ is given by Box I.

Substituting the relationship ${}^\mathrm{e}\mathbf{v} = \mathbf{R}_\mathrm{b}^\mathrm{e}\cdot\ {}^\mathrm{b}\mathbf{v}$ into Eq. (5) and taking the derivative w.r.t. time on both sides, one has

$$ {}^\mathrm{b}\dot{\mathbf{v}} = -[{}^\mathrm{b}\boldsymbol{\omega}_\mathrm{m} - \mathbf{b}_\mathrm{g}]_\times\cdot\ {}^\mathrm{b}\mathbf{v} + \left(\mathbf{R}_\mathrm{b}^\mathrm{e}\right)^{\mathrm{T}}\mathbf{e}_3 g - \mathbf{e}_3 u, $$

where $u = f/m$ is the acceleration generated by the propeller thrust to balance out the gravity effect. Assume $u = g - b_u$, where b_u denotes the thrust bias, and it is, in fact, the deviation from the nominal command that should be generated to compensate for the gravity. Then, the unique dynamic model of the multicopter can be denoted as

$$ \begin{aligned} {}^\mathrm{b}\dot{\mathbf{v}} &= -[{}^\mathrm{b}\boldsymbol{\omega}_\mathrm{m} - \mathbf{b}_\mathrm{g}]_\times\cdot\ {}^\mathrm{b}\mathbf{v} + \left(\mathbf{R}_\mathrm{b}^\mathrm{e}\right)^{\mathrm{T}}\mathbf{e}_3 g - \mathbf{e}_3(g - b_u) \\ \dot{b}_u &= n_u, \end{aligned} \tag{6} $$

where n_u is often considered as a GWN with variance Q_u. From Eq. (6), it is calculated that the estimated velocity ${}^\mathrm{b}\mathbf{v}$ is related to the pitch angle θ, roll angle ϕ, and de-biased angular velocity ${}^\mathrm{b}\boldsymbol{\omega}_\mathrm{m} - \mathbf{b}_\mathrm{g}$ obtained from a complementary filter using IMU information.

2.1.4. Linear camera model

A simplified linear pinhole camera model is shown in Fig. 2. The camera model projects a 3-dimensional ground point $\mathbf{p} = [p_{x_\mathrm{e}} \quad p_{y_\mathrm{e}} \quad p_{z_\mathrm{e}}]^{\mathrm{T}}$ in the EFCF to a 2-dimensional image point $\tilde{\mathbf{p}} = [u \quad v]^{\mathrm{T}}$ in the CCF with a relationship as

$$ s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{K}\begin{bmatrix} \mathbf{R}_\mathrm{e}^\mathrm{c} & \mathbf{T}_{o_\mathrm{e}}^\mathrm{c} \end{bmatrix}\begin{bmatrix} p_{x_\mathrm{e}} \\ p_{y_\mathrm{e}} \\ p_{z_\mathrm{e}} \\ 1 \end{bmatrix}, \tag{7} $$

where $s \in \mathbb{R}_+$ is a scaling factor representing the depth information of the point $\mathbf{p}$ w.r.t. the CCF. Define $\mathbf{K} \in \mathbb{R}^{3\times3}$ as an intrinsic camera matrix related to the inner structure of the camera. Let $\bar{\mathbf{p}} = [x \quad y \quad 1]^{\mathrm{T}}$ denote as the normalized homogeneous coordinate of the image point $[u \quad v]^{\mathrm{T}}$, then the normalized coordinate satisfies

$$ \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{K}^{-1}\begin{bmatrix} u \\ v \\ 1 \end{bmatrix}. \tag{8} $$

$$\mathbf{R}_{\mathrm{b}}^{\mathrm{e}} = \begin{bmatrix} \cos\theta\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix}.$$

Box I.

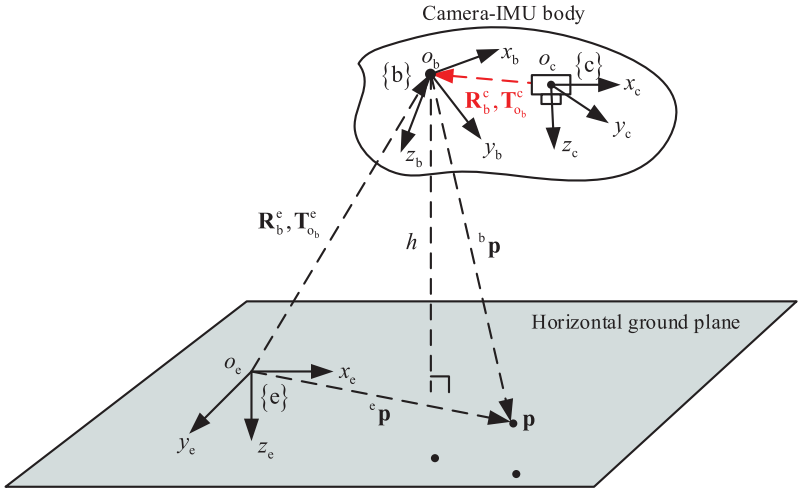


Fig. 1. The definitions of coordinate frames.

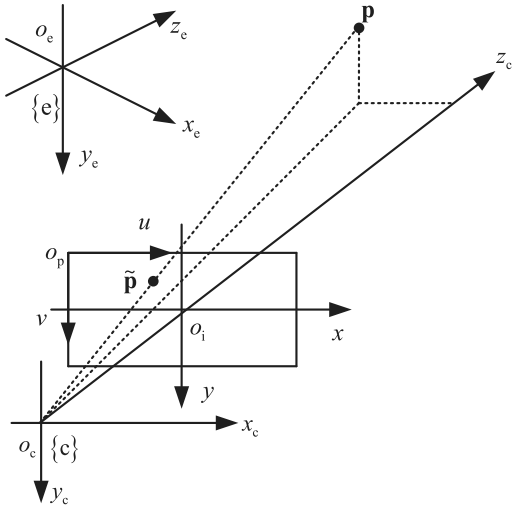


Fig. 2. The linear pinhole camera model.

2.2. Objective

To clarify the objective, we make some assumptions as follows.

Assumption 1. Any point $\mathbf{p}$ is on the ground plane and satisfies $\mathbf{e}_3^{\mathrm{T}} \cdot {}^{\mathrm{e}}\mathbf{p} = 0$, i.e., the third component of ${}^{\mathrm{e}}\mathbf{p}$ is zero.

Assumption 2. The multicopter height and inertial information including acceleration and angular velocity are measured through onboard sensors coupled with some biases and noises.

Towards the assumptions above, the objective is to accurately determine the velocity of the multicopter w.r.t. the body frame based on the fusion of IMU, monocular vision, and an ultrasonic range finder. We denote the estimated velocity as ${}^{\mathrm{b}}\mathbf{v} = [v_{x_{\mathrm{b}}} \quad v_{y_{\mathrm{b}}} \quad v_{z_{\mathrm{b}}}]^{\mathrm{T}}$.

Remark 3. In fact, the major reason of Assumption 1 is that the distance to the ground is considered to be approximately constant combined with the ultrasonic range finder. In practice, there may exist many scenarios which can be considered as flat planes. For example, a scene is flat when a downward-looking camera is mounted on a multicopter flying indoors. For the outdoor environment, the multicopter always flies at a high altitude that the camera scene can be assumed as a flat plane with GPS, and the most common approach used for altitude estimation in outdoor environments is fusion of multiple sensors: GPS, barometric pressure sensor, ultrasonic range finder, and possibly acceleration (as in M100). GPS-only altitude is inaccurate unless differential/RTK GPS is used. Besides, the GPS is not utilized in this paper, the altitude is obtained by the ultrasonic range finder which has its effective range limited to five meters at most. Therefore, the low-attitude environment with flat ground is considered in the proposed method.

Remark 4. Although sensor data contains noise, we can use them as inputs to the estimation algorithm. The height h can be measured directly by an ultrasonic range finder when the multicopter flies at a low altitude within the maximum effective range. Rotation matrix and de-biased angular velocity can be acquired from the IMU.

Remark 5. The camera is calibrated in advance so that the intrinsic camera parameter can be utilized directly.

3. Design of the visual–inertial estimation system

The structure of the proposed velocity estimation system is given in Fig. 3. The sensor measurements are merged by an 8th-order LKF. The system states are one-dimension height, 3-dimension velocity w.r.t. the ABCF, 3-dimension acceleration

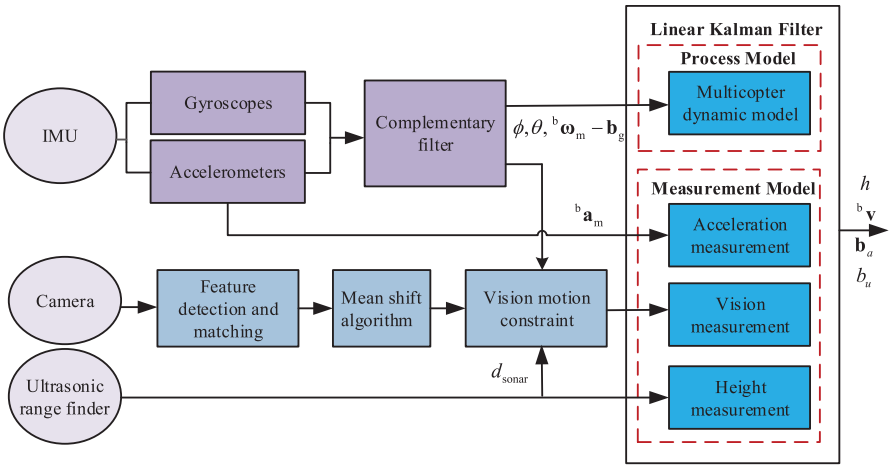


Fig. 3. The structure of the velocity estimation system. The pitch angle θ, roll angle ϕ and de-biased angular velocity ${}^{\mathrm{b}}\boldsymbol{\omega}_{\mathrm{m}} - \mathbf{b}_{\mathrm{g}}$ can be obtained by a complementary filter using IMU data. This information can be utilized to compose a visual motion constraint and a multicopter dynamic model. Finally, the velocity can be estimated by an LKF with the multicopter dynamic model based on feeds from the accelerometers, ultrasonic range finder, and monocular vision.

bias, and one-dimension thrust bias. It should be noted that the gyroscope measurements enter the LKF through the process model which propagates the system states while the measurements of the monocular vision, ultrasonic range finder, and accelerometers enter the LKF through the measurement model which updates the system states.

The estimation system consists of two primary sensors: a low-cost IMU (a three-axis accelerometer and a three-axis gyroscope) and a monocular camera. The IMU measures the specific force (the acceleration eliminating gravity) and angular velocity of the multicopter. Considering the inaccurate measurements of the low-cost IMU, it is assumed that the IMU measurements are corrupted by GWNs and constant (or slowly time-varying) biases. A commonly-used method to estimate the biases is to average the reading of these sensors for a short time while the multicopter remains static on the ground before taking off. Different from this method, in this paper, it is assumed that the biases may vary each time the IMU is initialized so that they need to be estimated online. The monocular camera is looking downwards to capture images of the ground scene. The vision measurements, namely the vision motion constraint, relate the image pixel information with the space motion of the multicopter based on matching points between two successive images. In our method, the Speed Up Robust Features (SURF) [30] are detected and extracted, and the scale-invariant features are matched based on optical flow and pyramid principle. Similarly, the vision measurements are assumed to be corrupted by GWNs. Furthermore, the depth information is essential for the metric measurement of the velocity due to the scale ambiguity of monocular vision, and the ultrasonic range finder can directly obtain the depth with small noise.

4. Monocular vision measurements

In this section, we analyze the measurement of the monocular vision. The onboard monocular camera is directly looking downwards to capture images of the ground scene. There exists a vision motion constraint between matching features of two successive images.

4.1. Vision motion constraint

As illustrated in Fig. 4, Define Δt as the sampling time interval, and let t_k and $t_{k-1} = t_k - \Delta t$ be the current and the last sampling time instant, respectively. Given two images captured at time t_k and t_{k-1}, the corresponding rotation matrices (note that the rotation is from frame {b} to frame {e}) and height information are denoted as $\mathbf{R}_k, h_k > 0$ and $\mathbf{R}_{k-1}, h_{k-1} > 0$. Denote coordinates of the CoG of the multicopter in the EFCF as $\mathbf{T}_k, \mathbf{T}_{k-1}$ expressed in the two successive images. Thus, the vision motion constraint is expressed as a theorem in the following.

Theorem 1 (*Vision Motion Constraint*)**.** *In the planar ground surface, given two successive images captured at time t_{k-1} and t_k, the normalized coordinates of corresponding point correspondences are denoted as*

$$\begin{aligned}\bar{\mathbf{p}}_{k-1} &= [x_{k-1} \quad y_{k-1} \quad 1]^{\mathrm{T}} \\ \bar{\mathbf{p}}_{k} &= [x_{k} \quad y_{k} \quad 1]^{\mathrm{T}}\end{aligned}.$$

Combined with corresponding rotation matrices and heights denoted as $\mathbf{R}_{k-1}, h_{k-1}$ and $\mathbf{R}_k, h_k$, the vision motion constraint relating the estimated velocity w.r.t. the body frame to the image pixel information is given by

$${}^{\mathrm{b}}\hat{\mathbf{v}}_k = -\frac{h_{k-1}\bar{\mathbf{p}}_{k-1}}{\Delta t \mathbf{e}_3^{\mathrm{T}}\mathbf{R}_{k-1}\bar{\mathbf{p}}_{k-1}} + \frac{h_k(\mathbf{I}_3 + [{}^{\mathrm{b}}\boldsymbol{\omega}_{k-1}]_\times \Delta t)\bar{\mathbf{p}}_k}{\Delta t \mathbf{e}_3^{\mathrm{T}}\mathbf{R}_k\bar{\mathbf{p}}_k}. \tag{9}$$

Proof of Theorem 1 is given in Appendix A. In our method, it is assumed that feature points can be easily extracted from successive images and then the potential inner-frame correspondences can be established. Based on each available point correspondence, Eq. (9) is utilized to estimate the velocity. However, there may exist some mismatching pairs because of outliers and noises in image data, thus leading to wrong velocity estimates. As illustrated in Fig. 5(a), feature points are established between two successive images. It is shown that most feature points are tracked accurately, but there also exist some mismatching points. To visualize the velocity results based on Eq. (9), we plot the horizontal velocities as shown in Fig. 5(b). We can observe that not all the correspondences are good enough to compute velocity, but in general, the velocity measurements will concentrate near the true value as the black circle indicates. Therefore, an approach based on the MS algorithm is designed to select optimal matching points between two successive images in the following.

Remark 6. Since the time interval is short, i.e., $\Delta t = 0.05$ s in the experiments, the estimated velocity is assumed to be constant during the time interval $[t_k - 1, t_k]$.

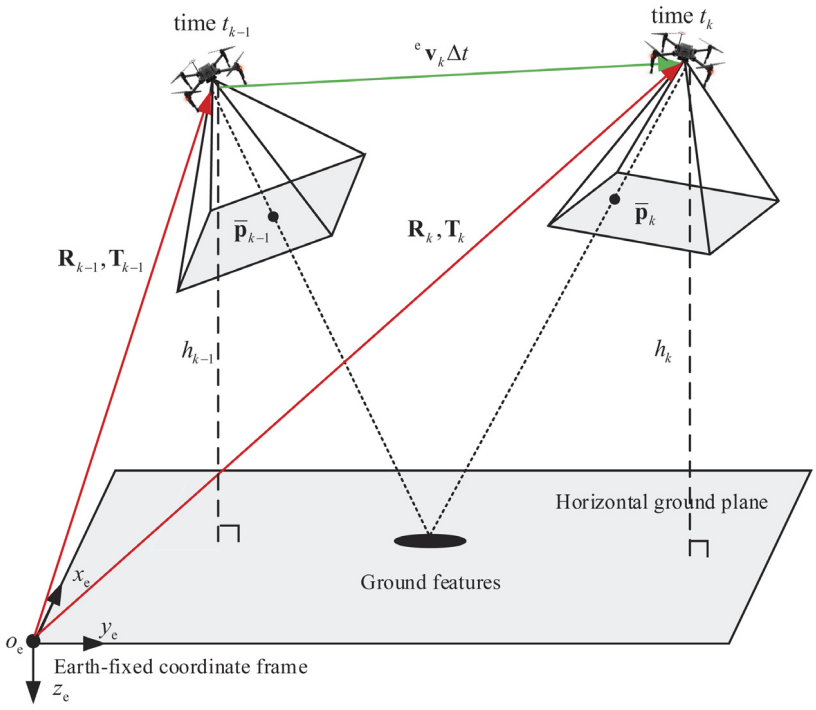


Fig. 4. An illustration of the necessary quantities on the derivation of vision motion constraint.

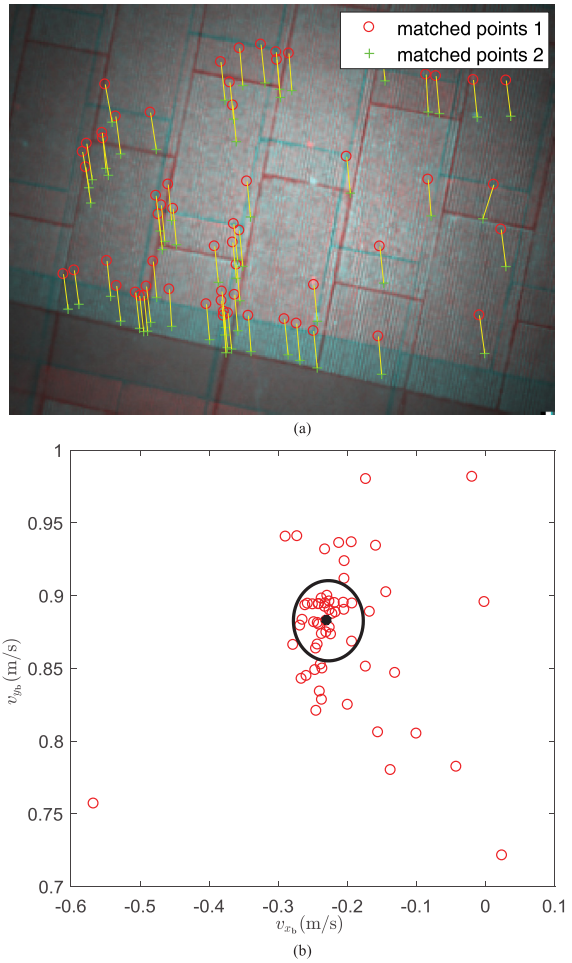


Fig. 5. (a) Results of feature detection and matching; (b) Velocities computed by Theorem 1.

4.2. Mean shift (MS) algorithm

MS algorithm is a simple nonparametric iterative technique for estimating density gradient which was proposed by Fukunaga in 1975 [31] and largely overlooked till Cheng provided an appropriate generalization 20 years later [32]. Since then, the MS algorithm has become popular with successful applications ranging from image segmentation to object tracking and clustering [33,34]. It is a stable iterative method to locate the maxima of a local density function given a collection of discretely sampled data [35].

Given n sampled data $\{\boldsymbol{\xi}_i\}_{i=1,2,\ldots,n}$ in the d-dimensional space $\mathbb{R}^d$, the multivariate kernel density estimate with only one search window radius r (also the bandwidth parameter), computed in the point $\boldsymbol{\xi} \in \mathbb{R}^d$ is given by

$$\hat{f}_r(\boldsymbol{\xi}) = \frac{c_{r,d}}{nr^d}\sum_{i=1}^{n}\exp(-\frac{1}{2}\|\frac{\boldsymbol{\xi}-\boldsymbol{\xi}_i}{r}\|^2), \tag{10}$$

where $c_{r,d}$ is a normalization constant. Then, the gradient of the kernel density estimate is established as

$$\nabla\hat{f}_r(\boldsymbol{\xi}) = -\frac{c_{r,d}}{nr^{d+2}}\sum_{i=1}^{n}(\boldsymbol{\xi}-\boldsymbol{\xi}_i)\exp(-\frac{1}{2}\|\frac{\boldsymbol{\xi}-\boldsymbol{\xi}_i}{r}\|^2). \tag{11}$$

Based on Eq. (11), when the gradient of the kernel density estimate equals zero, i.e., $\nabla\hat{f}_r(\boldsymbol{\xi}^*) = \mathbf{0}_{d\times 1}$, then $\boldsymbol{\xi}^*$ is the maxima of the estimated probability density function regardless of the value of the constant $c_{r,d}$ and

$$\boldsymbol{\xi}^* = \frac{\sum_{i=1}^{n}\boldsymbol{\xi}_i\exp(-\frac{1}{2}\|\frac{\boldsymbol{\xi}^*-\boldsymbol{\xi}_i}{r}\|^2)}{\sum_{i=1}^{n}\exp(-\frac{1}{2}\|\frac{\boldsymbol{\xi}^*-\boldsymbol{\xi}_i}{r}\|^2)}. \tag{12}$$

Furthermore, for iterative operation, the MS algorithm starts from one of the data points and iteratively update the location of the point until a maxima, namely the optimal value is reached. Thus, the optimal value among sampled data $\{\boldsymbol{\xi}_i\}_{i=1,2,\ldots,n}$ in the

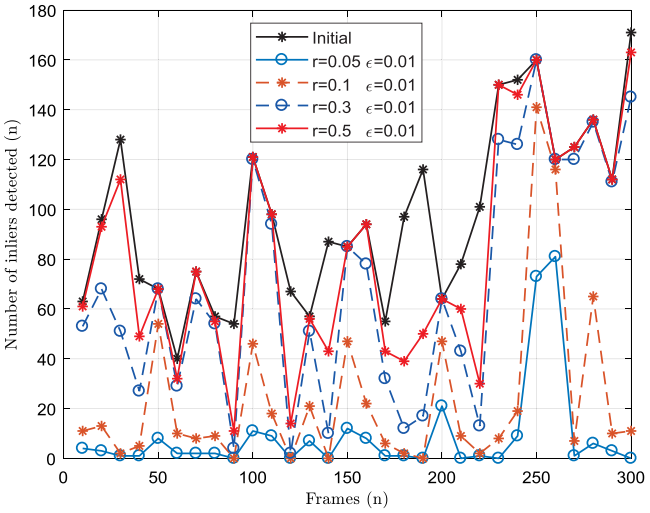


Fig. 6. Number of inliers detected w.r.t. different bandwidth r when the iteration precision ε is set to 0.01.

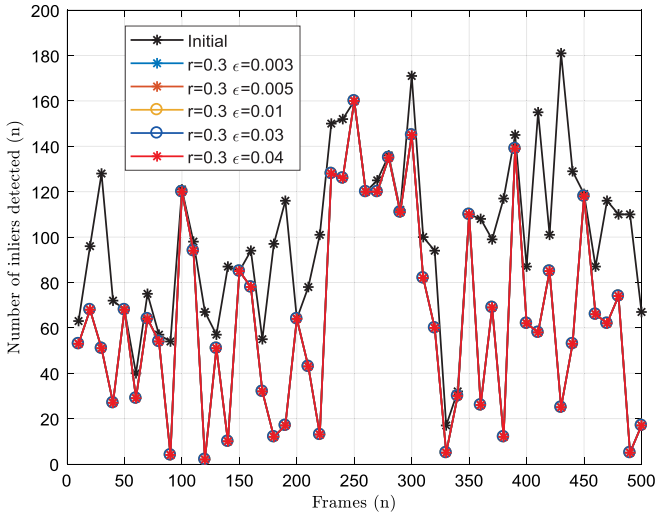


Fig. 7. Number of inliers detected w.r.t. different iteration precision ε when the bandwidth r is set to 0.3.

Table 1
Procedure of optimal matching points selection based on MS algorithm.

1.	Initialization: the bandwidth r; the initial value $\mathbf{y}_0$; the iterative precision ε
2.	Data acquisition: obtain the set of velocities computed from potential matching points, denoted as $\boldsymbol{\xi}_i$, $i = 1, 2, \ldots, n$
3.	Compute $\mathbf{y}_1$ based on Eq. (13), $j = 1$
4.	Main loop: while $\|\mathbf{y}_j - \mathbf{y}_{j-1}\| > \varepsilon$ $j = j + 1$ Compute $\mathbf{y}_j$ based on equation (13) end
5.	$\mathbf{v} = \mathbf{y}_j$
6.	Select the optimal matching points with velocity $\mathbf{v}_i$ which satisfy $\|\mathbf{v}_i - \mathbf{v}\| < r/3$
7.	Return $\mathbf{v}$, $\mathbf{v}_i$

jth iteration $\mathbf{y}_j$ can be simplified as

$$\mathbf{y}_j = \frac{\sum_{i=1}^{n} \boldsymbol{\xi}_i \exp(-\frac{1}{2}\|\frac{\mathbf{y}_{j-1}-\boldsymbol{\xi}_i}{r}\|^2)}{\sum_{i=1}^{n} \exp(-\frac{1}{2}\|\frac{\mathbf{y}_{j-1}-\boldsymbol{\xi}_i}{r}\|^2)}. \tag{13}$$

It should be noted that there is a sufficient condition for the convergence of the MS algorithm to guarantee the existence of the maxima. The conclusion and corresponding proof are summarized in [36]. When the mean shift is small enough, i.e., $\|\mathbf{y}_j - \mathbf{y}_{j-1}\| < \varepsilon$, the iteration is terminated. The procedure of the optimal matching points selection based on MS algorithm is depicted in Table 1.

During the *Initialization* process in Table 1, some parameters need to be determined. The iterative process will become fast and accurate if the initial value $\mathbf{y}_0$ is set near the neighbor of the optimal value. Considering the time interval between two successive images is short, the last estimated state is chosen as the current initial value. Towards the choice of the bandwidth r and the iterative precision ε, we need to analyze their effect on the estimation because their values are related to the accuracy and efficiency of the MS algorithm.

For such a purpose, we conduct some experiments in which two successive images captured by the downward-looking monocular camera are utilized. Then, we follow the procedure of the optimal matching points selection in Table 1 to compare the number of inliers detected before and after using the MS algorithm with a different choice of parameters.

In the experiments, we first calculate the number of inliers detected with respect to different bandwidth r when the iteration precision ε is set to 0.01. As depicted in Fig. 6, the solid black line indicates the number of inliers detected without using the MS algorithm while the other curves show the results after the process of MS algorithm. It is shown that some outliers are detected and rejected with the bandwidth controlling the searching range. When the bandwidth is small, e.g., $r = 0.05$ and almost all points are identified as outliers, resulting in few inliers detected. The number of inliers detected becomes greater with the bandwidth increasing. However, if the bandwidth is chosen as large as 0.5, the number of inliers detected almost remains unchanged and the MS algorithm takes little effect. Thus, the bandwidth value is selected as 0.3 in this paper.

Similarly, the results w.r.t different iteration precision ε when the bandwidth r is set to 0.3 are depicted in Fig. 7. It shows that the number of inliers detected remains the same with different iteration precision, and the results are reasonable because the precision only affects the number of iterations it takes the algorithm to converge. Therefore, in this paper, the bandwidth and the iteration precision are set as $r = 0.3$, $\varepsilon = 0.01$.

Based on the procedure in Table 1 with the presupposed parameters, we can calculate the histogram distribution of the estimated velocity. Fig. 8 shows the comparison to the resulting histogram of horizontal velocity before and after using MS algorithm. It is assumed that the velocity follows the Gaussian distribution. The true value is $v_{x_b} = -0.3266$ m/s, $v_{y_b} = 1.0884$ m/s. We also compute the mean and the variance for comparison. Results indicate that the distribution curves have become thinner with smaller variance after utilizing the MS algorithm and the mean remains almost unchanged. Thus, some outliers have been detected and rejected based on the proposed MS algorithm.

Remark 7. The vision motion constraint above only holds on the assumption that there exist some point correspondences between successive images. However, there may be few features that can be extracted and matched for some conditions with simple texture or the camera scene moves quickly as shown in Fig. 9. Thus, a complementary method may be considered to compensate for the insufficient of the vision motion constraint. Based on the work in [37], the brightness change constraint could be utilized to estimate the velocity when there are few point correspondences with known depth.

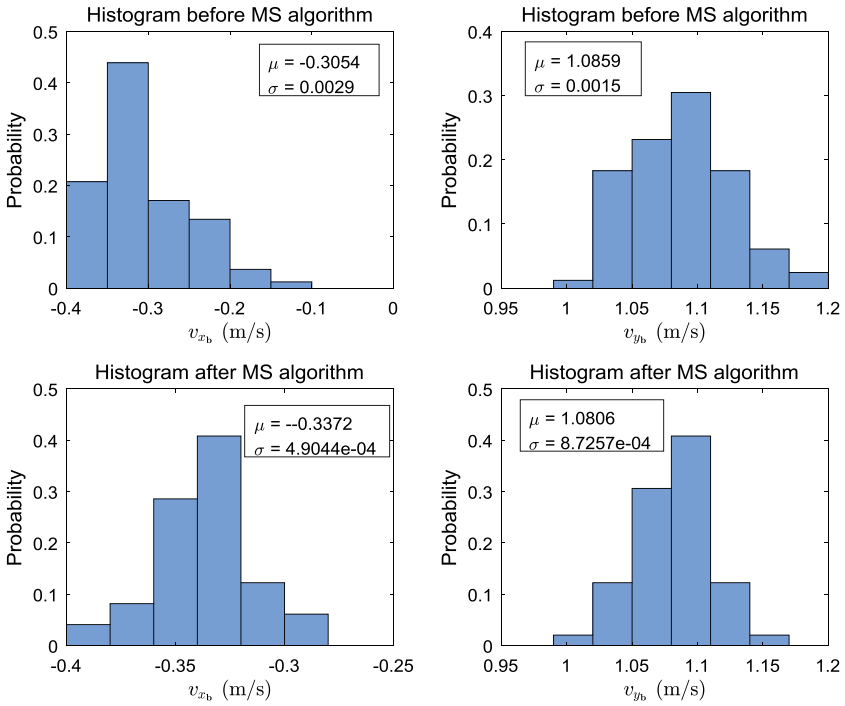


Fig. 8. Comparison to the histogram of horizontal velocity estimated before and after MS algorithm with the parameters $r = 0.3$ and $\varepsilon = 0.01$.

5. Observability analysis and Kalman filter

In this section, we will perform the observability analysis to the proposed visual–inertial estimation system. The purpose is to identify the observable quantities and theoretically verify whether the proposed system satisfies all the requirements. First, we need to obtain the process model and the measurement model of the estimation system. Then, a theorem to analyze the observability of the estimation system is proposed. Finally, we present the detailed procedure of the LKF and give some practical issues.

5.1. Process model

Define $\mathbf{x} = [h \quad {}^{\mathrm{b}}\mathbf{v} \quad \mathbf{b}_{\mathrm{a}} \quad b_u]^{\mathrm{T}} \in \mathbb{R}^8$ as the system states, based on the analysis in Section 2 , the process model of the proposed estimation system is built as

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{u} + \mathbf{w}, \tag{14}$$

where

$$\mathbf{x} = \begin{bmatrix} h \\ {}^{\mathrm{b}}\mathbf{v} \\ \mathbf{b}_{\mathrm{a}} \\ b_u \end{bmatrix}, \mathbf{A} = \begin{bmatrix} 0 & -\mathbf{e}_3^{\mathrm{T}}\mathbf{R}_{\mathrm{b}}^{\mathrm{e}} & \mathbf{0}_{1\times 3} & 0 \\ \mathbf{0}_{3\times 1} & -[{}^{\mathrm{b}}\boldsymbol{\omega}_{\mathrm{m}} - \mathbf{b}_{\mathrm{g}}]_\times & \mathbf{0}_{3\times 3} & \mathbf{e}_3 \\ \mathbf{0}_{3\times 1} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 1} \\ 0 & \mathbf{0}_{1\times 3} & \mathbf{0}_{1\times 3} & 0 \end{bmatrix},$$

$$\mathbf{u} = \begin{bmatrix} 0 \\ (\mathbf{R}_{\mathrm{b}}^{\mathrm{e}})^{\mathrm{T}}\mathbf{e}_3 g - \mathbf{e}_3 g \\ \mathbf{0}_{3\times 1} \\ 0 \end{bmatrix}, \mathbf{w} = \begin{bmatrix} 0 \\ \mathbf{0}_{3\times 1} \\ \mathbf{n}_{\mathbf{b}_{\mathrm{a}}} \\ n_u \end{bmatrix}.$$

5.2. Measurement model

The measurement of the filter comes from accelerometers, an ultrasonic range finder, and monocular vision. The measurement model of each sensor is separately built as follows.

As for the height measurement, based on Eq. (4), the measurement of an ultrasonic range finder is defined as

$$z_h = d_{\mathrm{sonar}} \cos\phi \cos\theta,$$

and then the measurement model is built as

$$z_h = \mathbf{C}_h\mathbf{x} + n_h, \tag{15}$$

where the measurement matrix is

$$\mathbf{C}_h = \begin{bmatrix} 1 & \mathbf{0}_{1\times 3} & \mathbf{0}_{1\times 3} & 0 \end{bmatrix}.$$

As for the acceleration measurement, based on Eq. (2), define the accelerometer measurement as

$$\mathbf{z}_a = {}^{\mathrm{b}}\mathbf{a}_{\mathrm{m}} + \mathbf{e}_3 g,$$

then, combined with the multicopter dynamic model (6), the measurement model of accelerometers is built as

$$\mathbf{z}_a = \mathbf{C}_a\mathbf{x} + \mathbf{n}_a, \tag{16}$$

where the accelerometer measurement matrix is

$$\mathbf{C}_a = \begin{bmatrix} \mathbf{0}_{3\times 1} & -[{}^{\mathrm{b}}\boldsymbol{\omega}_{\mathrm{m}} - \mathbf{b}_{\mathrm{g}}]_\times & \mathbf{I}_3 & \mathbf{e}_3 \end{bmatrix}.$$

As for the vision measurement, after obtaining M optimal matching points based on the MS algorithm, denoted as

$$\bar{\mathbf{p}}_{i,k-1} \leftrightarrow \bar{\mathbf{p}}_{i,k}, i = 1, 2, \ldots, M, \tag{17}$$

there will be an estimated velocity corresponding to each matching point based on vision motion constraint (9). We will choose the mean of the M velocities as vision measurement to speed up the filter. Thus, the measured velocity ${}^{\mathrm{b}}\mathbf{v}_{\mathrm{m}}$ can be described as

$${}^{\mathrm{b}}\mathbf{v}_{\mathrm{m}} = \frac{1}{M}\Big(\sum_{i=1}^{M} \frac{h_k(\mathbf{I}_3 + [{}^{\mathrm{b}}\boldsymbol{\omega}_{k-1}]_\times \Delta t)\bar{\mathbf{p}}_{i,k}}{\Delta t \mathbf{e}_3^{\mathrm{T}}\mathbf{R}_k\bar{\mathbf{p}}_{i,k}} - \sum_{i=1}^{M} \frac{h_{k-1}\bar{\mathbf{p}}_{i,k-1}}{\Delta t \mathbf{e}_3^{\mathrm{T}}\mathbf{R}_{k-1}\bar{\mathbf{p}}_{i,k-1}}\Big). \tag{18}$$

For simplicity, the measured velocity ${}^{\mathrm{b}}\mathbf{v}_{\mathrm{m}}$ in Eq. (18) is directly defined as the vision measurement $\mathbf{z}_{vis}$. Thus, the vision measurement model can be described as

$$\mathbf{z}_{vis} = \mathbf{C}_{vis}\mathbf{x} + \mathbf{n}_{vis}, \tag{19}$$

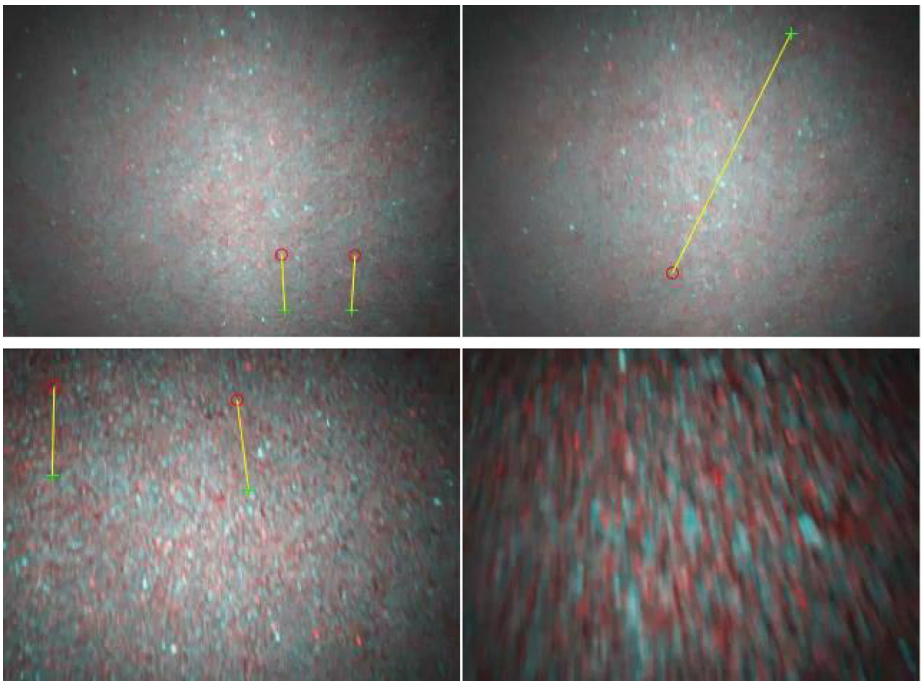


Fig. 9. Samples of captured images with few matching points. The small read circles and yellow crosses represent the matching points between successive images.

where $\mathbf{n}_{vis} \in \mathbb{R}^3$ is the vision measurement noise. Hence, the expression of $\mathbf{C}_{vis}$ is built as

$$\mathbf{C}_{vis} = \begin{bmatrix}\mathbf{0}_{3\times1} & \mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1}\end{bmatrix}.$$

To sum up, we can conclude the measurement model of the estimation system as

$$\mathbf{z} = \mathbf{C}\mathbf{x} + \mathbf{v}, \tag{20}$$

where

$$\mathbf{z} = \begin{bmatrix} z_h \\ \mathbf{z}_a \\ \mathbf{z}_{vis}\end{bmatrix}, \mathbf{v} = \begin{bmatrix} n_h \\ \mathbf{n}_a \\ \mathbf{n}_{vis}\end{bmatrix}$$

$$\mathbf{C} \triangleq \begin{bmatrix}\mathbf{C}_h \\ \mathbf{C}_a \\ \mathbf{C}_{vis}\end{bmatrix} = \begin{bmatrix} 1 & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 0 \\ \mathbf{0}_{3\times1} & -\left[{}^{\mathrm{b}}\boldsymbol{\omega}_{\mathrm{m}} - \mathbf{b}_{\mathrm{g}}\right]_\times & \mathbf{I}_3 & \mathbf{e}_3 \\ \mathbf{0}_{3\times1} & \mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1}\end{bmatrix}.$$

5.3. Observability analysis

Up to now, the process model and measurement model are built and they are linear. In this section, we derive the observability analysis of the proposed estimation system. The purpose is to identify the observable quantities and theoretically verify if the proposed estimation system can fulfill all the requirements. Since the process and measurement models are time-variant, the standard observability rank method is not suitable. For simplicity, we assume the flight condition of the multicopter as follows.

Assumption 3. The multicopter is in hovering condition or straight and steady flight with small angle and zero angular velocity, thus the multicopter states are approximately given by

$$\begin{aligned}&\phi = \theta = 0 \\ &{}^{\mathrm{b}}\boldsymbol{\omega} = \mathbf{0}_{3\times1}.\end{aligned} \tag{21}$$

Remark 8. Assumption 3 confines the flight condition of the multicopter irrespective of big maneuvering situation with large angle attitude, which leads to a convenient understanding of the observability analysis. Besides, the value of the yaw angle has no influence on the observability analysis since it is not related to the velocity w.r.t. the body frame. Thus, only the roll angle ϕ and pitch angle θ are assumed to be zero.

According to Assumption 3, the process model (14) and measurement model (20) can be simplified. First, we consider two conditions: one with vision measurement and the other without vision measurement. Both conditions are common in practice since the vision measurement is badly influenced by illumination, and the only difference between them is the measurement matrix $\mathbf{C}$.

For the condition with vision measurement, substituting the assumption (21) into the process model (14) and measurement model (20) gives the simplified model ($\mathbf{A}$, $\mathbf{C}_1$) as

$$\begin{aligned}\dot{\mathbf{x}} &= \underbrace{\begin{bmatrix} 0 & -\mathbf{e}_3^{\mathrm{T}} & \mathbf{0}_{1\times3} & 0 \\ \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{e}_3 \\ \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} \\ 0 & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 0\end{bmatrix}}_{\mathbf{A}}\mathbf{x} + \underbrace{\begin{bmatrix} 0 \\ (\mathbf{R}_{\mathrm{b}}^{\mathrm{e}})^{\mathrm{T}}\mathbf{e}_3 g - \mathbf{e}_3 g \\ \mathbf{0}_{3\times1} \\ 0\end{bmatrix}}_{\mathbf{u}} \\ &\quad + \underbrace{\begin{bmatrix} 0 \\ \mathbf{0}_{3\times1} \\ \mathbf{n}_{\mathbf{b}_{\mathrm{a}}} \\ n_u\end{bmatrix}}_{\mathbf{w}} \\ \mathbf{z} &= \begin{bmatrix} z_h \\ \mathbf{z}_a \\ \mathbf{z}_{vis}\end{bmatrix} = \underbrace{\begin{bmatrix} 1 & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 0 \\ \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} & \mathbf{I}_3 & \mathbf{e}_3 \\ \mathbf{0}_{3\times1} & \mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1}\end{bmatrix}}_{\mathbf{C}_1}\mathbf{x} \\ &\quad + \underbrace{\begin{bmatrix} n_h \\ \mathbf{n}_a \\ \mathbf{n}_{vis}\end{bmatrix}}_{\mathbf{v}}.\end{aligned} \tag{22}$$

While for the condition without vision measurement, similarly, we have the simplified model ($\mathbf{A}$, $\mathbf{C}_2$) as

Table 2
Procedure of the linear Kalman filter.

1.	Initialization: the initial state $\mathbf{x}_0$ and the initial error covariance $\mathbf{P}_0$
2.	For $k = 0$, set $\mathbf{P}_{0\|0} = \mathbf{P}_0$, $\hat{\mathbf{x}}_{0\|0} = \mathbf{x}_0$
3.	State estimate propagation: $\hat{\mathbf{x}}_{k\|k-1} = \mathbf{F}_{k-1}\hat{\mathbf{x}}_{k-1\|k-1} + \mathbf{G}_{k-1}$
4.	Error covariance propagation: $\mathbf{P}_{k\|k-1} = \mathbf{F}_{k-1}\mathbf{P}_{k-1\|k-1}\mathbf{F}_{k-1} + \mathbf{Q}_{k-1}$
5.	Kalman gain matrix: $\mathbf{K}_k = \mathbf{P}_{k\|k-1}\mathbf{C}_k^{\mathrm{T}}\left(\mathbf{C}_k\mathbf{P}_{k\|k-1}\mathbf{C}_k^{\mathrm{T}} + \mathbf{R}_k\right)^{-1}$
6.	State estimate update: $\hat{\mathbf{x}}_{k\|k} = \hat{\mathbf{x}}_{k\|k-1} + \mathbf{K}_k\left(\mathbf{z}_k - \hat{\mathbf{z}}_{k\|k-1}\right)$. where $\hat{\mathbf{z}}_{k\|k-1} = \mathbf{C}_k\hat{\mathbf{x}}_{k\|k-1}$
7.	Error covariance update: $\mathbf{P}_{k\|k} = (\mathbf{I}_n - \mathbf{K}_k\mathbf{C}_k)\,\mathbf{P}_{k\|k-1}$
8.	$k = k + 1$, Go back to Step 3

$$\dot{\mathbf{x}} = \underbrace{\begin{bmatrix} 0 & -\mathbf{e}_3^{\mathrm{T}} & \mathbf{0}_{1\times3} & 0 \\ \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{e}_3 \\ \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} \\ 0 & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 0 \end{bmatrix}}_{\mathbf{A}}\mathbf{x} + \underbrace{\begin{bmatrix} 0 \\ (\mathbf{R}_{\mathrm{b}}^{\mathrm{e}})^{\mathrm{T}}\mathbf{e}_3 g - \mathbf{e}_3 g \\ \mathbf{0}_{3\times1} \\ 0 \end{bmatrix}}_{\mathbf{u}} + \underbrace{\begin{bmatrix} 0 \\ \mathbf{0}_{3\times1} \\ \mathbf{n}_{\mathbf{b}_a} \\ n_u \end{bmatrix}}_{\mathbf{w}}$$
$$\mathbf{z} = \begin{bmatrix} z_h \\ \mathbf{z}_a \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 0 \\ \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} & \mathbf{I}_3 & \mathbf{e}_3 \end{bmatrix}}_{\mathbf{C}_2}\mathbf{x} + \underbrace{\begin{bmatrix} n_h \\ \mathbf{n}_a \\ \mathbf{n}_{vis} \end{bmatrix}}_{\mathbf{v}}. \tag{23}$$

Then, a theorem to analyze the observability of linear estimation system under the two conditions above is proposed as follows.

Theorem 2 (*Observability Analysis*)**.** *When the multicopter is in hovering condition or straight and steady flight with a small angle and zero angular velocity, the observability analysis is divided into two conditions. When there is at least one visual feature correspondence detected, the simplified system (22) is observable; When there is not any vision measurement, the simplified system (23) is unobservable, and the two unobservable quantities are horizontal velocities.*

Proof of Theorem 2 is given in Appendix B. When there are vision measurements, the conclusion that the system is observable is reasonable since at least one feature correspondence can directly calculate the estimated velocity based on the vision motion constraint (9). On the other hand, for the terrible condition without any vision measurement where only the height and acceleration measurements are available, it is proved that the horizontal velocities are unobservable, which may be inconsistent with the common sense that the velocity is just the integral of the acceleration. However, it should be noted that the velocity can be obtained by integrating the acceleration when the initial velocity is known, and the integral error will be accumulated leading the increasing error of the velocity estimate. Thus, the velocity can be propagated by integrating the acceleration based on the last estimated states just during a short time interval. If the condition without any vision measurement lasts a long time interval, the estimated error will increase.

5.4. Kalman filter

Since the system is linear so that a standard LKF is to be applied to fuse the measurements of IMU, vision, and ultrasonic range finder. First, we need to get the discrete form of the process model and measurement model. The discrete form of the process model is built as

$$\mathbf{x}_k = \mathbf{F}_{k-1}\mathbf{x}_{k-1} + \mathbf{G}_{k-1} + \mathbf{w}_{k-1},\ \mathbf{w}_{k-1} \sim \mathcal{N}(\mathbf{0}_{8\times1}, \mathbf{Q}_{k-1})$$

where

$$\mathbf{F}_{k-1} = \begin{bmatrix} 1 & -\mathbf{e}_3^{\mathrm{T}}\mathbf{R}_{\mathrm{b}}^{\mathrm{e}}\Delta t & \mathbf{0}_{1\times3} & -\frac{\Delta t^2}{2}\mathbf{e}_3^{\mathrm{T}}\mathbf{R}_{\mathrm{b}}^{\mathrm{e}}\mathbf{e}_3 \\ \mathbf{0}_{3\times1} & \mathbf{I}_3 - [{}^{\mathrm{b}}\boldsymbol{\omega}_{\mathrm{m}} - \mathbf{b}_{\mathrm{g}}]_\times\Delta t & \mathbf{0}_{3\times3} & \mathbf{e}_3\Delta t \\ \mathbf{0}_{3\times1} & \mathbf{0}_{3\times3} & \mathbf{I}_3 & \mathbf{0}_{3\times1} \\ 0 & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 1 \end{bmatrix}$$
$$\mathbf{G}_{k-1} = \begin{bmatrix} -\frac{\Delta t^2}{2}\mathbf{e}_3^{\mathrm{T}}\mathbf{R}_{\mathrm{b}}^{\mathrm{e}}((\mathbf{R}_{\mathrm{b}}^{\mathrm{e}})^{\mathrm{T}}\mathbf{e}_3 g - \mathbf{e}_3 g) \\ ((\mathbf{R}_{\mathrm{b}}^{\mathrm{e}})^{\mathrm{T}}\mathbf{e}_3 g - \mathbf{e}_3 g)\Delta t \\ \mathbf{0}_{3\times1} \\ 0 \end{bmatrix}.$$

The discrete form of the measurement model is built as

$$\mathbf{z}_k = \mathbf{C}_k\mathbf{x}_k + \mathbf{v}_k,\ \mathbf{v}_k \sim \mathcal{N}(\mathbf{0}_{7\times1}, \mathbf{R}_k) \tag{24}$$

where

$$\mathbf{z}_k = \begin{bmatrix} z_h \\ \mathbf{z}_a \\ \mathbf{z}_{vis} \end{bmatrix}, \mathbf{v}_k = \begin{bmatrix} n_h \\ \mathbf{n}_a \\ \mathbf{n}_{vis} \end{bmatrix}$$
$$\mathbf{C}_k = \begin{bmatrix} 1 & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 0 \\ \mathbf{0}_{3\times1} & -[{}^{\mathrm{b}}\boldsymbol{\omega}_{\mathrm{m}} - \mathbf{b}_{\mathrm{g}}]_\times & \mathbf{I}_3 & \mathbf{e}_3 \\ \mathbf{0}_{3\times1} & \mathbf{I}_3 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} \end{bmatrix}.$$

The details of the implementation of the LKF are depicted as in Table 2.

Besides, there are some practical issues to be noticed. First, the initial value of the system state is chosen as $\mathbf{x}_0 = [d_{\mathrm{sonar}}\cos\theta\cos\phi \quad {}^{\mathrm{b}}\mathbf{v}_{vis} \quad \mathbf{0}_{3\times1} \quad 0]^{\mathrm{T}}$ where the initial height is obtained by the ultrasonic range finder, the initial velocity is given through the vision motion constraint, and the other initial values can be set to zero. Second, there exist in fact two update rates in the filter. The update rate of IMU measurement is usually higher than that of vision or height measurement. Besides, their measurements are not synchronous. To solve this problem, we choose the update rate of IMU as the primary rate of the filter, and then define a flag signal to indicate whether vision or height measurements have arrived. The state update process starts whenever the vision or height information comes. Finally, sensor failure or abnormality may occur during flight because of changing lighting conditions or insufficient features of the ground scene. Motivated by this, in this paper, we can make a judgment on whether the measured value is within a normal range by comparing the innovation covariance matrix estimated via the Kalman filter. Define the difference between measurement $\mathbf{z}_k$ and prediction $\hat{\mathbf{z}}_{k/k-1}$ as the innovation, i.e., $\varepsilon_k = \mathbf{z}_k - \hat{\mathbf{z}}_{k/k-1}$. So the innovation can be considered as a Gaussian white noise with covariance $\mathbf{C}_k\mathbf{P}_{k/k-1}\mathbf{C}_k^{\mathrm{T}} + \mathbf{R}_k$. Thus

$$\varepsilon_k^{\mathrm{T}}(\mathbf{C}_k\mathbf{P}_{k/k-1}\mathbf{C}_k^{\mathrm{T}} + \mathbf{R}_k)^{-1}\varepsilon_k \sim \chi^2(p),$$

where p is the dimension of ε_k. We can detect the abnormal data by considering a threshold value σ. If the discrepancy exceeds the threshold, i.e., $\varepsilon_k^{\mathrm{T}}(\mathbf{C}_k\mathbf{P}_{k/k-1}\mathbf{C}_k^{\mathrm{T}} + \mathbf{R}_k)^{-1}\varepsilon_k > \sigma$ holds, it is considered that this measurement is abnormal and should be abandoned.

6. Simulation and experimentation

6.1. Simulation experiment with synthetic data

This section presents comprehensive simulation results in which all the experimental data is generated synthetically to validate the effectiveness of the proposed visual–inertial estimation

system. The simulation setup process and corresponding results are presented.

6.1.1. Simulation setup

We utilized the Robotics Toolbox (RT)[1] for MATLAB by Peter Croke to generate sensor data for a quadcopter with user defined translational and rotational motion. RT provides a collection of functions that support fundamental algorithms in robotics like the representations of orientation in $SO(3)$, kinematics, dynamics, and trajectory generation, covering areas in model robotics, mobile ground robots as well as flying robots, i.e., quadcopter. RT has a history of about twenty years and the latest release of the toolbox is the tenth version with some changes and extensions to support the second edition of the well-known book [38].

In the toolbox, there is a closed-loop Simulink model of the typical quadcopter. The vehicle takes off and flies in a circle at a constant height. As shown in Fig. 10, the simulation adopts a quadcopter model and an inner-and-outer flight controller, in which the outer loop controls the position while the inner loop regulates the velocity and attitude. Given the reference trajectory and yaw angle, the Simulink model can compute the true 12-element states including the position, velocity, orientation, and orientation rate. The orientation and its rate are represented regarding yaw–pitch–roll angles according to the "ZYX" rotation sequences. Note that position and attitude are in the EFCF and the rates are presented in the ABCF. Thus the velocity information can be chosen as the ground truth in the simulation.

A single image with resolution of 752×480 pixels is taken from a flat scene with enough features, by setting an appropriate intrinsic matrix of the downward-looking camera, a large number of synthetic images are generated corresponding to the position and attitude of the quadcopter, using perspective projective model. During the process, GWN and constant biases are added to the true translational and rotational states to obtain the simulated measurements of sensors.

We adopt Computer Vision System Toolbox in the simulation to realize the vision process. The procedure to choose optimal matching points between two successive images is as follows: (1) Obtaining two successive images (MATLAB function *imread*); (2) Extracting speeded up robust features in each image (MATLAB function *detectSURFFeatures* and *extractFeatures*); (3) Matching feature points of the two images (MATLAB function *matchFeatures*), during this stage, we can use the function *showMatchedFeatures* to visualize the matching points in the image; (4) Calculating the corresponding velocity using the matching points according to the vision motion constraint; (5) Employing the MS algorithm to choose the optimal velocity and matching points iteratively. After obtaining the optimal matching points, the proposed estimation system can be executed using an LKF as discussed before.

6.1.2. Simulation results

The reference trajectory in the simulation is a horizontal circular motion with constant height and yaw angle given by

$$x^* = 3\sin(0.2t)$$
$$y^* = 3\cos(0.2t)$$
$$z^* = -4$$
$$\psi^* = 0$$

Fig. 11 shows samples from the generated images. It is clear that almost all points are matched accurately. And more accurate matching points will give more accurate velocity estimation. The simulation results are presented in Fig. 12. The true quadcopter states are indicated by solid green lines, the states estimated using our method are shown as red dotted lines while cyan dashed lines plot the IMU-only estimation. Fig. 12(a) is a snapshot during flight by the quadcopter simulation, and the marker on the ground plane is a projection of the quadcopter's centroid. The 2D trajectory of the quadcopter is depicted in Fig. 12(b), and it is evident that the distance will drift quite fast using IMU-only estimation, and the visual–inertial estimation has a good performance. As shown in Fig. 12(c), our proposed method can get a smooth and accurate velocity estimation. Also, Fig. 12(d) shows the change of number of matching points after using the MS algorithm, and it is shown that the number has decreased and the proposed optimal matching points algorithm is efficient. Note that there exist some instances when there are no matching points, but it has a little effect on the fusion results since the height and IMU information will take effect for a short time. Furthermore, the observations above are consistent with the observability analysis in Section 5. At last, we have computed the Root Mean Square (RMS) of the velocity and position estimate taking IMU-only, Vision-only, and Visual–inertial fusion into account. The corresponding results are summarized in Table 3.

Table 3
RMS results in simulation.

RMS	v_{x_b} (m/s)	v_{y_b} (m/s)	p_{x_b} (m)	p_{y_b} (m)
IMU-only	1.2483	1.1776	20.9462	16.4961
Vision-only	0.0830	0.0677	0.0642	0.0654
Visual–inertial	0.0478	0.04355	0.0194	0.0239

Table 4
Main specifications of the quadcopter M100.

Specifications	Quadcopter M100
Diagonal wheelbase	650 mm
Weight (with TB47D battery)	2355 g
Maximum takeoff weight	3600 g
Flight endurance	22 min
Maximum velocity (No GPS)	22 m/s
DJI intelligent Flight battery	LiPo 6S

6.2. Flight experiment with real data

After validating the proposed method on synthetic data, the testing is performed on flight experiment with real data. In this section, experimental results are presented to verify the effectiveness of the visual–inertial estimation system.

6.2.1. Experimental platform

The experimental flight platform is a commercial DJI Matrice 100 (M100) autonomous quadcopter from DJI (Da-Jiang Innovations Science and Technology Co., Ltd.) company. The off-the-shelf quadcopter is shown in Fig. 13. The main specifications of the M100 are listed in Table 4. M100 is a stable, flexible, and powerful development platform designed specially for various complex tasks for research, business or fun. All the computation is performed on the high-performance onboard embedded system (DJI Manifold), which contains an NVIDIA Tegra K1's 4-Plus-1 Quad-core ARM Cortex-A15 Processor. For the sake of safety, the function of the GPS-compass module is enabled to be forbidden when flying at a low altitude in no-fly zones.

M100 is also equipped with the DJI Guidance module, a revolutionary visual sensing system with a powerful processor, five integrated stereo cameras, and ultrasonic range finders. The Guidance module seamlessly integrates with the inertial sensors to provide accurate position, velocity, and obstacle measurements for M100 using a fusion of onboard sensors [39]. Note that only the downward-looking vision sensors (a camera and an ultrasonic range finder) of the Guidance are utilized in the experiment instead of other four directional sensors. Using the quadcopter platform,

[1] http://petercorke.com/wordpress/toolboxes/robotics-toolbox.

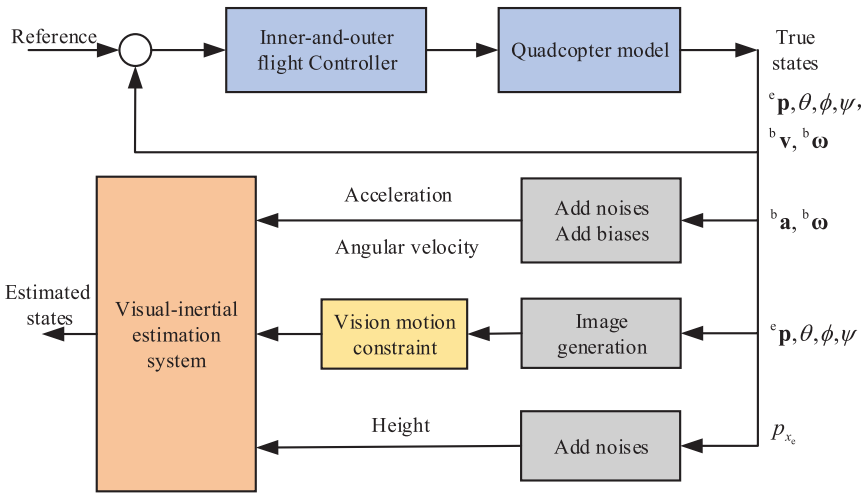


Fig. 10. Block diagram of the simulation.

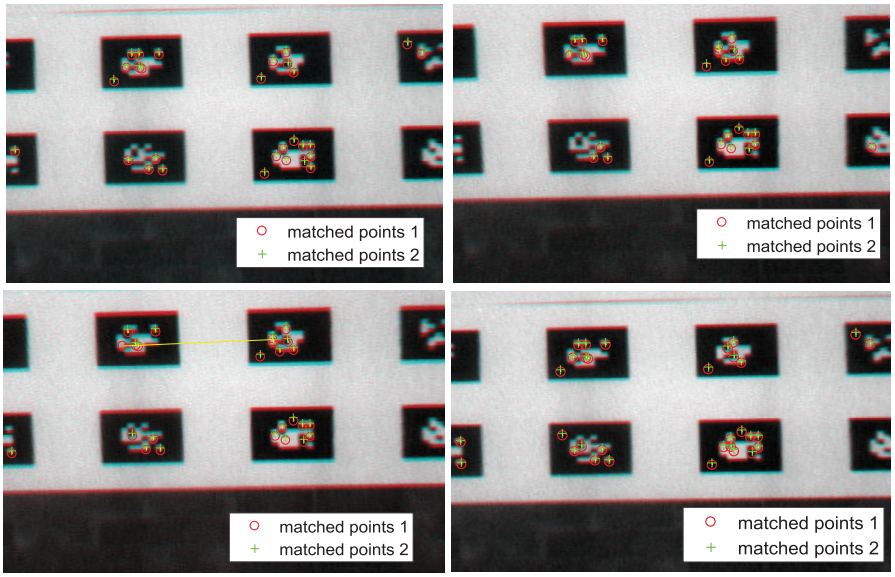


Fig. 11. Samples of the generated images in the simulation. The small red circles and yellow crosses represent the matching points between two images.

we can directly obtain IMU, height and image information. Besides, the Guidance can provide velocity using stereo vision algorithm as ground truth. We have already performed calibration in advance to obtain intrinsic camera matrix.

6.2.2. Experimental results

The flight experiments are conducted above a flat ground scene with colorful texture outdoors as shown in Fig. 14. The point correspondences between two successive images can be easily detected.

During the flight, the quadcopter is guided by a remote controller to fly in a square trajectory. The downward-looking camera captures the ground scene image at a fixed frequency of 20 Hz while the IMU and ultrasonic range finder also run at the same rate. It should be noted that the IMU data is synchronized with images inside the Guidance. The velocity can be estimated online using our proposed algorithm while the Guidance itself can give the ground truth at 10 Hz using stereo visual odometer. Also, we can record all the published sensor data through the Manifold to be dealt with offline using MATLAB. Also, the results of conventional method using optical flow and ultrasonic range finder are compared with that of the proposed method. The experimental flight results are depicted in Fig. 15. We have the following conclusions from the experimental results: (1) The proposed visual–inertial estimation can provide an accurate velocity w.r.t. the ABCF; (2) The position calculated by the estimated velocity is relatively accurate and drifts slowly such that multicopters can perform a short-term local navigation when the GPS signals are not available; (3) The proposed mean shift algorithm can detect the mismatching points and improve the accuracy of the estimation system; (4) The proposed method would outperform the optical flow method. Besides, the observations above are consistent with the observability analysis and simulation results. The processing time of the main functions is shown in Table 5, and it is clear that the image processing costs most of the total time. The efficiency of the proposed method can be improved by speeding up the image processing using GPU acceleration. However, in this paper, all the available data is recorded through ROS and all the processing of algorithms is run in MATLAB to demonstrate the effective of the proposed method. Furthermore, more experimental results are available at

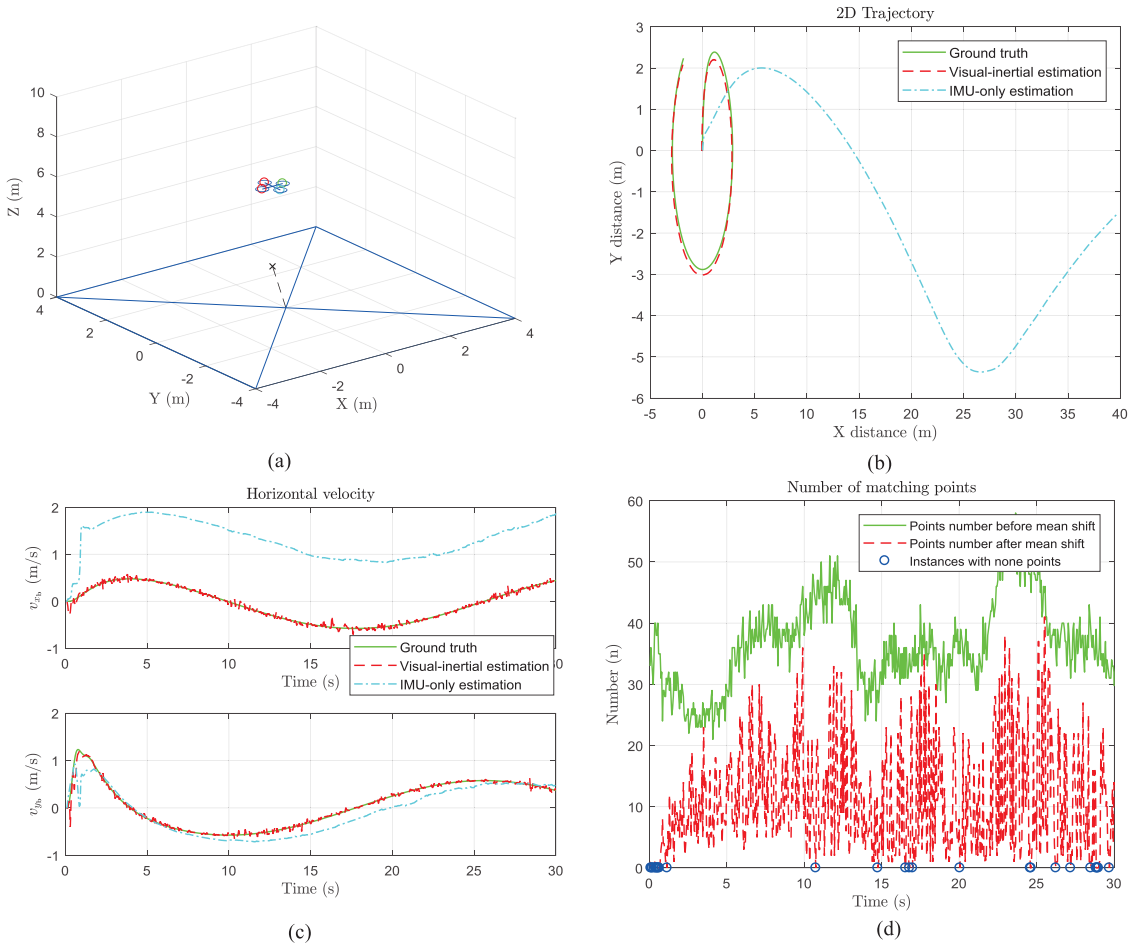


Fig. 12. Simulation results using the proposed method.

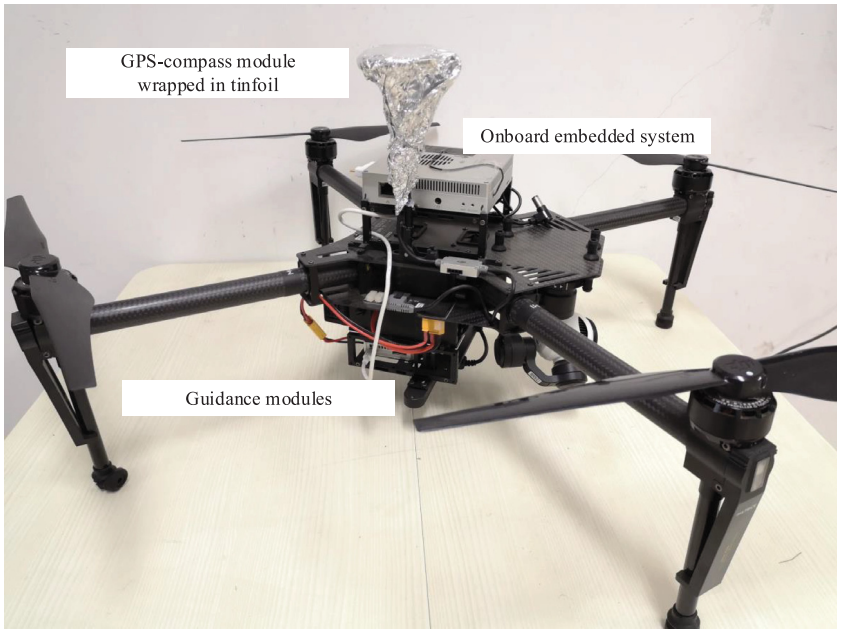


Fig. 13. The DJI Matrice 100 quadcopter.

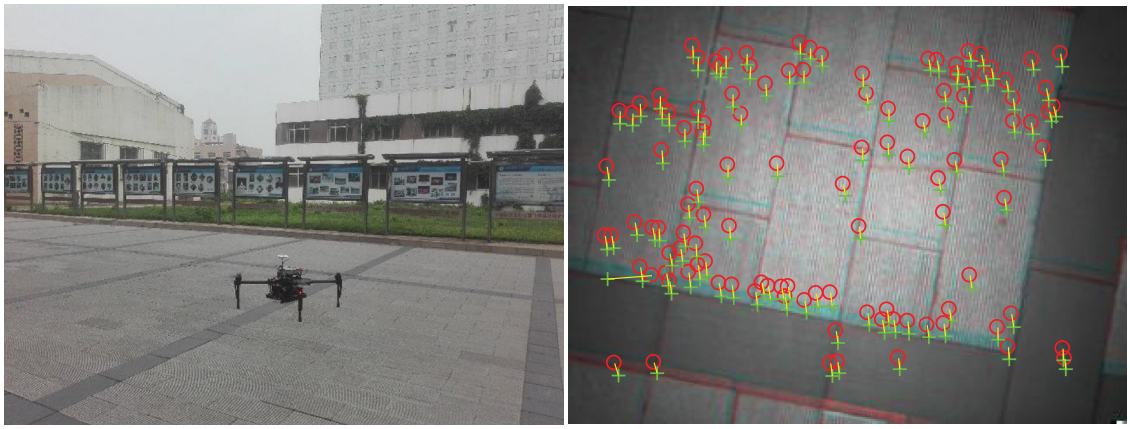


Fig. 14. The ground scene in the flight experiment. The size of the image captured is 320 × 240 pixels.

Table 5
Processing time of the main function using the MATLAB time.

Function	Total time (s)	Execute count (n)	Average time (ms)
Feature detection	21.729	1758	12.3
Feature extraction	58.671	1758	33.4
Feature matching	10.130	879	11.5
Mean shift	2.4184	879	2.75
Velocity by vision	0.864	879	0.983
LKF	0.125	879	0.142

"https://youtu.be/4J7sfCm_o0Y" and our Reliable Flight Control group website "http://rfly.buaa.edu.cn".

To further validate the effectiveness of the proposed method at larger velocity, more flight tests have been conducted in outside environment. The multicopter files freely by a remote controller, and the experimental results are depicted in Fig. 16. Results show that the proposed estimation still takes effect at larger speed nearly up to 4 m/s, but there could be some peaks at some time instances. Especially, it is observed that the estimation may become inaccurate when the speed is raised up. The inaccuracy is reasonable because the image quality becomes blurred and matching points are not easy to be extracted. It is noted that our proposed method relies much on accurate and enough point correspondences between successive images. Besides, the experiments are conducted without GPS signals, and only the attitude control using the remote controller is available. Thus, the attitude of the UAV is easily affected by external winds and the attitude may change a lot and therefore influences the accuracy of the ultrasonic range finder. The readings of the ultrasonic range finder may subject to zero if the measured distance is beyond the maximum effective range or there is no returned sound to be received when the attitude of the multicopter is large. Due to the unreliability and noises in the distance measurements, the height data is filtered in our method. Moreover, the supplementary experimental results of the proposed method at high speeds are available at: "https://youtu.be/OEvp4P-MyrE ".

7. Conclusions

This paper proposed a new visual–inertial method based on vision motion constraint to provide accurate velocity information merging the measurements from a monocular camera, an IMU, and an ultrasonic range finder, combined with a unique multicopter dynamic model. An efficient approach based on MS algorithm was developed to detect outliers and select optimal matching points between successive images. Observability analysis has shown that the proposed estimation system is observable with at least one feature correspondence detected when the multicopter is in hovering condition or straight and steady flight. Comprehensive simulation and experimental results have verified that the proposed estimation method works well in GPS-denied or confined environments with good texture or enough features. In our method, all we need is the onboard sensing without the aid of any external localization system, any artificial features, or any prior knowledge of the environment. Furthermore, the proposed method is valid with only one matching point detected. Even for the worst case without any feature detected, it will take effect during a short time interval. However, there exist some limitations in the estimation. First, it is known that there are few features for the poor-textured environment so the proposed method may not be effective in that case. Second, the ground is assumed to be flat which is a strong assumption, and it is also observed that the proposed method takes effect at a low altitude and slow velocity. Third, measurement delays are ignored for simplicity in our method. Last, we assume the system is nearly linear, but in fact, it is nonlinear if we choose the attitude as the unknown state. In future research, we will account for a situation where there are fewer features or the scene is not flat. Also, we need to study the characteristic of sensors further and handle time delays. Besides, we may try to consider the nonlinearity and utilize the extended Kalman filter or particle filter instead of the standard LKF. If possible, the estimated velocity can be employed in closed loop system.

Appendix A. Proof of Theorem 1

Based on the linear camera model (7) and (8), one has

$$s\bar{\mathbf{p}} = \mathbf{R}_{\mathrm{b}}^{\mathrm{c}}\left(\mathbf{R}_{\mathrm{e}}^{\mathrm{b}} \cdot {}^{\mathrm{e}}\mathbf{p} + \mathbf{T}_{o_{\mathrm{e}}}^{\mathrm{b}}\right) + \mathbf{T}_{o_{\mathrm{b}}}^{\mathrm{c}}. \tag{A.1}$$

According to Remark 1, the body frame and the camera frame are assumed to be the same in the derivation, thus one has

$$\mathbf{R}_{\mathrm{b}}^{\mathrm{c}} = \mathbf{I}_3, \mathbf{T}_{o_{\mathrm{b}}}^{\mathrm{c}} = \mathbf{0}_3. \tag{A.2}$$

Substituting Eq. (A.2) into Eq. (A.1) gives

$$s\bar{\mathbf{p}} = \mathbf{R}_{\mathrm{e}}^{\mathrm{b}} \cdot {}^{\mathrm{e}}\mathbf{p} + \mathbf{T}_{o_{\mathrm{e}}}^{\mathrm{b}}. \tag{A.3}$$

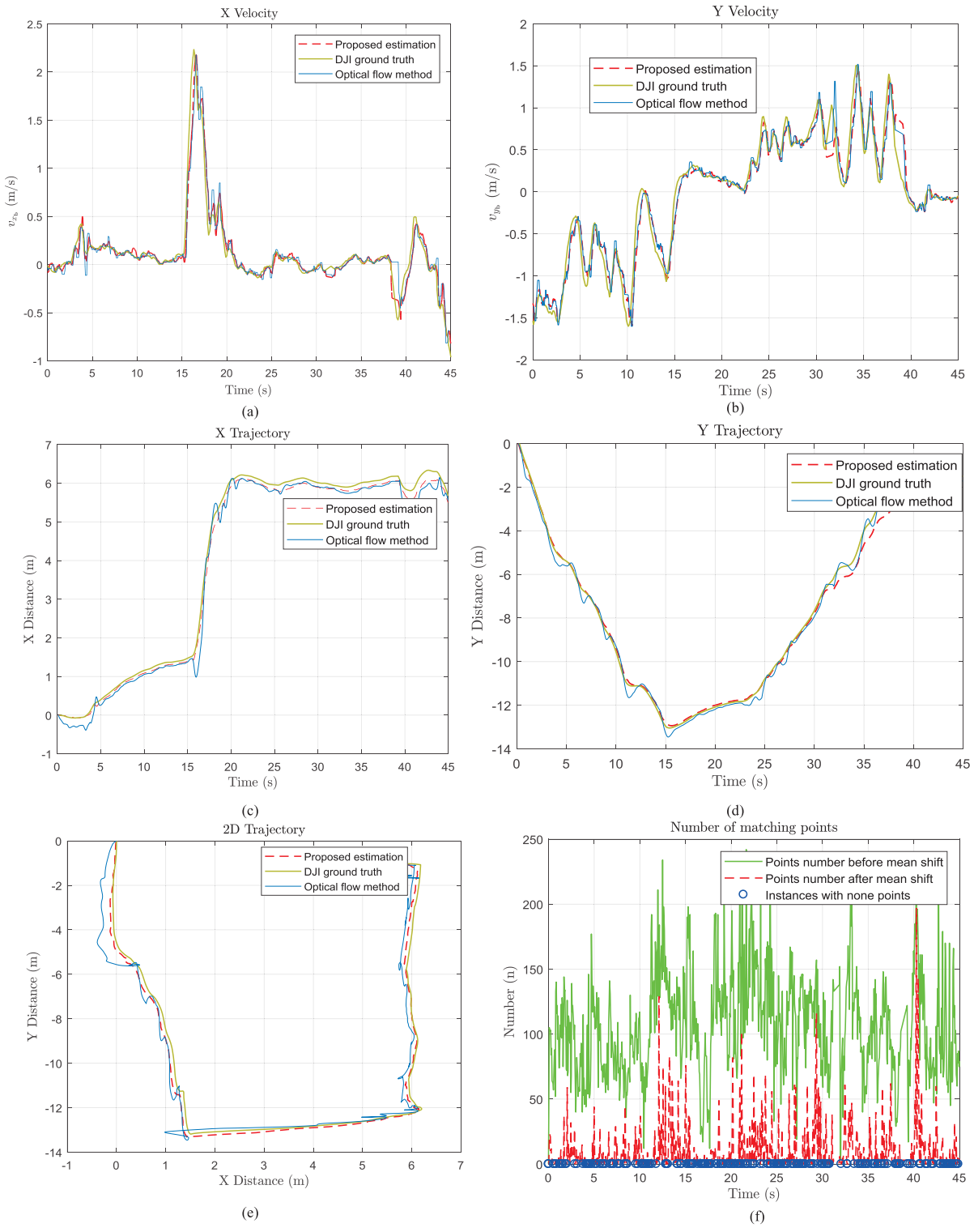


Fig. 15. Results of flight experiment outdoors.

Also, we have the relationship as

$$\mathbf{R}_{\mathrm{e}}^{\mathrm{b}} = \left(\mathbf{R}_{\mathrm{b}}^{\mathrm{e}}\right)^{\mathrm{T}}, \mathbf{T}_{o_{\mathrm{e}}}^{\mathrm{b}} = -\left(\mathbf{R}_{\mathrm{b}}^{\mathrm{e}}\right)^{\mathrm{T}}\mathbf{T}_{o_{\mathrm{b}}}^{\mathrm{e}}. \tag{A.4}$$

Based on Eqs. (A.3) and (A.4), we have

$$s\bar{\mathbf{p}} = (\mathbf{R}_{\mathrm{b}}^{\mathrm{e}})^{\mathrm{T}}\left({}^{\mathrm{e}}\mathbf{p} - \mathbf{T}_{o_{\mathrm{b}}}^{\mathrm{e}}\right). \tag{A.5}$$

According to Assumption 1, combined with Eqs. (1), (4) and (A.5), one has

$$p_{z_{\mathrm{b}}} = -\frac{d_{\mathrm{sonar}}\cos\theta\cos\phi}{\mathbf{e}_3^{\mathrm{T}}\mathbf{R}_{\mathrm{b}}^{\mathrm{e}}[x \quad y \quad 1]^{\mathrm{T}}}. \tag{A.6}$$

Based on Eq. (A.5), one has

$$p_{z_{\mathrm{b}}}^{k-1}\mathbf{R}_{k-1}\bar{\mathbf{p}}_{k-1} + \mathbf{T}_{k-1} = p_{z_{\mathrm{b}}}^{k}\mathbf{R}_k\bar{\mathbf{p}}_k + \mathbf{T}_k. \tag{A.7}$$

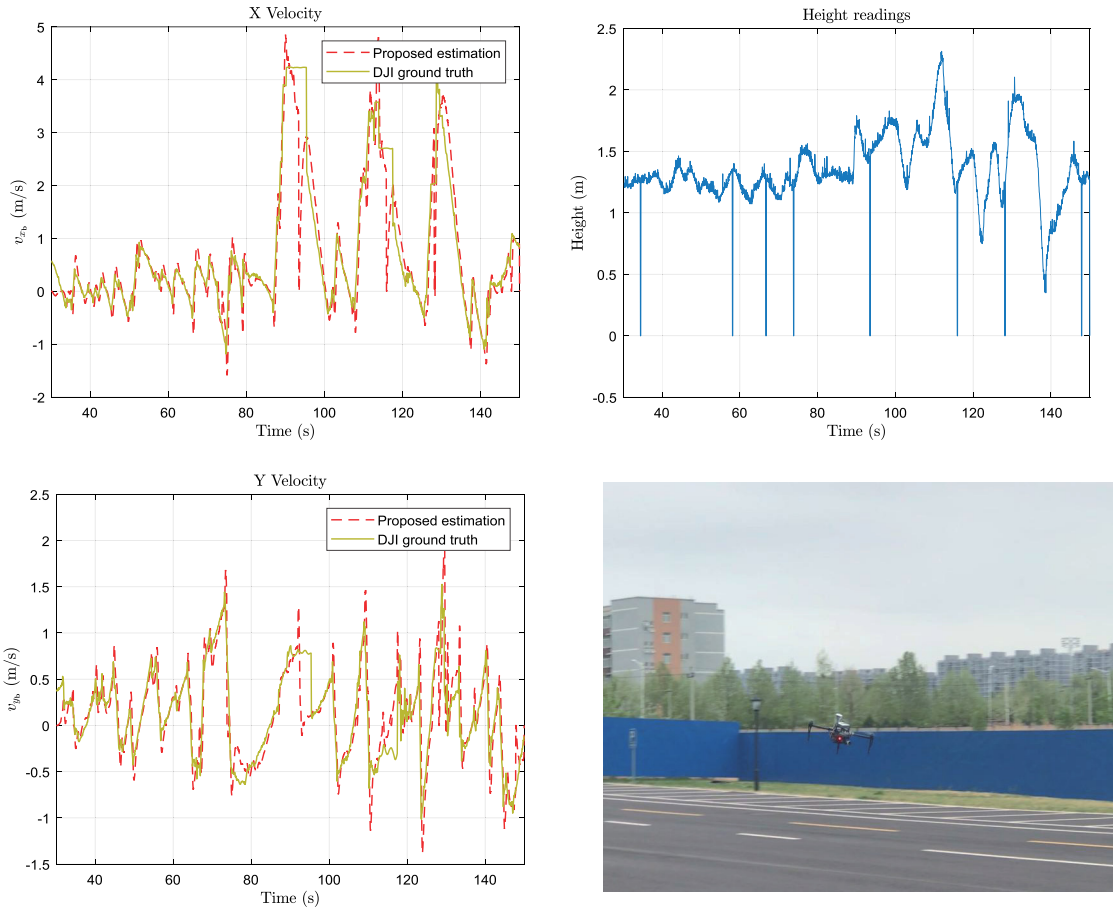


Fig. 16. Results of flight experiment outdoors at greater velocity.

According to Eq. (A.6), one has

$$\begin{aligned} p_{z_b}^{k-1} &= -\frac{h_{k-1}}{\mathbf{e}_3^T \mathbf{R}_{k-1} \bar{\mathbf{p}}_{k-1}} \\ p_{z_b}^{k} &= -\frac{h_k}{\mathbf{e}_3^T \mathbf{R}_k \bar{\mathbf{p}}_k}. \end{aligned} \tag{A.8}$$

Thus, from Eqs. (A.7) and (A.8), one has

$${}^e\mathbf{v}_k = \frac{\mathbf{T}_k - \mathbf{T}_{k-1}}{\Delta t} = \frac{p_{z_b}^{k-1} \mathbf{R}_{k-1} \bar{\mathbf{p}}_{k-1} - p_{z_b}^{k} \mathbf{R}_k \bar{\mathbf{p}}_k}{\Delta t}. \tag{A.9}$$

Recalling to the relationship $\dot{\mathbf{R}} = \mathbf{R}[\boldsymbol{\omega}]_\times$, we have

$$\mathbf{R}_k - \mathbf{R}_{k-1} = \mathbf{R}_{k-1}[{}^b\boldsymbol{\omega}_{k-1}]_\times \Delta t. \tag{A.10}$$

By multiplying Eq. (A.10) by $\mathbf{R}_{k-1}^T$ on both sides and rearranging it, one has

$$\mathbf{R}_{k-1}^T \mathbf{R}_k = \mathbf{I}_3 + [{}^b\boldsymbol{\omega}_{k-1}]_\times \Delta t. \tag{A.11}$$

Combined with Eqs. (A.9) and (A.11), Transforming the velocity to the ABCF results in

$$\begin{aligned} {}^b\hat{\mathbf{v}}_k &= \mathbf{R}_{k-1}^T \cdot {}^e\mathbf{v}_k \\ &= \frac{p_{z_b}^{k-1} \bar{\mathbf{p}}_{k-1} - p_{z_b}^{k} \mathbf{R}_{k-1}^T \mathbf{R}_k \bar{\mathbf{p}}_k}{\Delta t} \\ &= \frac{p_{z_b}^{k-1} \bar{\mathbf{p}}_{k-1} - p_{z_b}^{k} (\mathbf{I}_3 + [{}^b\boldsymbol{\omega}_{k-1}]_\times \Delta t) \bar{\mathbf{p}}_k}{\Delta t} \end{aligned}. \tag{A.12}$$

Combining Eqs. (A.8) and (A.12), we get the same expression of velocity as Theorem 1 illustrated.

Appendix B. Proof of Theorem 2

Since the simplified system (22) and (23) are time-invariant, we need to check the rank of the observability matrix

$$\mathcal{O}(\mathbf{A}, \mathbf{C}) = [\mathbf{C}^T \quad (\mathbf{CA})^T \quad \cdots \quad (\mathbf{CA}^7)^T]^T. \tag{B.1}$$

If the observability is of full rank, i.e., $\text{rank}\mathcal{O}(\mathbf{A}, \mathbf{C}) = 8$, the system is observable. First, consider the condition with vision measurement, the system is simplified to system (22), and the observability matrix of the simplified system (22) according to Eq. (B.1) is

$$\mathcal{O}(\mathbf{A}, \mathbf{C}_1) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}_{10\times 8}. \tag{B.2}$$

The zero rows of $\mathcal{O}(\mathbf{A}, \mathbf{C}_1)$ are omitted since they do not contribute to the rank of $\mathcal{O}(\mathbf{A}, \mathbf{C}_1)$. By examine the last four rows, there are two independent rows, thus, the rank of $\mathcal{O}(\mathbf{A}, \mathbf{C}_1)$ is eight. Hence, the system is observable, i.e., the velocity can be observable fusing the information of acceleration, height and vision measurement.

Similarly, consider the condition without vision measurement, the system is simplified to system (23), and the observability

matrix of the simplified system (23) according to Eq. (B.1) is

$$\mathcal{O}(\mathbf{A}, \mathbf{C}_2) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}_{6\times 8}. \tag{B.3}$$

It is calculated that the rank of $\mathcal{O}(\mathbf{A}, \mathbf{C}_2)$ is six, thus the system is unobservable, there are two unobservable quantities. In order to identify them, we need to obtain the null space of $\mathcal{O}(\mathbf{A}, \mathbf{C})$ that

$$\text{Null}(\mathcal{O}(\mathbf{A}, \mathbf{C}_2)) = \begin{bmatrix} 0 & 0 \\ 0 & -1 \\ -1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}_{8\times 2}. \tag{B.4}$$

The null space of the observability matrix suggests that the two unobservable quantities are the horizontal velocities v_{x_b}, v_{y_b}. The vertical velocity can be obtained by the difference of two height measurements.

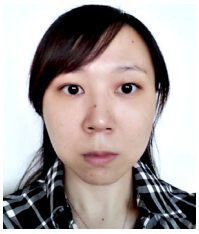
References

[1] R. Mahony, V. Kumar, P. Corke, Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor, IEEE Rob. Autom Mag. 19 (3) (2012) 20–32.
[2] T. Tomic, K. Schmid, P. Lutz, A. Domel, M. Kassecker, E. Mair, I.L. Grixa, F. Ruess, M. Suppa, D. Burschka, Toward a fully autonomous UAV: Research platform for indoor and outdoor urban search and rescue, IEEE Rob. Autom Mag. 19 (3) (2012) 46–56.
[3] I. Maza, K. Kondak, M. Bernard, A. Ollero, Multi-UAV cooperation and control for load transportation and deployment, in: Selected Papers from the 2nd International Symposium on UAVs, Reno, Nevada, USA June 8–10, 2009, Springer, 2009, pp. 417–449.
[4] R. Mebarki, V. Lippiello, B. Siciliano, Exploiting image moments for aerial manipulation control, in: ASME Dynamic Systems and Control Conference, Palo Alto, CA, 2013.
[5] B. Bethke, J.P. How, J. Vian, Multi-UAV persistent surveillance with communication constraints and health management, in: AIAA Guidance, Navigation, and Control Conference (GNC), 2009.
[6] Y. Huang, W.C. Hoffman, Y. Lan, B.K. Fritz, S.J. Thomson, Development of a low-volume sprayer for an unmanned helicopter, J. Agric. Sci. 7 (1) (2014) 148.
[7] N. Michael, D. Scaramuzza, V. Kumar, Special issue on micro-UAV perception and control, Auton. Robots 33 (1–2) (2012) 1–3.
[8] N. Michael, D. Mellinger, Q. Lindsey, V. Kumar, The grasp multiple micro-uav testbed, IEEE Robot. Autom. Mag. 17 (3) (2010) 56–65.
[9] S. Lupashin, A. Schöllig, M. Sherback, R. D'Andrea, A simple learning strategy for high-speed quadrocopter multi-flips, in: Proc. of IEEE 2010 International Conference on Robotics and Automation, 2010, pp. 1642-1648.
[10] J. Wendel, G.F. Trommer, Tightly coupled GPS/INS integration for missile applications, Aerosp. Sci. Technol. 8 (7) (2004) 627–634.
[11] B. Yun, K. Peng, B.M. Chen, Enhancement of GPS signals for automatic control of a UAV helicopter system, in: Proc. of IEEE 2007 International Conference on Control and Automation, 2007, pp. 1185-1189.
[12] N. Abdelkrim, N. Aouf, A. Tsourdos, B. White, Robust nonlinear filtering for INS/GPS UAV localization, in: 2008 16th Mediterranean Conference on Control and Automation, 2008, pp. 695-702.
[13] D. Floreano, R.J. Wood, Science, technology and the future of small autonomous drones, Nature 521 (7553) (2015) 460–466.
[14] S. Weiss, M.W. Achtelik, S. Lynen, M. Chli, R. Siegwart, Real-time onboard visual-inertial state estimation and self-calibration of mavs in unknown environments, in: Proc. of IEEE 2012 International Conference on Robotics and Automation, 2012, pp. 957–964.
[15] S. Weiss, M.W. Achtelik, S. Lynen, M.C. Achtelik, L. Kneip, M. Chli, R. Siegwart, Monocular vision for long-term micro aerial vehicle state estimation: A compendium, J. Field Robot. 30 (5) (2013) 803–831.
[16] D. Eberli, D. Scaramuzza, S. Weiss, R. Siegwart, Vision based position control for MAVs using one single circular landmark, J. Intell. Robot. Syst. 61 (1–4) (2011) 495–512.
[17] A. Borowczyk, D.T. Nguyen, A.P.V. Nguyen, D.Q. Nguyen, D. Saussié, J.L. Ny, Autonomous Landing of a Quadcopter on a High-Speed Ground Vehicle, J. Guid. Control Dyn. 40 (9) (2017) 1–8.
[18] M. Blösch, S. Weiss, D. Scaramuzza, R. Siegwart, Vision based MAV navigation in unknown and unstructured environments, in: Proc. of IEEE 2010 International Conference on Robotics and Automation, 2010, pp. 21–28.
[19] M. Achtelik, M. Achtelik, S. Weiss, R. Siegwart, Onboard IMU and monocular vision based control for MAVs in unknown in-and outdoor environments, in: Proc. of IEEE 2011 International Conference on Robotics and Automation, 2011, pp. 3056–3063.
[20] G. Nützi, S. Weiss, D. Scaramuzza, R. Siegwart, Fusion of IMU and vision for absolute scale estimation in monocular SLAM, J. Intell. Robot. Syst. 61 (1) (2011) 287–299.
[21] S. Shen, Y. Mulgaonkar, N. Michael, V. Kumar, Vision-based state estimation and trajectory control towards high-speed flight with a quadrotor. in: Robotics: Science and Systems, Vol. 1, Berlin, Germany, 2013.
[22] B. Herissé, T. Hamel, R. Mahony, F.-X. Russotto, Landing a VTOL unmanned aerial vehicle on a moving platform using optical flow, IEEE Trans. Robot. 28 (1) (2012) 77–89.
[23] P.-J. Bristeau, F. Callou, D. Vissiere, N. Petit, The navigation and control technology inside the ar. drone micro uav, IFAC World Congr. 44 (1) (2011) 1477–1484.
[24] D. Honegger, L. Meier, P. Tanskanen, M. Pollefeys, An open source and open hardware embedded metric optical flow cmos camera for indoor and outdoor applications, in: Proc. of IEEE 2013 International Conference on Robotics and Automation, 2013, pp. 1736–1741.
[25] L. Heng, D. Honegger, G.H. Lee, L. Meier, P. Tanskanen, F. Fraundorfer, M. Pollefeys, Autonomous visual mapping and exploration with a micro aerial vehicle, J. Field Robot. 31 (4) (2014) 654–675.
[26] V. Grabe, H.H. Bulthoff, P.R. Giordano, A comparison of scale estimation schemes for a quadrotor UAV based on optical flow and IMU measurements, in: Proc. of 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2013, pp. 5193–5200.
[27] S. Zhao, F. Lin, K. Peng, X. Dong, B.M. Chen, T.H. Lee, Vision-aided estimation of attitude, velocity, and inertial measurement bias for UAV stabilization, J. Intell. Robot. Syst. 81 (3–4) (2016) 531.
[28] A. Martinelli, Vision and imu data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination, IEEE Trans. Robot. 28 (1) (2012) 44–60.
[29] Q. Quan, Introduction to Multicopter Design and Control, Springer, Singapore, 2017.
[30] H. Bay, T. Tuytelaars, L. Van Gool, Surf: Speeded up robust features, in: European Conference on Computer Vision, Springer, Berlin, Heidelberg, 2006, pp. 404–417.
[31] K. Fukunaga, L. Hostetler, The estimation of the gradient of a density function with applications in pattern recognition, IEEE Trans. Inform. Theory 21 (1) (1975) 32–40.
[32] Y. Cheng, Mean shift, mode seeking, and clustering, IEEE Trans. Pattern Anal. Mach. Intell. 17 (8) (1995) 790–799.
[33] D. Comaniciu, P. Meer, Robust analysis of feature spaces: color image segmentation, in: 1997 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997, pp. 750–755.
[34] D. Comaniciu, V. Ramesh, P. and Meer, Real-time tracking of non-rigid objects using mean shift, in: Proc. of 2000 IEEE Conference on Computer Vision and Pattern Recognition, 2000, pp. 142–149.
[35] D. Comaniciu, P. Meer, Mean shift: A robust approach toward feature space analysis, IEEE Trans. Pattern Anal. Mach. Intell. 24 (5) (2002) 603–619.
[36] Y.A. Ghassabeh, A sufficient condition for the convergence of the mean shift algorithm with Gaussian kernel, J. Multivariate Anal. 135 (2015) 1–10.
[37] B.K.P. Horn, E.J. Weldon, Direct methods for recovering motion, Int. J. Comput. Vis. 2 (1) (1988) 51–76.
[38] P. Corke, Robotics, Vision and Control: Fundamental Algorithms In MATLAB® Second, Completely Revised, Vol. 118, Springer, 2017.
[39] G. Zhou, L. Fang, K. Tang, H. Zhang, K. Wang, K. Yang, Guidance: A visual sensing platform for robotic applications, in: Proc. of 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2015, pp. 9–14.

Heng Deng received the B.S. degree in Control Science and Engineering from Beijing Institute of Technology, Beijing, China, in 2015. He is currently a Ph.D. candidate in Guidance, Navigation and Control at School of Automation Science and Electrical Engineering from Beihang University. His main research interests are visual navigation, visual positioning system, multicopter design and control and data fusion.

Usman Arif received his B.SC. in Electrical Engineering from University of Engineering and Technology Taxila Pakistan and MS degree in Guidance Navigation and Control from Beihang University, Beijing China, in 2010 and 2017, respectively. His research interests include navigation, vision based navigation and robotics.

Qiang Fu received the B.S. degree in thermal energy and power engineering from Beijing Jiaotong University, Beijing, China, in 2009, and the Ph.D. degree in control science and engineering from Beihang University (formerly Beijing University of Aeronautics and Astronautics), Beijing, China, in 2016. He is currently a lecturer in the School of Automation and Electrical Engineering, University of Science and Technology Beijing. His main research interests include vision-based navigation and 3-D vision.



Zhiyu Xi received the B.Eng. degree in Control Science and Engineering from Harbin Institute of Technology, China in 2004. She then received M.Eng. and Ph.D. degrees in Automatic Control from The University of New South Wales, Australia in 2007 and 2011 respectively from the School of Electrical Engineering & Telecommunications, University of New South Wales, Australia. She is now an Associate Lecturer in the School of Electrical Engineering & Telecommunications, University of New South Wales. Her research interest includes: sliding mode control, fuzzy control, network control systems, hybrid systems, model predictive control and control applications.

Quan Quan received the B.S. and Ph.D. degrees in Control Science and Engineering from Beihang University (formerly Beijing University of Aeronautics and Astronautics), Beijing, China, in 2004 and 2010, respectively. He has been an Associate Professor at Beihang University since 2013. His main research interests include vision-based navigation and reliable flight control.

Kai-Yuan Cai received the B.S., M.S., and Ph.D. degrees in control science and engineering from Beihang University (formerly Beijing University of Aeronautics and Astronautics), Beijing, China, in 1984, 1987, and 1991, respectively. He has been a Full Professor at Beihang University since 1995. He is currently a Cheung Kong Scholar (Chair Professor), jointly appointed by the Ministry of Education of China and the Li Ka Shing Foundation of Hong Kong in 1999. His main research interests include software testing, software reliability, reliable flight control, and software cybernetics.