



REVIEW ARTICLE

Multi-rotor drone tutorial: systems, mechanics, control and state estimation

Hyunsoo Yang¹ · Yongseok Lee¹ · Sang-Yun Jeon¹ · Dongjun Lee¹

Received: 31 December 2016 / Accepted: 1 March 2017 / Published online: 16 March 2017
© Springer-Verlag Berlin Heidelberg 2017

Abstract We present a tutorial introduction to the multi-rotor unmanned aerial vehicles, often simply referred as drones. We first explain typical configuration, components and construction of the drones. We then provide basic kinematic and dynamic modeling of drones and their principle of flight. Some representative motion control techniques are then presented, which take into account the issue of underactuation of the drones. State estimation problem of the drones, that is crucial for their proper flying, yet, should be done only by using onboard sensors and their sensor fusion, is explained. Some emerging research directions requiring capability beyond typical drones are also mentioned.

Keywords Components · Control · Modeling · Multi-rotor drone · Sensor fusion

1 Introduction

Recently, multi-rotor unmanned aerial vehicles (UAVs), or often simply referred as “drones” have received substantial interests from the research community and the general public alike, due to their promise to extend the ground-bound 2D capability of mobile robots into the 3D space with relative ease of operation and affordability. This leads in the rapid growth of commercial drone markets, particularly those

Research supported in part by the Bio-Mimetic Robot Research Center (UD130070ID) funded by the Defense Acquisition Program Administration (DAPA) and by the Agency for Defense Development (ADD) of Korea.

✉ Dongjun Lee
djlee@snu.ac.kr

¹ Department of Mechanical and Aerospace Engineering and IAMD, Seoul National University, Seoul 151-744, Korea

for hobbyists with such successful commercial products as Parrot AR.Drone and Bebob, DJI Phantom and Mavic, 3D Robotics Solo, Syma Cheerwing, Yuneec Typhoon, etc. See Fig. 1 and Table 1.

In addition to these hobbyist drones, many companies, research groups and university labs around the globe are currently developing technologies to capture this capability of the drones and materialize new industrial applications. The most successful application of the drone to date has been aerial photography and cinematography (e.g., “Game of Throne” advertising film by Skynamic [1]). Other applications currently under development include parcel delivery (e.g., Amazon PrimeAir [2]), aerial mapping and geo-inspection (e.g., Trimble ZX5 [3]), vegetation analysis and pesticide spraying (e.g., PrecisionHawk [4]), inspection of large plants and infrastructure (e.g., [5, 6]), wild fire fighting and air quality monitoring (Scentroid Sampling Drone [7]), law enforcement and parking regulation, emergency communication or search and rescue aftermath of disaster (e.g., [8]), selfie-drone with follow-me capability (e.g., Air-Dog [9]), and even swarm flying for amusement park (e.g., 500 Drone Light Show by Intel [10]), to name just few. See Fig. 2 and Table 1.

This seemingly sudden blossoming of commercial drone market and their applications are in fact enabled by recent advancements in numerous background engineering technologies, namely materials (e.g., carbon fiber, engineering plastics, metal alloys), manufacturing and machining (e.g., laser cutting, 3D printing), battery (e.g., Lithium–Polymer), actuation (e.g., BLDC motors), micro-electro-mechanical system (MEMS) sensors, global positioning system (GPS), camera and image processing, onboard computing and communication, and algorithms (e.g., control, sensor fusion, computer vision, simultaneous localization and mapping (SLAM)).

Fig. 1 Examples of hobbyist drones. **a** Parrot AR.Drone [11], **b** 3D Robotics Solo [12], **c** DJI Phantom 4 [13], **d** DJI Mavic Pro [13], **e** Syma Cheerwing [14], **f** Yuneec Typhoon [15], **g** Cheerson CX-10C [16], **h** SKEYE Nano [17]



This paper aims to provide a tutorial introduction to these multi-rotor drones with focus on the systems (i.e., components and construction), dynamics modeling, motion control and state estimation. Given the vastness of results for the drones, here, we provide rather a high-level and non-exhaustive overview, while referring readers to the cited references for more details. The rest of the paper is organized as follows. Hardware, components and construction of the typical drone are explained in Sect. 2. System kinematic and dynamic modeling and basic flight principle are presented in Sect. 3. Some representative techniques for motion control of the drones and their state estimation via

multi-sensor fusion are then introduced in Sects. 4 and 5, respectively. Section 6 concludes this paper with some comments on emerging research directions for the drones.

2 Configurations and components

2.1 Configurations and frames

The (second) definition of the term “drone” in the Cambridge Dictionary reads as follows: *an aircraft without a pilot that is controlled by someone on the ground, used especially as*

Table 1 Specifications of some commercially-available drones

Usage	Model	# of rotors	Size (m)	Weight (kg)	Payload (kg)	Flight time (min)	Battery (mAh)
Hobbyist	Parrot AR.Drone [11]	4	0.52(L, W)	0.42	–	12	3000
	3D Robotics Solo [12]	4	0.46(L,W)	1.99	–	16	5200
	DJI Phantom 4 PRO [13]	4	0.247(L,W)	1.388	–	30	6000
	DJI MAVIC [13]	4	0.237(L,W)	0.743	–	27	3830
	Syma Cheerwing [14]	4	0.315(L,W)	–	–	–	500
	Yuneec Typhoon [15]	6	0.52(L), 0.457(W)	1.695	0.255	25	5400
	Cheerson CX-10C [16]	4	0.042(L,W)	0.015	–	2.5–4	120
	SKEYE Nano [17]	4	0.04(L,W)	0.014	–	3–4	120
	GTF AG-10 [18]	4	1.169(L, W)	9.8	13.2	9–12	16000
Agriculture	ATI AgBOT [19]	4	1.02(L,W)	4.7	19.6	26	13000
	DJI AGRAS MG-1 [13]	8	1.471(L,W)	8.8	15.2	24	24000
	DJI M600 PRO [13]	6	1.669(L), 1.518(W)	9.5	6	32	27000
Video/mapping/monitoring	DJI Spreading Wing [13]	8	1.045(L,W)	4.4	6.6	15	15000
	Trimble ZX5 [3]	6	0.85(L,W)	–	5	15	13200
	Scentroid DR300 [7]	4	0.41(L,W)	1.36	–	25	5200
Transportation	EHANG 184 [20]	8	3.989(L), 4.024(W)	240	100	25	–
	Volocopter VC200 [21]	18	7.6(L,W)	450	120	20	–

L Length, W width

a hobby. There are three types of drones: fixed-wing drones, helicopter drones and multi-rotor drones. The fixed-wing drone typically has the same configuration as airplane, where the front propeller or engine provides thrust, while control is done via control surfaces. Since the gravity is resisted by wing, this fixed-wing drone typically boasts fast speed, long operation time and flight range. Yet, it cannot hover and fly backward, rendering some applications mentioned in Sect. 1 impossible to achieve (e.g., aerial photography, follow-me, inspection). On the other hand, the helicopter drones can fly omni-directionally by relying on collective and cyclic pitches of the main rotor, while the reaction moment is compensated for by the tail rotor. However, this helicopter drone requires rather complex swash-plate mechanism and, furthermore, is difficult to control due to complicated translational-rotational control coupling (e.g., sideway drifting due to tail rotor force needs to be counteracted by tilting drone body).

The multi-rotor drones, instead, can rotate and lift independently simply by changing the speed of the typically co-linearly aligned rotors, thereby, rendering the flight control problem much easier than that of helicopter drones, while providing hovering and omni-directional flying capability. This ease of control, we believe, is the reason why the drones are so popular, particularly among hobbyists. Of course, there is a price to pay for this: The typical flight time of multi-rotor drones is rather short (5–20 min) with the payload often limited (see Table 1) due to the poor

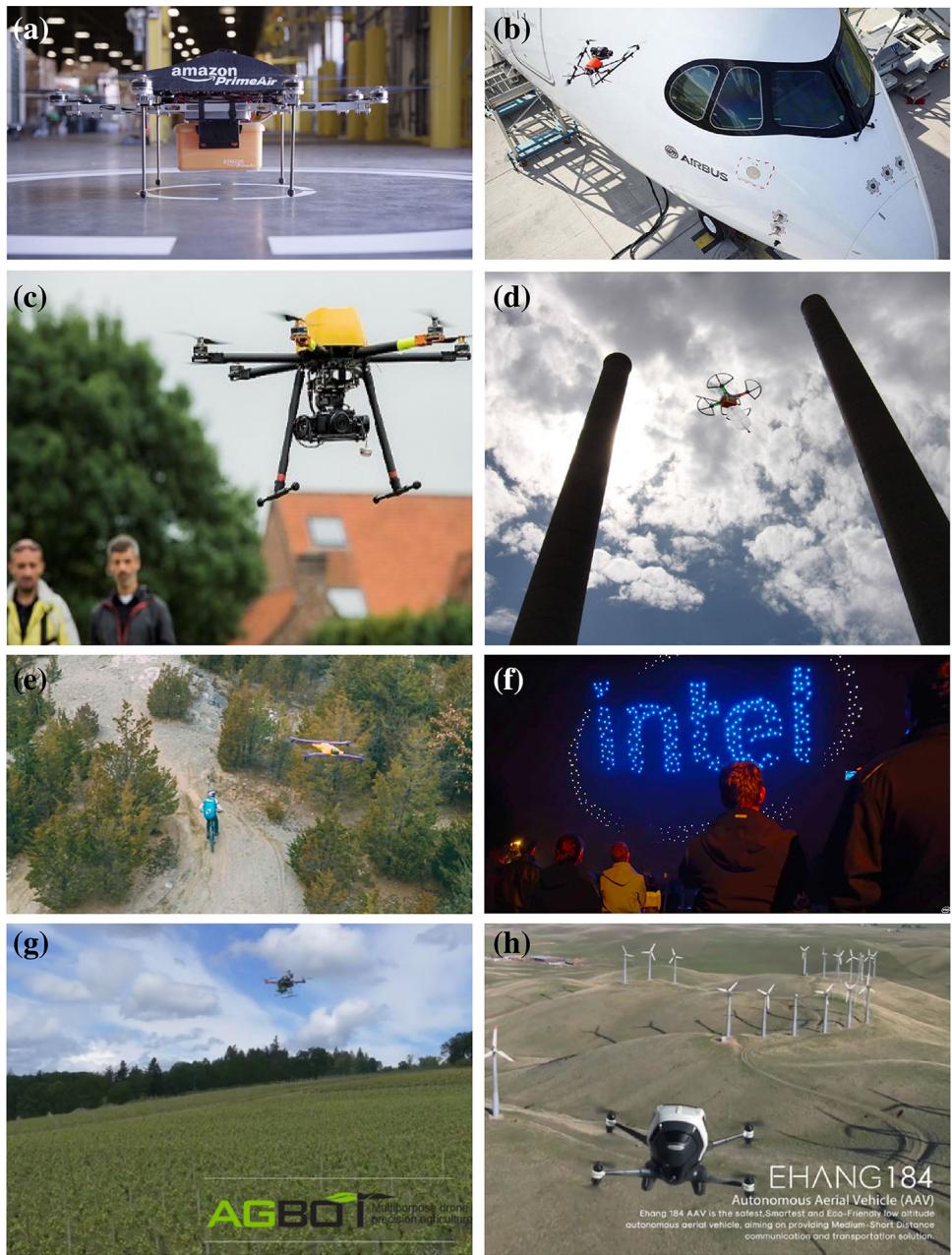
energy efficiency stemming from continuous overcoming the gravity by the rotors with smaller (i.e., less efficient) size props.

In this paper, we focus on this multi-rotor drone. For typical drones, all the rotors are aligned vertically to maximize gravity overcoming. Depending on the number of rotors, it is called tricopters, quadcopter, hexacopter, octocopters, etc. It is also typical that, due to the reason as elucidated in Sect. 3.1, even number of rotors is symmetrically attached to the drone frame—see Fig. 1. The frame of the drones should be as light, rigid and tough as possible for energy efficiency [24], control simplicity [25], and ruggedness [26,27], for which many frames adopt carbon fiber composite materials. Typically, large drones are heavy due to their frame weight, and hobbyist drones have lightweight for their price competitiveness. Some drones, which perform such tasks as watering plants, transporting people, have large size and heavyweight. See Fig. 3.

2.2 Rotor and ESC (electronic speed controller)

A rotor consists of a motor and a prop. Multiple rotors are symmetrically and co-linearly attached to the drone frame and generate thrust force, which then rotates and lifts the drone body. Typical diameter of props used for the drone is around 8–9 in. Key specifications of the motors include: operating voltage (typically 11.1 or 14.8 V with three cell or four cell Li-Po battery), kV (i.e., rpm/V: typically around

Fig. 2 Examples of drone applications. **a** Amazon PrimeAir [22], **b** Airplane inspection (Airbus) [23], **c** Trimble ZX5 [3], **d** Scentroid DR300 Sampling Drone [7], **e** Airdog with “follow-me” function [9], **f** 500 Drone Light Show by Intel [10], **g** ATI AgBOT [19], **h** EHANG 184 [20]



800–1000 rpm/V), maximum ampere (typically around 5–15 A), and ampere during hovering (typically around 3–8 A). The weight of a single rotor is usually around 12–20 g [12, 13, 15]. The thrust force is known to be proportional to the pitch angle, quadratic to the rotor speed, and quartic to the diameter of the prop, although there are many other subtle factors affecting the thrust force such as material, blade number, shape of propeller. On the other hand, the ESC assumes pulse width modulation (PWM) and produces AC current output to control the rotor speed, with typical running rate of around 400 Hz, weight of around 20–30 g, and maximum current rating of around 20 A.

2.3 Power source

The Lithium polymer (LiPo) battery is most widely used for drones for its moderate power-to-weight ratio and large discharge rate (e.g., ≥ 100 A) with the typical current-providing capacity of around 2000–6000 mAh with the weight of around 200–400 g for usual three or four cell batteries. Choosing the correct size of battery under the capacity-weight trade-off is one of the most challenging task of drone design. The battery is also currently posing perhaps the most formidable issue for the drones to be really useful for a wide range of applications, as the current state-of-art bat-

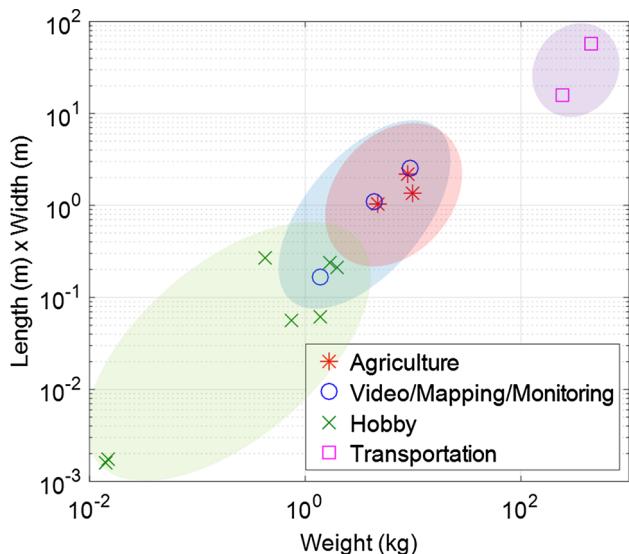


Fig. 3 Weight versus size (i.e., length \times width) of commercially-available drones in Table 1

tery technology only allows for fairly limited flying time (i.e., typically less than 5–20 min) and limited payload—see Table 1.

2.4 Sensors

Crucial for the drone to autonomous fly is the ability to estimate its orientation only with onboard sensing. This is necessary to correctly guide the thrust force, thereby, flying in a desired direction. For this, the key sensing element adopted virtually for all commercial drones is the MEMS IMU (inertial measurement unit), which consists of an accelerometer and a gyroscope, both measuring their respective quantities in the drone body frame. The accelerometer then provides the information of inclination of the drone body w.r.t. the gravity vector with its proper sensing bandwidth typically slower than 30 Hz due to the presence of noise, whereas the gyroscope the angular velocity expressed in its body frame in high update rate. Fusing the accelerometer and gyroscope information, the pitch and roll orientation motion of the drone can be estimated. Yaw angle (i.e., heading direction) can be estimated by integrating the gyroscope measurement, which yet resulting in the drift problem due to the gyroscope bias. To overcome this, a magnetometer (or compass) can be used in a manner similar to the accelerometer-gyroscope sensor fusion.

The drone only with a gyroscope is sometime called 3-axis drone, which can only be stabilized by directly commanding the angular velocity, thus, difficult to control, yet, agile, thus, adopted for some racing drones. The drone with a gyroscope and an accelerometer is called 6-axis drone, which allows a remote user or computing station to control the pitch and

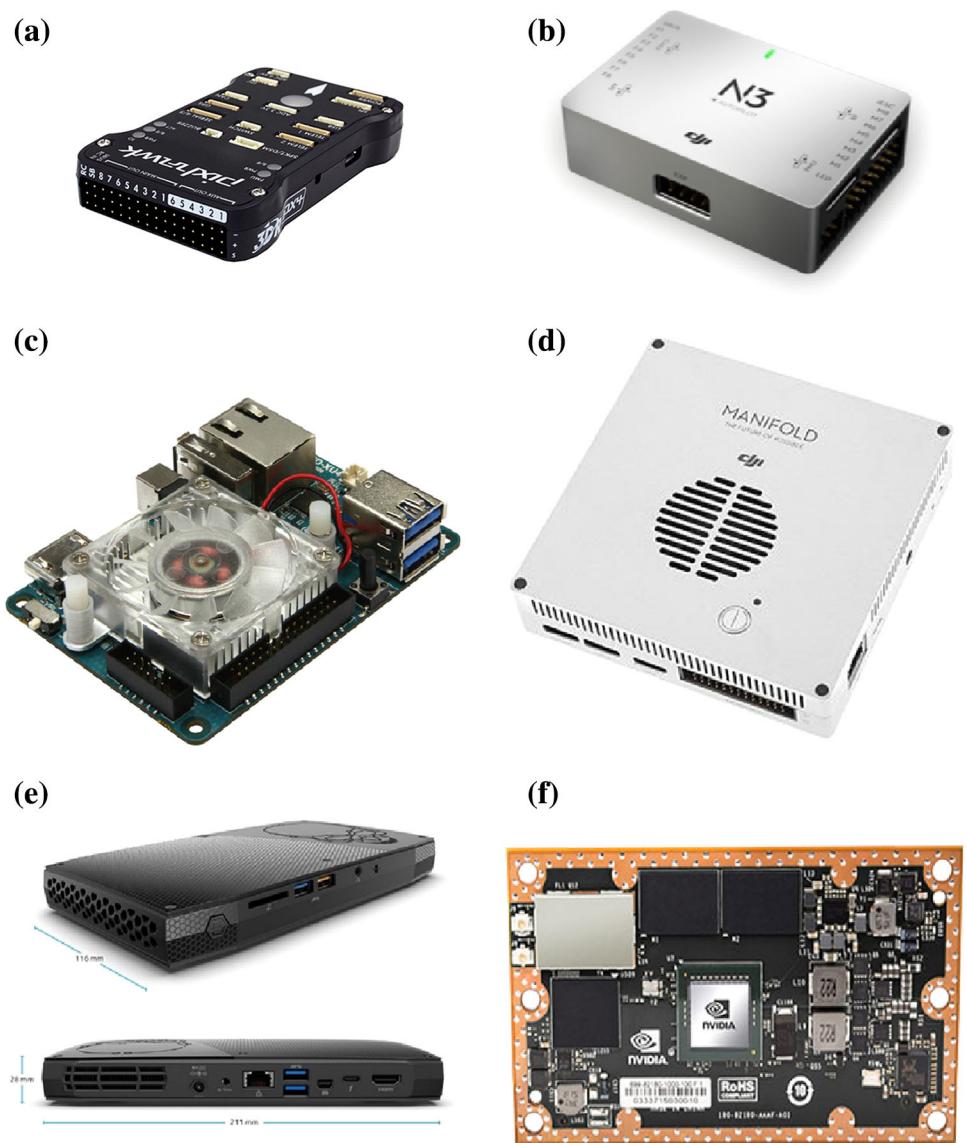
roll motions, yet, heading direction control is impossible. The 9-axis drones are those with the IMU and a magnetometer, which allows for direct angle control, thus, easiest to control and adopted by many state-of-the-art commercial drones (e.g., DJI Phantom 4 and Mavic Pro). More detailed description on using this IMU (and compass) for drone state estimation is given in Sect. 5.1.

On top of attitude estimation, for autonomous hovering and flying, it is also necessary for the drone to estimate its own velocity (for hovering) or to localize its position in 3D. This translation estimation problem becomes trivial if some MOCAP system (e.g., VICON[®], Optitrack[®]) is available, which can provide millimeter-level accuracy and 200–300 Hz refreshing update rate by using multiple IR (infrared) cameras. This MOCAP system, however, is expensive and not usually available outdoor. Another popular sensors for 3D localization are LiDAR or RGB-D camera, both of which are not so suitable for the drones, as the LiDAR sensor is usually too heavy and power hungry for the drones, whereas RGB-D camera is often too near-ranged and not so robust particularly for outdoor flying.

For outdoor 3D localization, GPS (global positioning system) is widely used, which estimates the position and velocity of drones using received signals from the GPS satellites [28]. At least four satellites should be within line of sight (LOS) for proper working of the GPS, which relies on triangulation with the information of traveled times for position estimation and Doppler effect for velocity estimation. With the precision (P) code not open to public [29], the GPS typically exhibits the (x, y) -localization error of around $\pm 2\text{--}5$ m, with the velocity error around 10 cm/s and the update rate of 1–5 Hz. The altitude information from the GPS is also usually not so accurate. Due to these accuracy issues, the GPS signal is often fused with the accelerometer sensor (to reduce translation error), and barometer (with resolution of around 20–30 cm) or sonar sensor (range limited typically up to 5–10 m) for height estimation in the framework of extended Kalman filtering (EKF)—see Sect. 5.2. Of course, Differential GPS (DGPS) [29] or RTK (real-time kinematic) GPS can provide centimeter or sub-centimeter level localization accuracy, which yet requires extra base stationary GPS station and also is typically too expensive to be adopted for many commercial drones.

However, this GPS is not working when enough number of GPS satellites is not in LOS (e.g., close to buildings, in valleys, under the bridges) or for indoor environments. Cameras can be used for such instances, which is lightweight, yet, requires rather high computing power, thus, only recently starts being adopted by commercial drones with specially-designed image processing apparatus, or accompanied with extra onboard PC on top of flight control unit (FCU) for in-lab research drones (see Sect. 2.5), both being detrimental for drones already under fairly tight power budget due

Fig. 4 Examples of onboard PCs for drones. **a** Autopilot Pixhawk [34], **b** DJI N3 [35], **c** Hardkernel Odroid XU4 [36], **d** DJI MANIFOLD [35], **e** Intel NUC [37], **f** Nvidia Jetson TX1 [38]



to the limited battery capacity (see Sect. 2.3). Perhaps, one of the earliest and simplest adoption of camera for drone autonomous flying is that by Parrot AR.Drone, in which a downward monocular camera is used with sonar sensor to estimate the drone velocity via the optical flow technique for autonomous hovering [30]. Recently, many groups attempt to use a monocular camera for 3D SLAM localization and fuse that information with IMU so that it can be used for flight control loop [31–33]. Stereo camera is less used, since its depth information is often not so suitable for outdoor flying. Its applicability of indoor flying, however, is promising though.

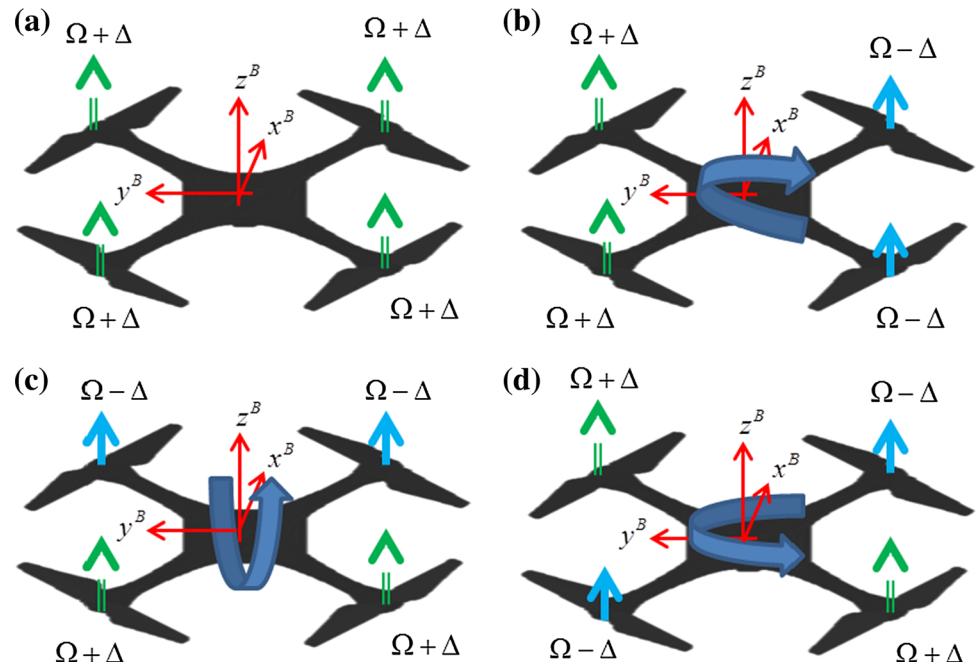
2.5 Flight control unit (FCU) and onboard PC

The drones need onboard computing to run algorithms to estimate its states (e.g., attitude, position, velocity) while

also computing its control action to fly. Most of the drones designed for general manual or simple way point flight utilize an unprogrammable flight control unit (FCU) to perform these estimation and control tasks. One of the most popular FCUs is Pixhawk [34] (see Fig. 4a), which includes an IMU and a barometer, provides state estimation by fusing IMU, barometer, and GPS information, and computes low-level motor control command. This Pixhawk is also compatible with a wide range of drone configurations. Recently, DJI also introduced their own N3 [35], which are designed to use their components (Fig. 4b).

However, more computing power is required to perform heavier algorithms such as image processing and precise autonomous flight mission. For this, we may add additional onboard PC on the drones. Hardkernel Odroid XU4 [36] in Fig. 4c has moderate computation power (2 GHz Octa core CPUs and 2 GB RAM), light weight and low price, thus, can

Fig. 5 The four basic flight modes of drones. **a** Vertical lift, **b** rolling rotation, **c** pitching rotation, **d** yawning rotation



be a cost-effective choice for lightweight drones. DJI Manifold [35] in Fig. 4d also has quad-core 2 GHz CPUs and 2 GB RAM, yet, allows only the DJI SDK-based software and components. Intel also announces their own mini-PC, NUC [37] (Fig. 4e), which provides an identical environment as a typical PC, thus, more convenient to work with than other embedded onboard PCs. Nvidia Jetson TX1 [38] in Fig. 4f has also been frequently adopted to run vision algorithms, which has 1 TFLOP/s 256 core GPU and supports CUDA.

2.6 Wireless communication

Wireless communication is necessary for drone operation, for only telemetry purpose, or for real-time data sending and receiving to/from a ground control station (GCS). XBee is one of the most popular communication choices for the drones, as it has rather long communication range (typically 50–100 m [39]) and lightweight. XBee communication is also known to have the advantage of low power consumption. The key weakness of the XBee communication is its low communication bandwidth (250 kbit/s). WiFi communication is another frequent choice for the drones, which utilizes the frequency range of 2.4 or 5 GHz, with its communication rate typically around of 54 Mbit/s, much higher than that of the XBee communication. The communication range of WiFi is also about 100 m suitable for most drone flying applications, although, depending on ambient conditions, its range may be shorter than this with its reception also sometimes possibly becoming precarious. Typical drones are equipped with their remote controller, whose communication frequency (2.4 GHz) may interfere with the wireless

communication. Recently, long-term evolution (LTE) is also utilized in few works for long-distance operation (≥ 100 m) or robust/fast communication for, e.g., real-time video surveillance (e.g., [40, 41]).

3 Flight principle and system modeling

3.1 Principle of flight

The basic movements of drones are the vertical lift, roll, pitch, and yaw movements as shown in Fig. 5, where the double-lined arrows indicate the increasing of the rotor rotation speed (i.e., $w_i + \Delta$), whereas the single-lined arrows indicate their decreasing (i.e., $w_i - \Delta$). As explained in Sects. 2.2 and 3.2, the rotor thrust is quadratically proportional to the rotor speed. The two adjacent rotors are also rotating in opposite direction using “mirrored” props to balance reaction moments generated by the rotor rotations. Although our explanation here is mainly for quadcopter drones, it is equally applicable for other types of drones (e.g., hexacopter, octacopter, etc.), as long as their rotors are co-linear all facing vertically.

For the vertical lift, all the rotor speeds are changed by the same amount; thus, the drone can only fly vertically while keeping its attitude constant (Fig. 5a). For the roll or pitch motions, two opposing rotors, respectively, increase and decrease their rotor speeds by the same amount, thereby, can only incur the attitude change while keeping the thrust force intact (Fig. 5b, c). For the yaw motion, a pair of two opposing rotors and the other pair of two opposing rotors collectively increase and decrease their speeds, respectively,

thereby, inducing yaw rotating, all the while keeping the thrust force again to be constant (Fig. 5d).

This quadcopter, in addition to being dynamically unstable, is under “under-actuation” i.e., to forward/side fly, it should tilt to guide the thrust force to the desired direction. This under-actuation stems from the fact that the quadcopter possesses 6-DOF, yet, its actuation only 4-DOF (i.e., four rotors). This under-actuation property degrades quality of photo-taking particularly when the drone change their motion frequently. For this, a gymbal is usually necessary for high-quality photo or movie shooting. Note also that, if we only have three rotors, the attitude of the drone cannot be arbitrarily controlled (e.g., uncontrolled yaw rotation) [42]. Due to this reason, typical tricopters adopt one extra rotor-tilting motor to counter-act yaw spinning.

3.2 System modeling

The dynamics of multi-rotor drone can be written by the following Newton–Euler rigid body dynamics equation [43]:

$$m\ddot{x} = -\lambda Re_3 + mge_3 + f_e \quad (1)$$

$$J\dot{\omega} + \omega \times J\omega = \tau + \tau_e, \quad \dot{R} = RS(\omega) \quad (2)$$

where $x \in \mathbb{R}^3$ is the quadrotor’s center-of-mass position represented in the inertial frame $\{O\} := \{N^o, E^o, D^o\}$, $m > 0$ is the mass, $\lambda \in \mathbb{R}$ is the thrust force, $R \in \text{SO}(3)$ represents the rotation of the body frame $\{B\} := \{N^B, E^B, D^B\}$ w.r.t $\{O\}$, $f_e \in \mathbb{R}^3$ is the tool interaction force represented in $\{O\}$, g is the gravitation constant, and $e_3 = [0, 0, 1]^T$ is the basis vector specifying the down direction. Also, for (2), $J \in \mathbb{R}^3$ is the body-fixed rotational inertia, $\omega := [\omega_1, \omega_2, \omega_3]^T \in \mathbb{R}^3$ is the angular velocity of $\{B\}$ relative to $\{O\}$ represented in $\{B\}$, $\tau, \tau_e \in \mathbb{R}^3$ are, respectively, the torque input and the external torque acting at the quadrotor’s center-of-mass (defined below) all represented in $\{B\}$, and $S(\omega)v := \omega \times v$ for any $v \in \mathbb{R}^3$. For typical drone flying, $(f_e, \tau_e) \approx 0$.

The attitude dynamics (2) can also be described by using Euler angle $\eta := [\phi \theta \psi]^T$, where ϕ, θ , and ψ are the roll, pitch, and yaw angles. That is, we can then write (2) in terms of η by using the parameterization of R w.r.t. η s.t.,

$$R = \begin{bmatrix} s\psi c\theta & c\psi s\theta s\phi - s\psi c\phi & c\psi s\theta c\phi + s\psi s\phi \\ s\psi c\theta & s\psi s\theta s\phi + c\psi c\phi & s\psi s\theta c\phi - c\psi s\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix}$$

and the following Jacobian relation between w and $\dot{\eta}$ s.t.,

$$w = T\dot{\eta}, \quad T := \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\phi & s\phi c\theta \\ 0 & -s\phi & c\phi c\theta \end{bmatrix} \quad (3)$$

where $c(\cdot) := \cos(\cdot)$ and $s(\cdot) := \sin(\cdot)$. This Euler angle representation, however, becomes singular when $\theta \rightarrow \pm\pi/2$.

In (1)–(2), the control inputs are the thrust force λ and the reaction torque τ generated by the rotors as explained in Sect. 3.1. For the quadcopter, this (λ, τ) can often be written by the following simplified relation:

$$\begin{pmatrix} \lambda \\ \tau_1 \\ \tau_2 \\ \tau_3 \end{pmatrix} = \underbrace{\begin{bmatrix} c_T & c_T & c_T & c_T \\ 0 & -d \cdot c_T & 0 & d \cdot c_T \\ -d \cdot c_T & 0 & d \cdot c_T & 0 \\ c_Q & -c_Q & c_Q & -c_Q \end{bmatrix}}_{=: \Gamma} \begin{pmatrix} w_1^2 \\ w_2^2 \\ w_3^2 \\ w_4^2 \end{pmatrix}$$

where w_i is the rotor speed, c_T is the aerodynamic coefficient between the thrust and the rotor speed, c_Q is the coefficient relating the yaw reaction moment by air resistance to the rotor speed, and d is offset from center of mass to each rotor. Using this mapping, we can decode the desired control input (λ, τ) into w_i , which is then achieved via low-level rotor speed control through the ESC (see Sect. 2.2). The rotors also produce gyroscopic force, which yet is often neglected as their magnitude is typically only less than 10% of those modeled above [44]. Similar mappings from the rotor speed to the control input can be also obtained for other types of multi-rotor drones (e.g., hexacopter, octacopter, etc.), for which, if the number of rotor is larger than four with $\Gamma \in \mathbb{R}^{4 \times n}$ being a fat matrix (i.e., $n > 4$), multiple solutions of w_i can exist to generate the same control input $(\lambda, \tau_1, \tau_2, \tau_3)$, that may be exploited to combat with, e.g., individual rotor failure and also often solved via norm-optimization for energy efficiency.

4 Motion control techniques

The key issue of the multi-rotor drone controls is the under-actuation as stated in Sect. 3.1, i.e., although their position control is usually necessary, all we can directly control is their attitude (see Fig. 5). It is also an inherently unstable system, necessitating a feedback control for its stabilization. There are two kinds of approaches for this problem of drone motion control while addressing the under-actuation: (1) inner-loop attitude control and outer-loop translation control utilizing time-scale separation property of faster attitude dynamics (2) and slower translation dynamics (1) of typical drones (e.g., [45–47]); and (2) unified translation-attitude control considering both the translation dynamics (1) and the attitude dynamics (2) at the same time (e.g., [48–50]).

One of the inner/outer-loop control schemes based on the time-separation property was presented in [47], which is summarized here. Let us first write $R(\eta)$ s.t.,

$$R(\eta) = R_{e_3}(\psi)R_{e_2}(\theta)R_{e_1}(\phi)$$

Then, from (1), we can have

$$R_{e_3}^T(\psi)m\ddot{x} = mge_3 - R_{e_2}(\theta)R_{e_1}(\phi)\lambda e_3$$

whose last row reads s.t.,

$$m\ddot{x}_3 = mg - \cos(\phi) \cos(\theta)\lambda$$

This then suggest the following thrust control

$$\lambda = -\frac{m}{c\phi c\theta}[-g + \ddot{p}_3 + k_{dp}(\dot{p}_3 - \dot{x}_3) + k_{pp}(p_3 - x_3)]$$

which ensures local exponential stability of $p_3 - x_3$ if $\cos\phi \cos\theta \neq 0$. On the other hand, the first two rows of (1) are given by

$$m \begin{pmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{pmatrix} = -\lambda \underbrace{\begin{bmatrix} \cos\phi \cos\psi & \sin\psi \\ \cos\phi \sin\psi & -\cos\psi \end{bmatrix}}_{=:Q(\phi,\psi) \in \mathbb{R}^{2 \times 2}} \begin{pmatrix} \sin\theta \\ \sin\phi \end{pmatrix}$$

where Q is always invertible as long as $\cos\phi \neq 0$. This then suggests the following desired attitude command

$$\begin{pmatrix} \sin\theta_d \\ \sin\phi_d \end{pmatrix} = \frac{mQ^{-1}}{-\lambda} \begin{pmatrix} \ddot{p}_1 + k_{dp}(\dot{p}_1 - \dot{x}_1) + k_{pp}(p_1 - x_1) \\ \ddot{p}_2 + k_{dp}(\dot{p}_2 - \dot{x}_2) + k_{pp}(p_2 - x_2) \end{pmatrix}$$

achieving local exponential stability of $(p_1 - x_1, p_2 - x_2)$ as well. The yaw angle command ψ_d is redundant and can be set arbitrarily (e.g., $\psi_d = 0$). The control can be further simplified for near hovering flying by $\sin\theta \approx 0$ and $\cos\phi \approx 1$ [46].

Once this attitude angle command $\eta_d = [\phi_d, \theta_d, \psi_d]^T$ is identified, many control techniques are applicable to attain that, since the attitude dynamics (2) is fully-actuated (e.g., local exponentially stable stabilization using η -parameterization [47] or even almost global attitude stabilization control in SO(3) [51]). For some commercial drones, angular velocity control mode is supported with some low-level control embedded. For this case, achieving the attitude angle η_d becomes even simpler, as the problem becomes a first-order kinematic control problem. This inner/outer-loop control is rather simple to implement and turns out to adequately performing for near hovering or slow flying. Yet, for aggressive flying, for which the coupled translation and attitude dynamics should be fully taken into account, its performance is often lacking. For this, we would need the unified translation-attitude control as mentioned above.

One of such unified translation-attitude control for multirotor drones, which are typically relying on backstepping like ideas to address the under-actuation issue, is the passivity-based control technique presented in [50], which is derived

for the “mixed” drone (i.e., control input is (λ, w) instead of (λ, τ)). More specifically, the backstepping control of [50] is derived from

$$\lambda Re_3 = -m\ddot{x}_d + mge_3 + b\dot{e}_x + ke_x + v_e$$

where $e_x := x - x_d$ and v_e is the control generation error due to the under-actuation. Then, the close-loop dynamics with this control is given by

$$m\ddot{e} + b\dot{e} + ke = -v_e$$

where if $v_e = 0$, $e_x \rightarrow 0$ exponentially. Now, by augmenting the perturbed Lyapunov of the above dynamics with v_e and also incorporating the attitude kinematics in (2) through $\dot{\psi}_e$, the backstepping control of [50] is designed s.t.,

$$\begin{pmatrix} \lambda w_2 \\ -\lambda w_1 \\ \dot{\lambda} + a\lambda \end{pmatrix} = R^T[v + \alpha v + \gamma(e + \epsilon e)]$$

where α, γ, ϵ are control gains and

$$v = -m\ddot{x}_d - b\ddot{x}_d + k\dot{e}_x + \frac{b}{m}(-\lambda Re_3 + mge_3)$$

This backstepping control then guarantees exponential stability to the desired trajectory x_d [50]. A backstepping position control for “dynamic” drones with the control input (λ, τ) was also presented in [48].

The above feedback control may be used in conjunction with flatness-based motion planning [52], that is, for the multi-rotor drones, if we choose (x, ψ) as the flat output, all the other states and control inputs can be expressed by them and their derivatives. Other control results include integral backstepping [53], sliding mode control [54], adaptive and velocity-field control [50, 54]. Of course, most of commercial products and drones still rely on PD/PID controls, and, for this, see [45], which contains nice exposition on this. Needs still exist for control design to combat with wind gust, actuator failure, model uncertainty, body flexibility, etc.

5 State estimation

One of the key challenges for drone flying is how to estimate its state using only onboard sensors that also need to be affordable to be commercially viable. Attitude estimation is necessary for all the drones, even for pilot-flying ones, as it defines the core part for proper working of flight control of the drone. Translation estimation is necessary for autonomous flying drones, as, without it, the drone cannot know its own and desired locations, thus, cannot fly to the target position. Due to this importance, the state estimation problem of the

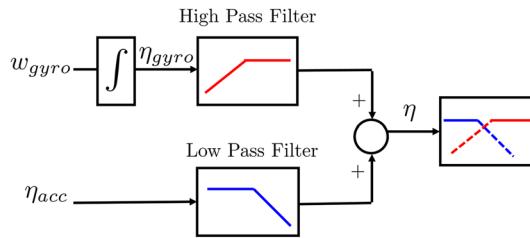


Fig. 6 Complementary filter

drone has been studied extensively (e.g., [32, 55–60]), with some of them introduced below.

5.1 Attitude estimation via IMU

As stated above and also in Sect. 4, the key requirement for drone flying, regardless if it is remotely-controlled or autonomous flying, is accurate estimation of the attitude. The widely used way for this is to use an IMU and a compass, and fuse their information (e.g., [61–63]). As stated in Sect. 2.4, the accelerometer and the compass can provide angle information with their bandwidth yet typically limited slower than 10–20 Hz due to the presence of noise. On the other hand, the gyroscope provides angular velocity information, that is, typically adequate for high-frequency bandwidth, yet, suffers from the drift problem, which is related more to low-frequency bandwidth. This leads into the idea of complementary filtering as depicted in Fig. 6, where low pass component of gyroscope and high pass component of accelerometer or magnetometer are opportunistically fused together to estimate the drone attitude (e.g., [63, 64]).

The configuration space of the attitude, however, is $\text{SO}(3)$, which is not a vector space, thus, the vector-based approach of complementary filtering is not often working properly for large attitude change. To address this Lie group structure of attitude motion, a nonlinear complementary filter for $\text{SO}(3)$ was proposed in [55], for which we introduce the sensor models as follows: (1) the gyroscope model is given by

$$w_m = w + b_w + n_w \quad (4)$$

where w, w_m are the true and measured angular velocity, b_w is the bias, and n_w is the noise; (2) the accelerometer measurement is modeled s.t.,

$$a_m = R^T(\ddot{x} - ge_3) + b_a + n_a \quad (5)$$

where \ddot{x} is the true acceleration in the inertial frame and a_m is the measured acceleration including the gravity $-ge_3$ and b_a, n_a are the bias and noise, all in the body frame; and (3) the compass measurement model is written by

$$m_m = R^T m_0 + n_m \quad (6)$$

where m_o is the Earth magnetic field expressed in the inertial frame, m_m is its measurement expressed in the body frame, and n_m is the noise. The nonlinear complementary filter of [55] is then given by

$$\dot{\hat{R}} = \hat{R}(S(w_m - \hat{b}) + k_P S(w_{mes}))$$

$$\dot{\hat{b}} = -k_I w_{mes}$$

$$w_{mes} := \sum_{i=1}^2 k_i v_i \times \hat{v}_i$$

where \hat{v} are the estimated quantities, v_i is the measured sensor value defined by $v_1 = m_m / |m_m|$, $v_2 = a_m / |a_m|$, k_P, k_I, k_i are the gains.

5.2 Pose estimation via EKF

To estimate the pose (i.e., attitude and position) of the drone, extended Kalman filtering (EKF) is one of the most frequently used approaches due to its computational simplicity [32, 60, 65]. A difficult aspect for this is that the configuration space of the drone is again not a vector space, but $\text{SE}(3)$ with the Lie group structure. We first start by defining the state of the drone following [32] s.t.,

$$x = [p^T \ v^T \ q^T \ b_g^T \ b_a^T]^T \in \Re^{16} \quad (7)$$

where p, v are the position and velocity, q is the unit quaternion and b_\star is the biases of the gyroscope and the accelerometer. The state propagation can then be modeled by

$$\begin{aligned} \dot{q}(t) &= \frac{1}{2} \mathcal{Q}(w_m(t) - b_g) q(t), \quad \dot{p}(t) = v(t) \\ \dot{v}(t) &= C(q(t))(a_m(t) - b_a(t)) + g \\ \dot{b}_g(t) &= 0_{3 \times 1}, \quad \dot{b}_a(t) = 0_{3 \times 1} \end{aligned}$$

where $\mathcal{Q}(\cdot)$ is a mapping of angular velocity for the product between quaternions and $C(q)$ is the rotation matrix corresponding to the quaternion q .

Instead of formulating the EKF directly for the state x defined above, here, we do so for its error state as done in [32] s.t.,

$$\tilde{x} := [\tilde{p}^T \ \tilde{v}^T \ \delta\theta^T \ \tilde{b}_g^T \ \tilde{b}_a^T]^T \in \Re^{15}$$

where $\delta\theta$ is the multiplicative attitude error. This error state formulation provides advantage in computing minimal representation of covariance and avoiding issues about unity constraint of the quaternion. The EKF propagation equation can then be written as follows:

$$\begin{aligned}\dot{\tilde{x}} &= \begin{bmatrix} 0_3 & I_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & -C(\hat{q})S(\hat{a}) & 0_3 & -C(\hat{q}) \\ 0_3 & 0_3 & -S(\hat{w}) & -I_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_3 \end{bmatrix} \tilde{x} \\ &+ \begin{bmatrix} 0_3 & 0_3 & 0_3 & 0_3 \\ -C(\hat{q}) & 0_3 & 0_3 & 0_3 \\ 0_3 & -I_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & I_3 \end{bmatrix} n =: F\tilde{x} + Gn\end{aligned}$$

where $n := [n_a^T \ n_g^T \ n_{b_g}^T \ n_{b_a}^T]^T \in \mathbb{R}^{12}$ are the noise vector with zero-mean Gaussian noise.

Using the EKF propagation model described above, a priori state estimation can be computed. The correction step can then be performed by utilizing some sensor outputs, which can be chosen depending on task objectives and available sensor configurations. For instance, we may use GPS (for localization), vision-based SLAM (for localization) or optical flow (for velocity estimation) as explained in Sect. 5.3, barometer or sonar sensor (for scale information), LiDAR or RGB-D camera, or some combination of these sensors. One of the frequently used combinations for outdoor autonomous flying is GPS-compass combination, for which the compass sensor model is given by (6), whereas the GPS can be modeled by the following simplified model s.t.,

$$p_m(t) = p(t) + n_{\text{gps}}(t)$$

where p is the true position, p_m is the GPS measurement and $n_{\text{gps}}(t)$ is a Gaussian noise [66, 67]. Of course, more complex modeling is also possible for the GPS sensing (e.g., bias, random walk [68], etc.). For this GPS-compass combination, we then have the following linearized output equation

$$\begin{aligned}\tilde{y} &= \begin{bmatrix} 0_3 & 0_3 & S(C^T(\hat{q})m_0) & 0_3 & 0_3 \\ I_3 & 0_3 & 0_3 & 0_3 & 0_3 \end{bmatrix} \tilde{x} + \begin{bmatrix} n_m \\ n_{\text{GPS}} \end{bmatrix} \\ &=: H\tilde{x} + w\end{aligned}$$

which is then used together with the above EKF propagation equation to estimate the pose of the drone in SE(3).

5.3 Vision-based state estimation

For the position correction in the EKF mentioned earlier, GPS is the most prevailing sensor due to its low price, size and robust accessibility for outdoor usage. However, GPS has such inherent limitations as slow update rate (e.g., 1–5 Hz), low accuracy (± 2 –5 m) as well as position drift [69, 70]. Moreover, it is not properly working when the number of

satellites in LOS is not enough (e.g., close to building, under bridges, in valleys, etc.) or cannot be relied on at all for the indoor flying (Sect. 2.4). To overcome these limitations of GPS, camera-based vision algorithms are actively investigated as a alternative or complimentary sensor for drone. This vision-based approach can provide better estimation accuracy and faster update rate (e.g., 20–60 Hz) than GPS, thus, can be complementarity utilized with GPS for outdoor flying [60]. Further, it can be relied on for the indoor flying, where GPS fails [71].

Since the drone is virtually always equipped with IMU, many researches investigated how to fuse the IMU and vision sensor to estimate the state of drone. This is often referred as visual-inertial navigation system (VINS), which can be categorized as loosely-coupled VINS or tightly-coupled VINS. The loosely-coupled VINS executes vision-only state estimation and fuse it with IMU readings via fusion algorithm such as the EKF as explained above, while the tightly-coupled VINS estimates the state by fusing the IMU readings and visual information from the camera all together. Below, we first introduce the two major vision-only estimation techniques for the drones: optical flow (OF) and SLAM, which can be used for the loosely-coupled VINS. We then briefly explained the tightly-coupled VINS.

One of the earliest and perhaps simplest vision-based estimation is optical flow technique, where the velocity of the drone is estimated by using the difference between two sequential visual scenes (obtained via the partial derivative of images) along with the angular velocity and height estimates of the drone, under the assumption that the scene is planar and static. This optical flow approach was adopted for commercial drones (e.g., AR.Drone [30] and Bebop of Parrot), applied for some drone autonomous flying results (e.g., [72, 73]) and also commercialized as an open-source optical flow system (i.e., PX4FLOW, [74]), which integrates a CMOS image sensor, a gyroscope and a sonar sensor all in one package. Despite of their low computational cost, from being based on the relative motion of sequential scenes, this optical flow technique is more suitable for autonomous hovering, not for autonomous flying, although it can improve the overall system robustness when used together with other estimation techniques.

Thus, many recent results for vision-based autonomous drone flying has been focusing to use the SLAM techniques [71]. Typical SLAM algorithms consist of the following two main tasks: mapping of an unknown environment and simultaneously determining the robot's pose in that map. The core of any SLAM algorithms is to establish correspondence among the feature points in different visual scenes and to apply epipolar geometry to those to recover the camera pose in 3D from 2D images. Through these processes, SLAM algorithms incrementally build a consistent map of an unknown environment while simulta-

Fig. 7 Autonomous flying experiment using PTAM-GPS sensor fusion, with PTAM enhancing performance and robustness of GPS-based localization [60]: down-view camera scene with tracking features (left); and visualization of drone position trajectory and map points (right)

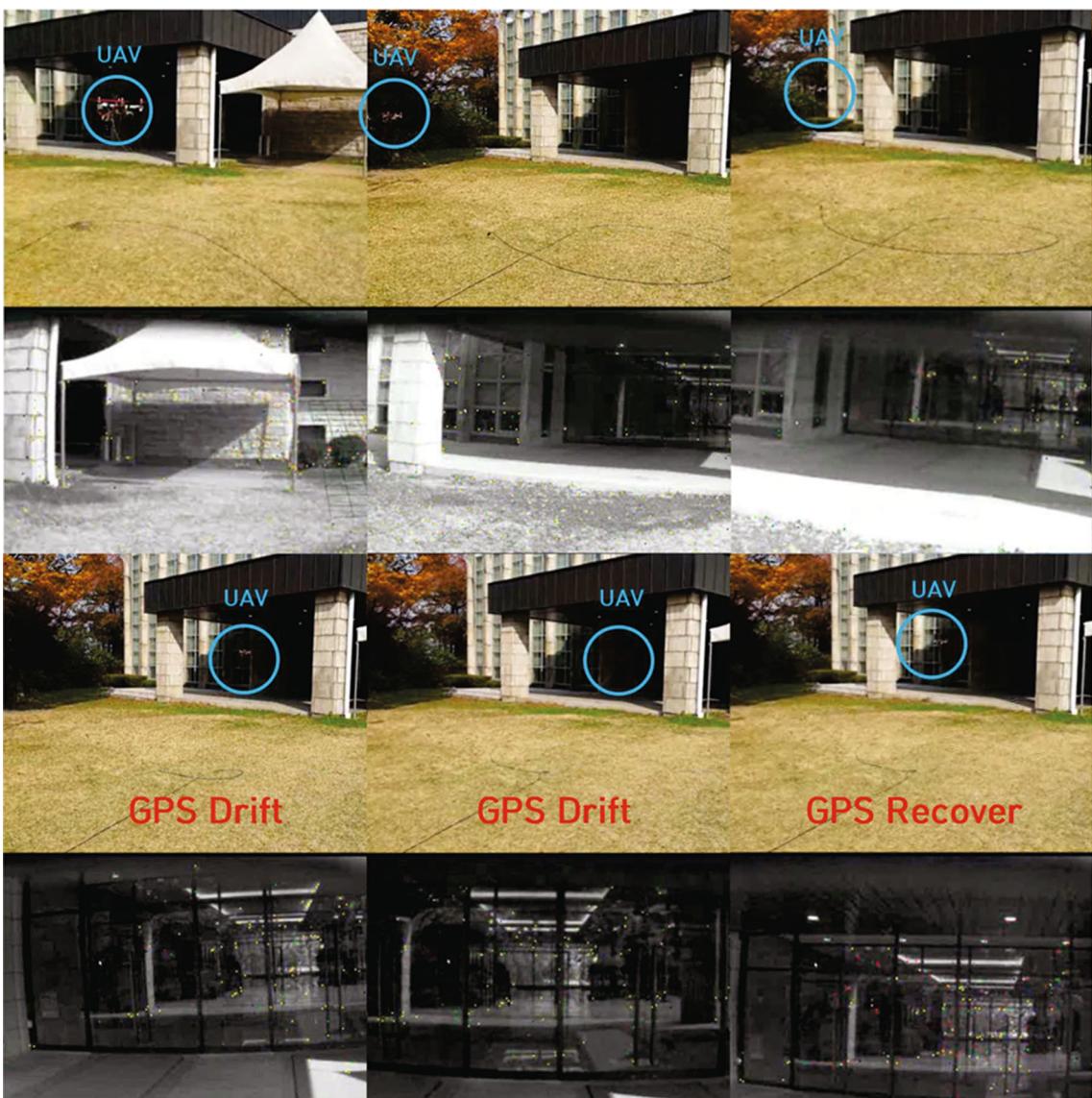
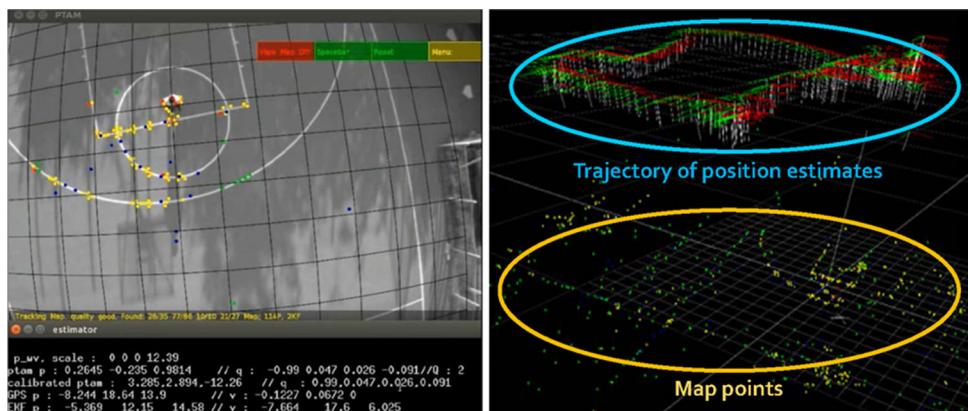


Fig. 8 Seamless flying from outdoor to GPS-denied environment (i.e., under canopy) via the PTAM-GPS sensor fusion of [60]: flying of the drone from open space to canopy to open space again (*first/third lines*); and front-view camera scene with tracking features (*second/fourth lines*)

neously tracking the robot pose in the map. One of the first SLAM results based on a monocular camera was presented in [75], which is particularly suitable for robots that should be light and affordable such as drones. The algorithm of [75] is designed so that both the localization (or tracking) and the mapping are running at every single video frame, thereby, resulting in rather heavy computational load and slower algorithm running, especially for the mapping process, where a large number of corresponding features are to be triangulated and continuously added to the map.

For typical robot operations, yet, the localization of the robot is more critical (e.g., for control stability), thus, should be performed much faster, than the mapping. Along this reasoning, [76] proposed parallel tracking and mapping (PTAM), which separates the tracking (i.e., localization) and mapping threads and execute them in parallel, yet, with different speeds. That is, the mapping process is performed occasionally only when new key frames are added with enough motion of camera, whereas the tracking process runs with a faster rate separately from the mapping process using the map points in the current visual frame. Through this tracking-mapping separation, the PTAM algorithm can provide fast localization while requiring modest computing power, suitable for the drones equipped only with lightweight onboard PCs (see Sect. 2.5). The output of this PTAM can then be fused with other sensors (e.g., EKF sensor fusion in Sect. 5.2). Some successful applications of PTAM for drone flying are [8, 60, 77]. See also Figs. 7 and 8. Of course, more advanced SLAM techniques can also be applied for the drone flying (e.g., ORB-SLAM [78], graph-SLAM [79], LSD-SLAM [80], dense-SLAM [81]), which is a topic of active research currently being pursued by many research labs and groups.

The optical flow and SLAM techniques as explained above can be performed by camera alone, so it can be easily integrated with IMU via, e.g., the EKF of Sect. 5.2, resulting in a loosely-coupled VINS. This loosely-coupled VINS, however, cannot correct vision-only estimation error (e.g., drift) and also requires additional estimation of extrinsic scale. For this, the tightly-coupled VINS has been investigated (e.g., [82–85]), which, despite its complexity in computation and implementation, can provide better state estimation than loosely-coupled VINS [85]. For instance, the work of [84] defines the state to be the combination of the IMU poses of sliding window, depths of observed features and IMU-camera inter-sensor offset and estimates this (combined) state by solving a nonlinear optimization given the IMU and visual-scene measurements. They also mounted this tightly-coupled VINS on a quadrotor and reported the final position drift to be around 2.80 m for a large-distance (i.e., around 640 m traversing distance) outdoor flying experiment.

6 Conclusion

In this paper, we present a tutorial introduction to the field of multi-rotor drones, encompassing: configuration, components and constructions; flight principle and system modeling; flying control methodologies incorporating SE(3) motion of the drones; and state estimation techniques, particularly EKF sensor fusion and vision-based estimation. The field of drones is expected to continue its growth, as it fundamentally extends the ground-bound 2D capability of conventional robots to 3D space with rather ease of operation and affordability. We are already witnessing some established or soon-to-be-materialized drone applications as stated in Sect. 1, which are mostly based on the capability of autonomous flying of the drones—the main concern of the current tutorial paper. Of course, although not covered in this tutorial paper, there still are many issues to resolve and/or new emerging applications for the drones to truly change our everyday life, some of which include: (1) drastic battery improvement and new power sources; (2) physical safety and cyber security; (3) high-speed perception and collision-free flying; (4) aerial operation and manipulation; (5) new drone designs to overcome limitation of conventional multi-rotor drones, to name only few.

References

1. Skynamic shooting for game of thrones commercial with RED EPIC Dragon Drone. <https://www.skynamic.net/en/news/page/2>. Accessed 20 Feb 2017
2. D'Andrea R (2014) Guest editorial can drones deliver? IEEE Trans Autom Sci Eng 11(3):647–648
3. Trimble Inc Trimble ZX5. <http://www.trimble.com/Survey/ZX5.aspx?tab=Overview>. Accessed 29 Dec 2016
4. PrecisionHawk. <http://www.precisionhawk.com/>. Accessed 29 Dec 2016
5. Özslan T, Mohta K, Keller J, Mulgaonkar Y, Taylor CJ, Kumar V, Wozencraft JM, Hood T (2016) Towards fully autonomous visual inspection of dark featureless dam penstocks using MAVs. In: Proceedings of IEEE international conference on robotics and automation
6. Li Z, Liu Y, Walker R, Hayward R, Zhang J (2010) Towards automatic power line detection for a UAV surveillance system using pulse coupled neural filter and an improved hough transform. Mach Vis Appl 21(5):677–686
7. Scentroid DR300, Flying Laboratory. <http://scentroid.com/scentroid-sampling-drone/>. Accessed 29 Dec 2016
8. Achtelik M et al (2012) Sfly: swarm of micro flying robots. In: Proceedings of IEEE/RSJ International conference on intelligent robots and systems, pp 2649–2650
9. Airdog, Inc Airdog. <https://www.airdog.com/>. Accessed 29 Dec 2016
10. Intel, Corp Intel's 500 Drone Light Show. <http://www.intel.com/content/www/us/en/technology-innovation/aerial-technology-overview.html>. Accessed 29 Dec 2016
11. Parrot SA AR Drone 2.0. <https://www.parrot.com/us/#drones>. Accessed 29 Dec 2016

12. 3D Robotics, Inc 3D Robotics Solo. <https://3dr.com/solo-drone/>. Accessed 29 Dec 2016
13. DJI DJI drones. <http://www.dji.com/products/drones>. Accessed 29 Dec 2016
14. Syma Co, Ltd Syma Cheerwing X5SW. <http://www.symatoys.com/goods/fpv-quadcopter.html>. Accessed 29 Dec 2016
15. Yuneec Yuneec Typhoon. https://www.yuneec.com/en_US/products/typhoon/h/overview.html. Accessed 29 Dec 2016
16. Cheerson Co, Ltd, Cheerson CX-10C. <http://www.cheersonhobby.com/en-US/Home/ProductDetail/125>. Accessed 17 Feb 2017
17. TRNDlabs SKEYE Nano. <https://www.trndlabs.com/product/skeye-nano-drone-with-camera>. Accessed 17 Feb 2017
18. GTF GTF AG-10. http://en.gtfdrone.com/goods/good?goods_id=45. Accessed 17 Feb 2017
19. Aerial Technology International ATI AgBOT. <http://www.aerialtechnology.com>. Accessed 17 Feb 2017
20. EHANG Inc EHANG 184. <http://www.ehang.com/ehang184>. Accessed 17 Feb 2017
21. e-volo GmbH Volocopter VC200. <http://volocopter.com>. Accessed 20 Feb 2017
22. Amazoncom, Inc Amazon Prime Air. <https://www.amazon.com/Amazon-Prime-Air/b?ie=UTF8&node=8037720011>. Accessed 29 Dec 2016
23. Airbus SAS Airbus demonstrates aircraft inspection by drone at farnborough. <http://www.airbusgroup.com/int/en/news-media/media-item=33fc85f6-e973-43c2-8259-e196158d7f80~.html>. Accessed 29 Dec 2016
24. Uragun B (2011) Energy efficiency for unmanned aerial vehicles. In: Proceedings of IEEE international conference on machine learning and applications and workshops, pp 316–320
25. Stock D (2015) Fast and furious. *New Sci* 228(3052):60–61
26. Briod A, Kornatowski P (2014) A collision-resilient flying robot. *J Field Robot* 31(4):496–509
27. Scaramuzza D, Achtelik MC, Doitsidis L, Friedrich F, Kosmatopoulos E, Martinelli A, Achtelik MW, Chli M, Chatzichristofis S, Kneip L, Gurdan D, Heng L, Lee GH, Lynen S, Pollefeys M, Renzaglia A, Siegwart R, Stumpf JC, Tanskanen P, Troiani C, Weiss S, Meier S (2014) Vision-controlled micro flying robots: from system design to autonomous navigation and mapping in GPS-denied environments. *IEEE Robot Autom Mag* 21(3):26–40
28. Tsui JB (2000) Fundamentals of global positioning system receivers: a software approach. Wiley, New York
29. Kaplan ED, Hegarty CJ (2005) Understanding GPS: principles and applications, 2nd edn. Artech House, Norwood, MA
30. Bristeau PJ, Callou F, Vissiere D, Petit N (2011) The navigation and control technology inside the AR.Drone micro UAV. In: Proceedings of IFAC world congress, pp 1477–1484
31. Weiss S, Achtelik MW, Lynen S, Chli M, Siegwart R (2012) Real-time onboard visual-inertial state estimation and self-calibration of MAVs in unknown environments. In: Proceedings of IEEE/RSJ international conference on intelligent robots and systems, pp 957–964
32. Hesch JA, Kottas DG, Bowman SL, Roumeliotis SI (2014) Camera-IMU-based localization: observability analysis and consistency improvement. *Int J Robot Res* 33(1):182–201
33. Shen S, Michael N, Kumar V (2015) Tightly-coupled monocular visual-inertial fusion for autonomous flight of rotorcraft MAVs. In: Proceedings IEEE international conference on robotics and automation, pp 5303–5310
34. 3D Robotics, Inc Pixhawk. <https://pixhawk.org/modules/pixhawk>. Accessed 29 Dec 2016
35. DJI Dji flight controller. <http://www.dji.com/products/industrial#developer-nav>. Accessed 29 Dec 2016
36. Hardkernel co, Ltd Odroid xu4. http://www.hardkernel.com/main/products/prdt_info.php?g_code=G143452239825. Accessed 29 Dec 2016
37. Intel, Corp Intel nuc. <http://www.intel.eu/content/www/eu/en/nuc/overview.html>. Accessed 29 Dec 2016
38. Nvidia, Corp Nvidia jetson tx1. <http://www.nvidia.com/object/jetson-tx1-dev-kit.html>. Accessed 29 Dec 2016
39. Vásárhelyi G, Virágó C, Somorjai G, Tarcai N, Szörényi T, Nepusz T, Vicsek T (2014) Outdoor flocking and formation flight with autonomous aerial robots. In: Proceedings IEEE/RSJ international conference on intelligent robots and systems, pp 3866–3873
40. Athukoralage D, Guvenc I, Saad W, Bennis M (2016) Regret based learning for UAV assisted LTE-u/wifi public safety networks. In: Proceedings IEEE conference on global communications, pp 1–7
41. Qazi S, Siddiqui AS, Wagan AI (2015) UAV based real time video surveillance over 4g LTE. In: Proceedings of IEEE international conference on open source systems and technologies, pp 141–145
42. Mueller MW, D'Andrea R (2014) Stability and control of a quadrotor despite the complete loss of one, two, or three propellers. In: Proceedings IEEE international conference on robotics and automation, pp 45–52
43. Lee D (2012) Distributed backstepping control of multiple thrust-propelled vehicles on a balanced graph. *Automatica* 48(11):2971–2977
44. Ghadiok V, Goldin J, Ren W (2012) On the design and development of attitude stabilization, vision-based navigation, and aerial gripping for a low-cost quadrotor. *Auton Robots* 33:41–68
45. Lim H, Park J, Lee D, Kim HJ (2012) Build your own quadrotor: open-source projects on unmanned aerial vehicles. *IEEE Robot Autom Mag* 19(3):33–45
46. Mahony R, Kumar V, Corke P (2012) Multirotor aerial vehicles: modeling, estimation, and control of quadrotor. *IEEE Robot Autom Mag* 19(3):20–32
47. Lee DJ, Franchi A, Son HI, Ha C, Bulthoff HH, Giordano PR (2014) Semiautonomous haptic teleoperation control architecture of multiple unmanned aerial vehicles. *IEEE/ASME Trans Mechantron* 18(4):1334–1345
48. Mahony R, Hamel T (2004) Robust trajectory tracking for a scale model autonomous helicopter. *Int J Robust Nonlinear Control* 14:1035–1059
49. Hua MD, Hamel T, Morin P, Samson C (2009) A control approach for thrust-propelled underactuated vehicles and its application to vtol drones. *IEEE Trans Autom Control* 54(8):1837–1853
50. Ha C, Zuo Z, Choi FB, Lee DJ (2014) Passivity-based adaptive backstepping control of quadrotor-type UAVs. *Robot Auton Syst* 62(9):1305–1315
51. Lee T, Leok M, McClamroch NH (2010) Geometric tracking control of a quadrotor UAV on $se(3)$. In: Proceedings IEEE conference on decision and control, pp 5420–5425
52. Mellinger D, Kumar V (2011) Minimum snap trajectory generation and control for quadrotors. In: Proceedings IEEE international conference on robotics and automation, pp 2520–2525
53. Bouabdallah S, Siegwart R (2007) Full control of a quadrotor. In: Proceedings IEEE/RSJ international conference on intelligent robots and systems, pp 153–158
54. Lee D, Kim HJ, Sastry S (2009) Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter. *Int J Control Autom Syst* 7(3):419–428
55. Mahony R, Hamel T, Pflimlin JM (2008) Nonlinear complementary filters on the special orthogonal group. *IEEE Trans Autom Control* 53(5):1203–1217
56. Shen S, Mulgaonkar Y, Michael N, Kumar V (2013) Vision-based state estimation for autonomous rotorcraft MAVs in complex environments. In: Proc. IEEE international conference on robotics and automation, pp 1758–1764
57. Shen S, Mulgaonkar Y, Michael N, Kumar V (2014) Multi-sensor fusion for robust autonomous flight in indoor and outdoor environments with a rotorcraft MAV. In: Proceedings of IEEE international conference on robotics and automation, pp 4974–4981

58. Forster C, Pizzoli M, Scaramuzza D (2014) SVO: fast semi-direct monocular visual odometry. In: Proceedings of IEEE international conference on robotics and automation, pp 15–22
59. Bloesch M, Omari S, Hutter M, Siegwart R (2015) Robust visual inertial odometry using a direct EKF based approach. In: Proceedings of IEEE/RSJ international conference on intelligent robots and systems, pp 298–304
60. Lee Y, Yoon J, Yang H, Kim C, Lee D (2016) Camera-GPS-IMU sensor fusion for autonomous flying. In: Proceedings of IEEE international conference on ubiquitous and future networks, pp 85–88
61. Gebre-Egziabher D, Hayward RC, Powell JD (2004) Design of multi-sensor attitude determination systems. *IEEE Trans Aerosp Electron Syst* 40(2):627–649
62. Markley FL, Crassidis JL, Cheng Y (2005) Nonlinear attitude filtering methods. In: Proceedings of AIAA guidance, navigation, and control conference, pp 15–18
63. Baerveldt AJ, Klang R (1997) A low-cost and low-weight attitude estimation system for an autonomous helicopter. In: Proceedings of IEEE International Conference on Intelligent Engineering Systems, pp 391–395
64. Vik B, Fossen TI (2001) A nonlinear observer for GPS and INS integration. In: Proceedings of IEEE conference on decision and control, vol 3, pp 2956–2961
65. Lefferts EJ, Markley FL, Shuster MD (1982) Kalman filtering for spacecraft attitude estimation. *J Guid Control Dyn* 5(5):417–429
66. Gross J, Gu Y, Gururajan S, Seanor B, Napolitano MR (2010) A comparison of extended kalman filter, sigma-point Kalman filter, and particle filter in GPS/INS sensor fusion. In: Proceedings of AIAA guidance, navigation, and control conference, pp 1–19
67. Zhang P, Gu J, Milios EE, Huynh P (2005) Navigation with IMU/GPS/digital compass with unscented Kalman filter. In: Proceedings of IEEE international conference on mechatronics and automation, vol 3, pp 1497–1502
68. Yun B, Peng K, Chen BM (2007) Enhancement of GPS signals for automatic control of a UAV helicopter system. In: Proceedings of IEEE international conference on control and automation, pp 1185–1189
69. El-Rabbany A (2002) Introduction to GPS: the global positioning system. Artech House, Norwood, MA
70. Duffield R, Reid M, Baker J, Spratford W (2010) Accuracy and reliability of GPS devices for measurement of movement patterns in confined spaces for court-based sports. *J Sci Med Sport* 13(5):523–525
71. Grzonka S, Grisetti G, Burgard W (2012) A fully autonomous indoor quadrotor. *IEEE Trans Robot* 28(1):90–100
72. Santamaria-Navarro A, Sola J, Andrade-Cetto J (2015) High-frequency MAV state estimation using low-cost inertial and optical flow measurement units. In: Proceedings of IEEE/RJS international conference on intelligent robots and systems, pp 1864–1871
73. Kendoul F, Fantoni I, Nonami K (2009) Optic flow-based vision system for autonomous 3D localization and control of small aerial vehicles. *Robot Auton Syst* 57(6):591–602
74. Honegger D, Meier L, Tanskanen P, Pollefeyns M (2013) An open source and open hardware embedded metric optical flow CMOS camera for indoor and outdoor applications. In: Proceedings IEEE international conference on robotics and automation, pp 1736–1741
75. Davison AJ (2003) Real-time simultaneous localisation and mapping with a single camera. In: Proceedings of IEEE international conference on computer vision, pp 1403–1410
76. Klein G, Murray D (2007) Parallel tracking and mapping for small AR workspaces. In: Proceedings of IEEE/ACM international symposium on mixed and augmented reality. IEEE, pp 225–234
77. Monajjemi VM, Wawerla J, Vaughan R, Mori G (2013) Hri in the sky: creating and commanding teams of UAVs with a vision-mediated gestural interface. In: Proceedings of IEEE/RJS international conference on intelligent robots and systems, pp 617–623
78. Mur-Artal R, Montiel J, Tardós JD (2015) ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Trans Robot* 31(5):1147–1163
79. Grisetti G, Kummerle R, Stachniss C, Burgard W (2010) A tutorial on graph-based SLAM. *IEEE Intell Transp Syst Mag* 2(4):31–43
80. Engel J, Schöps T, Cremers D (2014) LSD-SLAM: Large-scale direct monocular SLAM. In: Proceedings of European conference on computer vision, pp 834–849
81. Nieto J, Guivant J, Nebot E (2006) Denseslam: simultaneous localization and dense mapping. *Int J Robot Res* 25(8):711–744
82. Li M, Mourikis AI (2013) High-precision, consistent EKF-based visual-inertial odometry. *Int J Robot Res* 32(6):690–711
83. Hesch JA, Kottas DG, Bowman SL, Roumeliotis SI (2014) Camera-IMU-based localization: observability analysis and consistency improvement. *Int J Robot Res* 33(1):182–201
84. Yang Z, Shen S (2017) Monocular visual-inertial state estimation with online initialization and camera-IMU extrinsic calibration. *IEEE Trans Autom Sci Eng* 14(1):39–51
85. Leutenegger S, Lynen S, Bosse M, Siegwart R, Furgale P (2015) Keyframe-based visual-inertial odometry using nonlinear optimization. *Int J Robot Res* 34(3):314–334