Image-Based Visual Servo Control of Aerial Robotic Systems Using Linear Image Features

Robert Mahony, *Member, IEEE,* and Tarek Hamel

***Abstract*—An image-based "eye-in-hand" visual servo-control design is proposed for underactuated rigid-body dynamics. The dynamic model considered is motivated by recent work on vertical takeoff and landing aerial robotic vehicles. The task considered is that of tracking parallel linear visual features. The proposed design exploits the geometry of the task considered and passivity-like properties of rigid-body dynamics to derive a control Lyapunov function using backstepping techniques.**

***Index Terms*—Aerial robotic vehicle, image-based visual servo (IBVS), rigid-body dynamics, underactuated systems.**

I. Introduction

VISUAL servo control concerns the problem of using a camera to provide sensor information to servo-position a robotic system. Classical visual servo control was developed for serial-link robotic manipulators, with the camera typically mounted on the end-effector [18]. More recently, applications involving mobile systems have been considered [7], [28], [34]. Visual servo control has the potential to become an important enabling technology for aerial robotic vehicles capable of quasi-stationary flight regimes, such as helicopters [2], [11], [38], [40], dirigibles [4], and flying robots developed explicitly for indoor operation [1], [16]. Most aerial robotic systems may be modeled as underactuated rigid-body dynamics, with possibly additional dynamics due to actuator effects such as the flapping dynamics of the main rotor of a helicopter [30]. Such systems are inherently difficult to control, and add significantly to the complexity of image-based visual servo (IBVS) control design.

Visual servo systems may be divided into two main classes [18], [26]. *Position-based visual servo* (PBVS) involves reconstruction of the target pose with respect to the camera, and leads to a classical dual estimation and control problem, in which the state (camera pose) of the system is estimated using visual information [13], [29], [43], and the control design is a classical state-space design [1], [2], [38], [40]. PBVS control leads to a straightforward control-design problem. However, the pose estimation is highly sensitive to camera calibration errors and errors in the three-dimensional (3-D) world model used to derive pose estimates [18]. The sensitivity of PBVS design to camera calibration is particularly worrying when low-cost, and consequently low-quality, sensor systems are employed. In contrast, for IBVS, the control task is posed directly in terms of image features. A controller is designed to maneuver the image features toward a goal configuration that implicitly solves the original Cartesian motion-planning problem [37]. The approach is inherently robust to camera calibration and target modeling errors [10], [18]. However, the control-design problem for IBVS is more complex than for PBVS control, and the approach used in classical IBVS control [9] is not suitable to be extended to the integrated control of nonlinear fully dynamic systems. Recent approaches to robustly extending IBVS to dynamic systems include dissipative control design in the image space [27], robust backstepping [16], [47], and optimal control techniques [48]. The specific challenges posed by the dynamics of aerial robotic vehicles have proved difficult to overcome, and very few rigorous developments of IBVS control exist. In prior work by the authors [16], we propose an algorithm that uses simple point landmarks as visual targets for control of underactuated rigid-body dynamics. An alternate approach that has proved fruitful uses insight from the behavior of flying insects and animals (particularly techniques related to visual flow) to develop control strategies for aerial robotic vehicles [5], [45]. To our knowledge, there is no prior work that uses more structured geometric targets, such as linear features for control of aerial robotic systems.

In this paper, we propose an IBVS control design to track parallel linear features for underactuated rigid-body dynamics. The material presented is an extension of earlier work [25]. The dynamic system model considered is based on models proposed recently in the literature [14], [20], [40], [42] for aerial robotic vehicles capable of quasi-stationary flight. Linear features are represented using the binormalized Euclidean Plücker coordinates [3]. The representation achieves two key objectives. First, the tasks of line tracking and velocity regulation are decoupled, and second, passivity-like properties of rigid-body dynamics are preserved separately in each control task. The proposed design exploits the passivity-like properties of rigid-body dynamics to derive a control Lyapunov function, using backstepping techniques. The derivation is based on that undertaken in [16] with several modifications and improvements. In addition to providing a rigorous control design for the full dynamics of the system considered, the approach taken provides a natural manner to deal with uncertainty associated with the unknown depths of observed image features. Indeed, the approach taken avoids estimation and inversion of the kinematic image Jacobian of the system, a major source of nonrobustness and global

Manuscript received May 7, 2003; revised February 11, 2004. This paper was recommended for publication by Associate Editor Y. H. Liu and Editor S. Hutchinson upon evaluation of the reviewers' comments. This paper was presented in part at the IEEE Conference on Intelligent Robots and Systems, Maui, HI, October 2001.

R. Mahony is with the Department of Engineering, Australian National University, Canberra, ACT 0200, Australia (e-mail: mahony@ieee.org).

T. Hamel is with I3S, UNSA-CNRS, BP 121, 06903 Sophia Antipolis, France (e-mail: thamel@i3s.unice.fr).

Digital Object Identifier 10.1109/TRO.2004.835446

stability limitations for the linear approximation and feedback linearization control designs standard in the literature. Local exponential stability of the closed-loop system around a goal trajectory is proved. The underactuated nature of the dynamics, and specification of primary control task, leaves a degree of freedom (DOF) in the attitude dynamics of the model that must be dealt with via a secondary control task [37], without disturbing the primary closed-loop response. The approach taken here exploits the geometric properties of the linear image features to avoid requiring additional visual information, and provides a slicker stability analysis than that employed in prior work [16]. A full stability analysis is provided, showing local asymptotic stability of the secondary task variables. This paper provides a rigorous nonlinear analysis of a practical visual servo task for underactuated rigid-body dynamics, and forms a part of an ongoing worldwide effort to control aerial robots in real-world conditions using low-cost light sensor systems [8].

The paper is arranged into five sections. Section II presents the dynamic system model considered, introduces the binormalized Plücker coordinates to represent linear visual features and defines the image-based error used. Section III derives a Lyapunov control function for the trajectory tracking task considered, and analyzes the stability of the closed-loop system. Section IV applies the control strategy to a simplified model for the dynamics of a four-rotor vertical take-off and landing (VTOL) vehicle known as an X4 flyer [1], [15], and presents some simulation results. The final section provides a short summary of conclusions and an indication of future research directions.

II. Problem Formulation

The dynamics of the airframe of an aerial robotic vehicle are modeled as rigid-body dynamics subject to exogenous forces and torques generated by its thrusters and aerodynamic lift surfaces. Let $\mathcal{I} = \{E_x, E_y, E_z\}$ denote the world frame, and let $\mathcal{A} = \{E_1^a, E_2^a, E_3^a\}$ denote the body-fixed frame of the rigid body. The position of the airframe in the world frame is denoted $\xi = (x, y, z) \in \mathcal{I}$, and its attitude (or orientation) is given by a rotation $R : \mathcal{A} \to \mathcal{I}$, where $R \in SO(3)$ is an orthogonal rotation matrix. Let V (resp. Ω) denote the linear (resp. angular) velocity of the body expressed in the body-fixed frame. Let m denote the total mass, and $\mathbf{I}$ denote the inertia matrix expressed in the body in the body-fixed frame. The dynamics of a rigid body are[1] [44]

$$\dot{\xi} = RV \tag{1}$$

$$m\dot{V} = -m\Omega_\times V + F \tag{2}$$

$$\dot{R} = R\Omega_\times \tag{3}$$

$$\mathbf{I}\dot{\Omega} = -\Omega_\times \mathbf{I}\Omega + \Gamma. \tag{4}$$

The exogenous force and torque are denoted F and Γ, respectively (cf. Fig. 1).

The exogenous force and torque inputs considered correspond to a typical arrangement found on a VTOL aircraft (cf. Section IV). The exogenous inputs are written as a single translational force, denoted F in Fig. 1, along with full torque control, shown by the torques Γ_1, Γ_2, and Γ_3 around axes E_1^a, E_2^a, and E_3^a, respectively. The force F combines thrust, lift, gravity, and drag components. It is convenient to separate the gravity component $mgE_z = mgR^T E_3^a$ from the thrust force

$$F := -T\mathbf{F} + mgR^T E_3^a \tag{5}$$

where $\mathbf{F} \in \mathcal{A}$ is a unit vector in the body-fixed frame representing the orientation of force input, and $T \in \mathbb{R}$ is a scalar input representing the magnitude of the thrust. Control of the airframe is obtained by using the torque control $\Gamma = (\Gamma_1, \Gamma_2, \Gamma_3)$ to align the force $F_0 := T\mathbf{F}$ as required to track the goal trajectory.

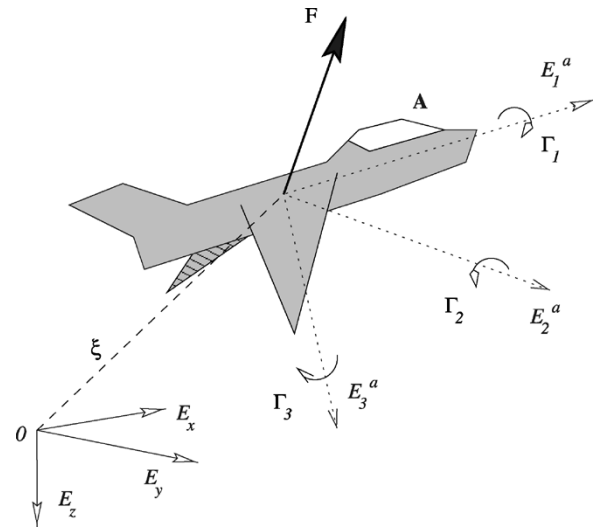

Fig. 1. References frames, forces, and torques for an unmanned aerial vehicle (UAV).

The visual features considered are a collection of parallel lines. A line in Euclidean space can be represented in either an implicit or explicit mathematical form. A linear feature is represented implicitly as the set of points verifying a set of linear equations [12]. For applications in visual servo control, the observed feature is a projected image (onto the image plane) of the line in Euclidean space. The relationship between the representation of a line in the image plane and its implicit representation in Euclidean space leads to technical complications in the development of IBVS control laws. Several successful implementations of kinematic IBVS control laws based on this approach have been reported [6], [36]. However, in the authors' opinion, the approach does not lend itself to extension to systems with dynamics. An explicit representation of a linear feature involves specifying its direction and offset with respect to a given point. The Plücker coordinates [6], [12], [35] are an example of an explicit representation of a line. In *normalized Euclidean Plücker coordinates* [36], a line is represented by its direction $U \in \mathcal{A}$, a unit vector,[2] and a vector H

$$H = P \times U$$

where P is the point on the line closest to the center of the frame $\mathcal{A}$ (cf. Fig. 2). In binormalized Plücker coordinates (U, h), a line is represented by its normalized direction U, along with

$$h = \frac{H}{|H|} \in \mathcal{A}.$$

[1]The notation $\Omega_\times$ denotes the skew-symmetric matrix such that $\Omega_\times v = \Omega \times v$ for the vector cross-product $\times$ and any vector $v \in \mathbb{R}^3$.

[2]In Euclidean Plücker coordinates, the vector U need not be normalized.

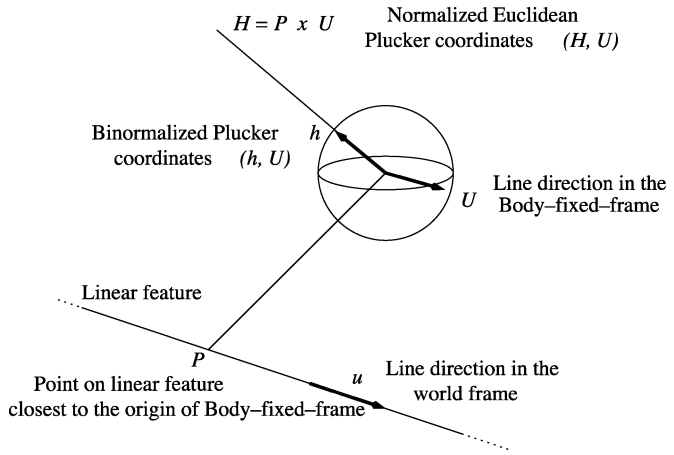

Fig. 2. Geometric of the construction of binormalized Plücker coordinates.

Note that h is the unit normal to the plane containing the line and the origin of $\mathcal{A}$. Thus, although the point P and the norm of $|H|$ are difficult to determine from the visual data, the normal direction h is known, once the linear feature has been identified in the image. The information that was coded in the norm $|H|$ is not represented by the binormalized Euclidean Plücker coordinates. As a consequence, there are an infinity of lines that are represented by the same binormalized Plücker coordinates (U, h) [3, Prop. 1]. The strong geometric interpretation of the unit vector h confers a structural passivity to image-based error criterion similar to that exploited in the authors' previous work [16].

Remark 2.1: Extracting linear features from an image sequence is a well-studied problem and is a fundamental routine in image-segmentation algorithms (cf. e.g., [41, Sec. 10.3–10.4]). The use of Hough transform techniques applied to an image sequence preprocessed by an edge-detector algorithm (e.g., the Canny edge detector) leads to a robust method to identify image lines.

The kinematics of U, P, and H are inherited from the motion of the camera relative to the observed targets. The direction of the line in the world frame is denoted $u := RU \in \mathcal{I}$ and is assumed to be constant. Straightforward calculations show

$$\dot{U} = -\Omega \times U \tag{6}$$
$$\dot{P} = -\Omega \times P + \pi_U V \tag{7}$$
$$\dot{H} = -\Omega \times H + V \times U \tag{8}$$

where $\pi_x = (I_3 - xx^T)$ is the projection $\pi_x : \mathbb{R}^3 \to T_x S^2$ onto the tangent space of the sphere S^2 for any point $x \in S^2$. The kinematics of the normalized direction h are given by

$$\begin{aligned}\dot{h} &= \frac{1}{|H|}\dot{H} - \frac{\frac{d}{dt}|H|}{|H|^2}H \\ &= -\Omega_\times h + \frac{1}{|H|}\pi_h(V \times U).\end{aligned} \tag{9}$$

The observed target is represented by a set of n normalized features h_i which are visible to the camera (cf. Fig. 3). The

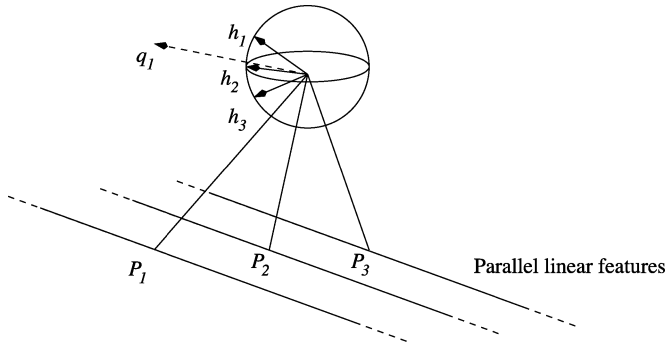

Fig. 3. Structure of the visual feature q_1 for the case of three parallel lines.

information from each feature is summed together to provide a single 3-D image feature

$$q_1 := \sum_{i=1}^{n} h_i. \tag{10}$$

Note that each h_i is orthogonal to U, and consequently, q_1 is also orthogonal to U. Since each h_i is a normalized direction, the quantity q_1 is dimensionless.

The visual error used is the difference between the weighted centroid q_1 and a dimensionless goal vector q^*

$$\delta_1 := q^* - q_1. \tag{11}$$

The goal vector used is generated from *a priori* assumption of the target configuration $\{H_i^*\}$ as observed from the desired set point

$$q^* = \sum_{i=1}^{n} h_i^*, \quad h_i^* = \frac{H_i^*}{|H_i^*|}.$$

Note that since both q_1 and q^* are dimensionless, δ_1 is a dimensionless measure of error. The goal vector is chosen fixed in the world frame (independent of the attitude of the camera), but is written in the body-fixed frame. Thus, q^* inherits ego-motion from the attitude dynamics of the rigid-body frame

$$\dot{q}^* = -\Omega \times q^*.$$

Since the goal is constructed from *a priori* information on the target, the following assumption is natural.

Assumption 2.2: There exists a point $\xi \in \mathbb{R}^3$ in task space such that the visual feature q_1 is equal to the goal, $q_1 = q^*$. Moreover, $U^T q^* = 0$. Furthermore, if all the target lines lie in an affine plane, then it is assumed that ξ does not lie in this plane.

A consequence of the second part of the assumption is that at the point ξ, at least two of the h_i image features associated with the observed target are linearly independent. If there are three or more linear features that are not coplanar, then this part of the assumption is not required.

Lemma 2.3: Consider an observed target comprising $n \geq 2$ distinct lines parallel to world direction $u = RU \in \mathcal{I}$. Let *Assumption 2.2* hold and denote the given point at which $\delta_1 = q^* - q_1 = 0$ by $\xi \in \mathcal{I}$. Then, $\delta_1 \equiv 0$ on the set of points $y = \xi + tu \in \mathcal{I}$ for $t \in \mathbb{R}$. Moreover, for any point y such that $\delta_1 = 0$, then $y = \xi + tu$ for some $t \in \mathbb{R}$.

Proof: Deriving the visual error δ_1 yields

$$\dot{\delta}_1 = -\Omega_\times \delta_1 - Q(V \times U) \tag{12}$$

where

$$Q := \sum_{i=1}^{n} \frac{1}{|H_i|} \pi_{h_i}. \tag{13}$$

Each individual projection operator π_{h_i} is positive semidefinite with null space h_i. The lemma conditions guarantee at least two distinct features $h_i \neq \alpha h_j$, for $\alpha \in \mathbb{R}$ and $i \neq j$, and it follows that Q is full rank. Consider the quadratic cost $|\delta_1|^2$. From (12), one has

$$\frac{d}{dt}|\delta_1|^2 = -\delta_1^T Q(V \times U). \tag{14}$$

Choosing $V = U$, it is clear that $(d/dt)|\delta_1|^2 \equiv 0$ and the cost is degenerate in the direction u. Thus, for any $\xi \in \mathcal{I}$ such that $\delta_1 = 0$, $\delta_1 \equiv 0$ on the set $y = \xi + tu \in \mathcal{I}$ for $t \in \mathbb{R}$. Given any point y with $\delta_1 = 0$, choose V in the direction orthogonal to U and fix $\Omega = 0$. Recalling (12), it is clear that $\delta_1 \not\equiv 0$ along the trajectory generated. This shows that the cost $|\delta_1|^2$ is nondegenerate in directions orthogonal to u. It remains to show that there is a unique line $y = \xi + tu$ for which $\delta_1 \equiv 0$. The proof proceeds by contradiction. *Assume that there exist two distinct lines of points (parallel to u) on which* $\delta_1 = 0$. From these lines, choose two points lying on a plane perpendicular to U, and consider a line connecting the two points. The above arguments show that $|\delta_1|^2 \not\equiv 0$ along this line. By continuity, there exists a local maximum of the cost $|\delta_1|^2$ on this line, for which $\delta_1 \neq 0$. Since δ_1 is perpendicular to U, it is possible to choose V such that $V \times U = \delta_1$ at this point. Substituting into (14) provides a contradiction of the local maxima property and completes the proof. ∎

Equation (12) defines the kinematics of the visual error δ_1. The image Jacobian[3] [18] (denoted J) is obtained by rewriting (12) in the classical form

$$\dot{\delta}_1 = [Q\mathrm{sk}(U) \quad \mathrm{sk}(\delta_1)] \begin{pmatrix} V \\ \Omega \end{pmatrix} \tag{15}$$

$$= J \begin{pmatrix} V \\ \Omega \end{pmatrix}. \tag{16}$$

In common with classical IBVS algorithms, the image Jacobian depends on the unknown depth of the image features. That is, although the directions $\{h_i\}$ are known, the depth parameters $\{|H_i|\}$ cannot be measured directly from visual data. However, for the proposed image error, the unknown parameters enter only into the definition of the matrix Q (13), and here they enter in a structured manner such that the matrix $Q > 0$ is always positive definite. It is this property that is exploited in Section III in the control design, avoiding the necessity of estimating or approximating the image Jacobian, a fundamental difficulty in most IBVS algorithms [9], [17], [26], [32], [33]. Nevertheless, some bounds are required on the trajectories considered, to avoid ill conditioning of the control design. In this development, we define a region of space by a pair of uniform bounds on the matrix Q

$$\lambda_{\min} < \{\lambda_i(Q)\} < \lambda_{\max}. \tag{17}$$

[3]Also known as the interaction matrix [9].

The proposed control design presented in Section III-A is valid for any trajectory contained in this region. The bounds $\lambda_{\max} > \lambda_{\min} > 0$ are used as robustness margins in the control design. Further analysis of the nature of the region defined by (17) is given in Section IV.

III. Control Design and Stability Analysis

In this section, a visual servo control is derived for the system described in Section II. The task considered is to stabilize the position of the vehicle with respect to the observed lines using visual feedback, and regulate the velocity of the vehicle in the direction u using additional sensor feedback.

There are two parts to the primary control error associated with position and velocity control, respectively. The first primary control error is the visual feature error δ_1 (11). Let v_d denote a desired constant velocity in the direction U. The second primary control error is defined to be[4]

$$\eta_2 := m\left((U^T V) - v_d\right). \tag{18}$$

The primary goal of the control algorithm is to (exponentially) stabilize the two error signals $(\delta_1, \eta_2) \in R^4$ (cf. Section III-A). The primary error criterion is *four*-dimensional, but only contains *three* DOFs due to the degenerate nature of δ_1. Due to the underactuated nature of the rigid-body dynamics considered, the primary error is sufficient to fix at least 5 DOFs of the rigid-body dynamics (cf. *Lemma 3.3*). To control the final DOF, a secondary control task [37] is introduced (cf. Section III-B).

From the development in Section II, the system dynamics may be written

$$\dot{\delta}_1 = -\Omega_\times \delta_1 - Q(V \times U) \tag{19}$$

$$m\frac{d}{dt}(V \times U) = -\Omega \times m(V \times U) + (F \times U) \tag{20}$$

$$m\frac{d}{dt}(V^T U) = F^T U \tag{21}$$

$$\dot{R} = R\Omega_\times \tag{22}$$

$$\mathbf{I}\dot{\Omega} = -\Omega \times \mathbf{I}\Omega + \Gamma. \tag{23}$$

The state of the system as written is (δ_1, V, R, Ω). Due to its construction, the error only corresponds to 2 DOFs of the underlying system. Thus, (19) and (20) correspond to a 2-DOF (four dynamic states) system. The third DOF in the translation movement is modeled by (21). The kinematics of this DOF are ignored, since the control task is to regulate the forward velocity.

Remark 3.1: The camera is used to measure the relative position of the rigid body with respect to the set of parallel lines (via the visual error δ_1). The remaining dynamic states (V, R, Ω), are obtained as the output of a filter based on a wide range of observed data. Data used includes "visual flow" calculations, GPS measurements, magnetometers, accelerometers, inertial gyroscopic sensors, and gyroscopic rate sensors. A properly tuned extended Kalman filter provides good estimates of the required state variables.

[4]The mass m is introduced in the definition of η_2 to precompensate for the effective difference in gain between the linear kinematics (1) and the linear dynamics (2). The same idea is embedded in the definition of δ_2 (24).

A. Primary Control Design

Let $k_1 > 0$ (in kg.m.s^{-1}) be a positive constant, and define δ_2 to be

$$\delta_2 := \frac{m}{k_1}(V \times U) - \delta_1, \quad k_1 > 0. \tag{24}$$

The units of the gain k_1 are chosen to ensure that (24) is dimensionally correct. Since δ_1 is dimensionless, this ensures that δ_2 is also dimensionless. Indeed, the units chosen for all the control gains defined in the following guarantee that the virtual backstepping errors introduced are dimensionless.

The term δ_2 may be viewed physically as the tracking error between the actual velocity (orthogonal to the lines) and the desired stabilizing kinematic feedback $(k_1/m)\delta_1$. Indeed, if $\delta_2 \equiv 0$, then the design degenerates into a classic (nonlinearizing) kinematic IBVS control design in the chosen visual features. Since the system is dynamic and underactuated, it is impossible to keep the error δ_2 small, except for certain trajectories where the dynamic errors are asymptotically stable. The backstepping methodology allows us to propagate the dynamic error δ_2 into the system dynamics, and construct a control Lyapunov function that depends on the dynamic tracking error δ_2 and further errors δ_3 and δ_4, etc., that are needed to compensate for additional dynamic tracking errors in the system, due to its underactuated nature. For certain trajectories, where the dynamic system effects are asymptotically stable, the Lyapunov function constructed may be used to construct a stabilizing control law. This result is formalized in *Lemma 3.3*. Before presenting the result, we provide a brief overview of the backstepping methodology applied to the system. Further details can be obtained from the authors for interested readers.

Define a storage function $S_{1,2}$

$$S_{1,2} = \frac{1}{2}|\delta_1|^2 + \frac{1}{2}|\delta_2|^2 + \frac{1}{2}|\eta_2|^2. \tag{25}$$

Taking the time derivative of $S_{1,2}$ and substituting for (19) yields

$$\dot{S}_{1,2} = -\frac{k_1}{m}\delta_1^T Q \delta_1 + \frac{k_1}{m}\delta_2^T Q \delta_2 + \frac{1}{k_1}\delta_2^T (F \times U) + \eta_2 (F^T U). \tag{26}$$

Note that the derivative of $S_{1,2}$ is independent of the angular velocity Ω.

The desired set point chosen for F, ensuring the stabilization of the primary task (δ_1, η_2), is

$$\begin{aligned}(F \times U) &:= -\frac{k_1^2 k_2}{m}\delta_2, \text{ where } k_2 \text{ (in m}^{-1}) > \lambda_{\max}(Q) \\ (F^T U) &:= -l_2 \eta_2 + m\dot{v}_d, \text{ where } l_2 \text{ (in s}^{-1}) > 0.\end{aligned} \tag{27}$$

Due to the fact that the rigid-body system considered is underactuated, the force input F does not provide sufficient control to directly assign the desired dynamics specified by (27). It is necessary to use (27) as "virtual" or "desired" force inputs for the next stage of a backstepping procedure. Recalling that the virtual control input to (26) enters via the terms $F \times U$ and $F^T U$, two new error terms (η_3, δ_3), incorporating information on the attitude of the camera via (5), are defined as the difference between the actual terms ($F \times U$ and $F^T U$) and their virtual expressions (27)

$$\delta_3 := \frac{m}{k_1^2 k_2}(F \times U) + \delta_2 \tag{28}$$

$$\eta_3 := \frac{1}{l_2}(F^T U) + \eta_2. \tag{29}$$

The derivative of the storage $S_{1,2}$ function is then

$$\begin{aligned}\dot{S}_{1,2} = &-\frac{k_1}{m}\delta_1^T Q \delta_1 - \frac{k_1}{m}\delta_2^T (k_2 I - Q)\delta_2 \\ &+ \frac{k_1}{m} k_2 \delta_2^T \delta_3 - l_2 \eta_2^2 + l_2 \eta_2 \eta_3.\end{aligned}$$

Deriving δ_3 and η_3 and recalling (19), (24), and (18) yields

$$\begin{aligned}\dot{\delta}_3 = &-\Omega_\times \delta_3 + \frac{k_1}{m} Q \delta_1 - \frac{k_1}{m}(k_2 I - Q)\delta_2 + \frac{k_1}{m} k_2 \delta_3 \\ &+ \frac{m}{k_1^2 k_2}\left(\frac{d}{dt}(F \times U) + \Omega_\times (F \times U)\right) \\ \dot{\eta}_3 = &-l_2 \eta_2 + l_2 \eta_3 + \frac{1}{l_2}\frac{d}{dt}(F^T U).\end{aligned}$$

Following standard backstepping procedures, let $(\dot{F}^v, \Omega_\times^v)$ denote the virtual control inputs for the next iteration of the backstepping. The following augmented vectorial term incorporating the angular velocity is assigned:

$$\begin{aligned}\left(\frac{d}{dt}(F \times U) + \Omega_\times (F \times U)\right)^v &= -\frac{k_1^2 k_2 (k_1 k_2 + k_3)}{m^2}\delta_3 \\ \left(\frac{d}{dt}(F^T U)\right)^v &= -l_2(l_2 + l_3)\eta_3\end{aligned}$$

where k_3 and l_3 are two positive constants (in kg.s^{-1} and s^{-1}, respectively). The expressions for the derivatives of δ_3 and η_3 may now be written

$$\begin{aligned}\dot{\delta}_3 = &-\Omega_\times \delta_3 + \frac{k_1}{m} Q \delta_1 - \frac{k_1}{m}(k_2 I - Q)\delta_2 - \frac{k_3}{m}\delta_3 \\ &+ \frac{(k_1 k_2 + k_3)}{m}\delta_4 \\ \dot{\eta}_3 = &-l_2 \eta_2 - l_3 \eta_3 + l_2(l_2 + l_3)\eta_4\end{aligned} \tag{30}$$

where δ_4 and η_4 are the final error terms used in the backstepping procedure

$$\begin{aligned}\delta_4 &= \frac{m^2}{k_1^2 k_2 (k_1 k_2 + k_3)}\left(\frac{d}{dt}(F \times U) + \Omega_\times (F \times U)\right) + \delta_3 \\ \eta_4 &= \frac{1}{l_2^2 (l_2 + l_3)}\frac{d}{dt}(F^T U) + \eta_3.\end{aligned} \tag{31}$$

Let S_3 be a storage function associated with this stage of the procedure

$$S_3 = \frac{1}{2}|\delta_3|^2 + \frac{1}{2}|\eta_3|^2. \tag{32}$$

Taking the derivative of S_3 and recalling (30), one obtains

$$\begin{aligned}\dot{S}_3 = &\frac{k_1}{m}\delta_3^T Q \delta_1 - \frac{k_1}{m}\delta_3^T (k_2 I - Q)\delta_2 - \frac{k_3}{m}\delta_3^T \delta_3 \\ &+ \frac{(k_1 k_2 + k_3)}{m}\delta_3^T \delta_4 - l_2 \eta_3 \eta_2 - l_3 \eta_3^2 + l_2(l_2 + l_3)\eta_4.\end{aligned}$$

The derivatives of the error terms δ_4 and η_4 are

$$\dot{\delta}_4 = \frac{m^2}{k_1^2 k_2 (k_1 k_2 + k_3)} w_1 - \Omega_\times \delta_3 + \frac{k_1}{m} Q \delta_1 - \frac{k_1}{m}(k_2 I - Q)\delta_2 - \frac{k_3}{m}\delta_3 + \frac{(k_1 k_2 + k_3)}{m}\delta_4 \quad (33)$$

$$\dot{\eta}_4 = \frac{1}{l_2^2 (l_2 + l_3)} w_2 - l_2 \eta_2 - l_3 \eta_3 + l_2 (l_2 + l_3)\eta_4 \quad (34)$$

where

$$w_1 := \frac{d^2}{dt^2}(F \times U) + \dot{\Omega}_\times (F \times U) + \Omega_\times \frac{d}{dt}(F \times U) \quad (35)$$

$$w_2 := \frac{d^2}{dt^2}(U^T F). \quad (36)$$

The intermediary control signals $w_1 \in \mathbb{R}^3$ and $w_2 \in \mathbb{R}$ are introduced to simplify the following analysis. Clearly, w_1 and w_2 can be used to control (33) and (34). It remains to show that the physical-control inputs of the system can be used to arbitrarily assign w_1 and w_2. Consider a linearizing control transformation of (4)

$$\tau := -\mathbf{I}^{-1}\Omega \times \mathbf{I}\Omega + \mathbf{I}^{-1}\Gamma \quad (37)$$

leading to the relation

$$\dot{\Omega} = \tau. \quad (38)$$

Recalling (5), the cross-product $F \times U$ and scalar product $U^T F$ can be expressed as

$$F \times U = -T\mathbf{F} \times U + mgR^T (E_3^a \times u)$$
$$U^T F = -TU^T \mathbf{F} + mgu^T E_3^a.$$

For any two vectors $a, b \in \mathbb{R}^3$, $\mathrm{sk}(a)\mathrm{sk}(b) - \mathrm{sk}(b)\mathrm{sk}(a) = \mathrm{sk}(a \times b)$ [31]. Thus, (35) and (36) may be written

$$\mathrm{sk}(U)\left(T(\mathbf{F} \times \tau) + \ddot{T}\mathbf{F}\right) = -w_1 - T(\mathbf{F} \times \Omega) \times (\Omega \times U) + \dot{T}(\mathbf{F} \times (\Omega \times U) + (\mathbf{F} \times \Omega) \times U) \quad (39)$$

$$U^T\left(T(\mathbf{F} \times \tau) + \ddot{T}\mathbf{F}\right) = -w_2 - 2\dot{T}U^T(\mathbf{F} \times \Omega) - TU^T\Omega \times (\mathbf{F} \times \Omega). \quad (40)$$

The second-order time derivative of the heave control $\ddot{T}$ enters into the lefthand side (LHS) of these expressions. It is possible to assign the heave control T directly (earlier in the backstepping procedure), however, such an approach leads to an aggressive time-scale separation of the system dynamics [39], [46]. A more balanced control action is obtained by dynamically extending the heave control

$$\ddot{T} = f$$

where f is a new control input. The dynamics $\ddot{T} = f$ are internal to the controller. Since $|U| = 1$ and $I_3 = UU^T - \mathrm{sk}(U)^2$, multiplying (39) by $\mathrm{sk}(U)$ and (40) by U, and adding the two together, one obtains a nondegenerate algebraic relation

$$T\mathrm{sk}(\mathbf{F})\tau + \mathbf{F}f = \mathrm{sk}(U)w_1 - Uw_2 - \mathrm{sk}(U) \times \Big(T(\mathbf{F} \times \Omega) \times (\Omega \times U) - \dot{T} \times (\mathbf{F} \times (\Omega \times U) - (\mathbf{F} \times \Omega) \times U)) + U\left(2\dot{T}U^T(\mathbf{F} \times \Omega) + TU^T\Omega \times (\mathbf{F} \times \Omega)\right). \quad (41)$$

The control inputs τ and f enter directly into the LHS of this equation. If $T \neq 0$, then there are physical control inputs corresponding to any choice of intermediary control signals w_1 and w_2. Since T corresponds to the thrust needed to support a UAV in quasi-stationary flight, $T \approx mg$ for the applications considered. When this condition fails, the rigid-body dynamics are known to be nonholonomic [23], and the backstepping approach proposed in this paper is not applicable.

Remark 3.2: Note that the control $F^T\tau$ is not assigned by (41). This control input is used to control the secondary task (cf. Section III-B).

The control design continues by assigning suitable intermediary inputs w_1 and w_2 to stabilize (33) and (34)

$$w_1 = \frac{k_1^2 k_2 (k_1 k_2 + k_3)}{m^3} \times [m\Omega_\times \delta_3 - k_1 k_2 \delta_3 - (k_1 k_2 + k_3 + k_4)\delta_4]$$
$$w_2 = l_2^2 (l_2 + l_3) \times [l_2 \eta_2 + (l_3 - l_2(l_2 + l_3)) \times \eta_3 - (l_2(l_2 + l_3) + l_4)\eta_4] \quad (42)$$

where k_4 and l_4 are two positive constants (in kg.s^{-1} and s^{-1}, respectively). Substituting the control into the dynamics for δ_4 and η_4 leads to

$$\dot{\delta}_4 = \frac{k_1}{m} Q\delta_1 - \frac{k_1}{m}(k_2 I - Q)\delta_2 - \frac{(k_1 k_2 + k_3)}{m}\delta_3 - \frac{k_4}{m}\delta_4 \quad (43)$$

$$\dot{\eta}_4 = -l_2(l_2 + l_3)\eta_3 - l_4\eta_4. \quad (44)$$

Consequently, choosing

$$S_4 := \frac{1}{2}|\delta_4|^2 + \frac{1}{2}|\eta_4|^2 \quad (45)$$

as the final storage function, one obtains

$$\dot{S}_4 = \frac{k_1}{m}\delta_4^T Q\delta_1 - \frac{k_1}{m}\delta_4^T(k_2 I - Q)\delta_2 - \frac{(k_1 k_2 + k_3)}{m} \times \delta_4^T\delta_3 - \frac{k_4}{m}\delta_4^T\delta_4 - (l_2 + l_3)\eta_4\eta_3 - l_4\eta_4^2. \quad (46)$$

Lemma 3.3: Consider the dynamics defined by (19)–(23). For suitable $\lambda_{\max}, \epsilon > 0$, assume that $Q < \lambda_{\max} I$ and $T > \epsilon$ along closed-loop trajectories of the system, and that *Assumption 2.2* holds. Specify the primary control inputs by (41) and

(42). If the control gains $\{k_1,\dots,k_4\}$, $\{l_2,\dots,l_4\}$ are positive and

$$k_2 > \lambda_{\max}, \quad k_3 > \frac{1}{4}k_1k_2\frac{\lambda_{\max}}{k_2-\lambda_{\max}}$$

then the closed-loop solution exists for all time, and the error signals $\{\delta_1,\dots,\delta_4\}$ and $\{\eta_2,\dots,\eta_4\}$ converge exponentially to zero.

Proof: By continuity, the solutions of the closed-loop system must exist for some period $[0,T_0)$, $T_0>0$. By inspection, the physical inputs F and τ are bounded on any finite time interval, and the total energy of the system is bounded by a linear growth rate in time. As a consequence, the solution to the closed-loop system cannot display finite escape-time behavior, and the solutions exist for all time. Define a candidate Lyapunov function

$$\mathcal{L} = S_{1,2} + S_3 + S_4.$$

From the development (19)–(46), one has

$$\dot{\mathcal{L}} = -\frac{1}{m}X^T\Sigma X$$

where $X=(\delta_1,\dots,\delta_4,\eta_2,\dots,\eta_4)^T$ and

$$\Sigma = \begin{pmatrix} k_1Q & 0 & -\frac{k_1}{2}Q & -\frac{k_1}{2}Q & 0 & 0 & 0\\ 0 & (k_2I-Q) & -\frac{k_1}{2}Q & -\frac{k_1}{2}Q & 0 & 0 & 0\\ -\frac{k_1}{2}Q & -\frac{k_1}{2}Q & k_3 & 0 & 0 & 0 & 0\\ -\frac{k_1}{2}Q & -\frac{k_1}{2}Q & 0 & k_4 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & ml_2 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & ml_3 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & ml_4 \end{pmatrix}.$$

The above quadratic expression is negative definite if and only if (iff) the symmetric matrix Σ is positive definite. This is true iff the principal minors of Σ are positive. It may be directly verified that the gains $k_1,\dots,k_4,l_2,\dots,l_4$ chosen ensure the positiveness of the principal minors of the matrix Σ. The result follows from Lyapunov's theorem [21, Th. 3.1] restricted to the error coordinates. ∎

The error coordinates δ_1 regulate the position of the camera with respect to the target lines (cf. *Lemma 2.3*). The errors δ_2 and η_2 regulate the linear velocity of the camera. The additional error coordinates (δ_3,η_3) and (δ_4,η_4) incorporate information on the attitude kinematics and dynamics, respectively, of the camera. This is natural for the underactuated rigid-body dynamics considered, since the desired motion is obtained by rotating the airframe in order that the fixed thrust direction $\mathbf{F}$ is oriented to provide the necessary force inputs for the linear dynamics.

Although the Lyapunov function $\mathcal{L}$ is positive definite in the error signals, it is only positive semidefinite in the full state. The remaining dynamics of the system are "yaw" rotations around the thrust vector direction $\mathbf{F}$. In certain situations, these dynamics are naturally asymptotically stable; for example, due to an offset between the center of mass and the point at which the thrust is applied, or due to aerodynamic drag effects. In such cases, local stability of the full system follows from Iggidr *et al.* [19] (cf. also [39, Th. 2.24]). Local asymptotic stability may be proved using La Salle's invariance principle. If the remaining dynamics are not asymptotically stable, it is necessary to consider a secondary control task to stabilize the full system.

B. Secondary Control Design

In this section, a secondary control-task error criterion [37] is proposed to stabilize the yaw dynamics around the thrust direction $\mathbf{F}$. The dynamic effects of the secondary control input is directly compensated for in the control input for the primary control design, in order that the stability analysis of *Lemma 3.3* is not affected. This is inherent in the structure of (41). However, the closed-loop dynamics of the primary loop act as an exponentially decaying perturbation to the secondary control analysis. In contrast to the situation encountered in prior work [16], the proposed approach does not require any additional visual information. Indeed, the direction U of the parallel lines provides a natural feature, $q_2^0 \approx U$, to use as the goal for yaw stabilization. In practice, q_2^0 may be computed from image data

$$q_2^0 = \frac{q_2}{|q_2|}, \quad \text{where} \quad q_2 = \sum_{i\le j} a_{ij}(h_i \times h_j)$$

with $a_{ij}=\pm 1$ chosen to ensure that all contributions $h_i\times h_j$ lie approximately in the same direction. The visual feature $q_2^0 \approx U$ inherits dynamics from the motion of the camera

$$\dot{q}_2^0 = -\Omega \times q_2^0. \tag{47}$$

Let $\mathbf{F}_2$ be a reference direction in the body-fixed frame. The control objective considered is to align $\mathbf{F}_2$ as closely as possible with the visual features q_2^0, subject to the constraints imposed by the primary closed-loop response. Set

$$\sigma_1 := q_2^0 - \mathbf{F}_2 \in \mathcal{A}. \tag{48}$$

Taking the derivative of σ_1 yields

$$\begin{aligned}\dot{\sigma}_1 &= -\Omega\times q_2^0 + \Omega\times\mathbf{F}_2 - \Omega\times\mathbf{F}_2\\ &= -\Omega\times\sigma_1 + \mathrm{sk}(\mathbf{F}_2)(\pi_{\mathbf{F}}+\mathbf{F}\mathbf{F}^T)\Omega.\end{aligned} \tag{49}$$

Define the cost

$$W_1 = \frac{1}{2}|\sigma_1|^2. \tag{50}$$

To simplify notation in the sequel set

$$Y := \mathbf{F}_2\times\mathbf{F} \tag{51}$$

a vector orthogonal to the plane span $(\mathbf{F},\mathbf{F}_2)$. Deriving W_1, one obtains

$$\dot{W}_1 = \sigma_1^T\mathrm{sk}(\mathbf{F}_2)\pi_{\mathbf{F}}\Omega + \sigma_1^TY\mathbf{F}^T\Omega. \tag{52}$$

The virtual control input used for the secondary task is the component $\mathbf{F}^T\Omega$. A second error function is introduced

$$\sigma_2 := \frac{\mathbf{F}^T\Omega}{c_1} + Y^T\sigma_1 \tag{53}$$

for $c_1>0$, a positive constant (in s^{-1}). Thus

$$\dot{W}_1 = \sigma_1^T\mathrm{sk}(\mathbf{F}_2)\pi_{\mathbf{F}}\Omega - c_1\sigma_1^TYY^T\sigma_1 + c_1\sigma_1^TY\sigma_2. \tag{54}$$

Define

$$W_2 = \frac{1}{2}|\sigma_2|^2. \tag{55}$$

Recalling the transformation (38), one has

$$\dot{W}_2 = \sigma_2 \frac{\mathbf{F}^T\tau}{c_1} - \sigma_2 Y^T \left(\Omega \times \sigma_1 - \mathrm{sk}(\mathbf{F}_2)\pi_{\mathbf{F}}\Omega\right) - c_1\sigma_2 Y^T\sigma_1 + c_1|Y|^2\sigma_2^2. \tag{56}$$

The control input to this equation is $\mathbf{F}^T\tau$. Recalling (41), it is seen that the primary control assigns $\mathrm{sk}(\mathbf{F})\tau$ and constrains the DOFs of τ orthogonal to $\mathbf{F}$. It follows that the remaining DOF in the attitude control, $\mathbf{F}^T\tau$, is free to be used for the control of the secondary task. Set

$$\mathbf{F}^T\tau := c_1 Y^T\Omega_\times\sigma_1 - c_1 Y^T\mathrm{sk}(\mathbf{F}_2)\pi_{\mathbf{F}}\Omega - c_1\left(c_1|Y|^2 + c_2\right)\sigma_2. \tag{57}$$

Define

$$\mathcal{W} = W_1 + W_2. \tag{58}$$

Then for $c_1, c_2 > 0$ (both in s^{-1}), one has

$$\dot{\mathcal{W}} = \sigma_1^T\mathrm{sk}(\mathbf{F}_2)\pi_{\mathbf{F}}\Omega - c_1\sigma_1^T YY^T\sigma_1 - c_2\sigma_2^2. \tag{59}$$

Lemma 3.4: Consider the dynamics defined by (19)–(23). For suitable $\lambda_{\max}, \epsilon > 0$, assume that $Q < \lambda_{\max} I$ and $T > \epsilon$ along closed-loop trajectories of the system, and that *Assumption 2.2* holds. Specify the primary control inputs by (41) and (42), and the secondary control input by (57). Then

$$(\delta_1, \ldots, \delta_4, \eta_2, \ldots, \eta_4) \to 0, \quad \text{exponentially}$$
$$Y^T\sigma_1 \to 0 \quad \text{and} \quad \sigma_2 \to 0, \quad \text{asymptotically}.$$

Proof: Using the same argument used in the proof of *Lemma 3.3*, it follows that the solutions of the closed-loop system exist and are unique for all time. Setting the primary control error to zero, and inspecting the controlled zero dynamics in the secondary error coordinates introduced, it is easily seen that the zero dynamics of the system are asymptotically stable. Local Lyapunov stability of the full system follows from Iggidr *et al.* [19] (cf. also [39, Th. 2.24]). It remains to show that the closed-loop system is asymptotically stable.

Although the secondary control task perturbs the dynamics of the full system, the control for the primary task (41) directly compensates for any dynamic coupling, and the stability analysis of *Lemma 3.3* remains valid. It follows that bounds $C_0, \alpha > 0$ exist (possibly depending on initial conditions), such that

$$|\mathrm{sk}(F)\Omega| = |\pi_F\Omega| \le C_0 e^{-\alpha t}.$$

This is best seen by inspecting the form of (28) and (29), and noting that exponential bounds will apply to the second derivatives of these expressions, in which the control inputs enter in the form shown. Note that $|\sigma_1| \le |q_2^0| + |U| = 2$, and hence

$$\left|\sigma_1^T\mathrm{sk}(\mathbf{F}_2)\pi_{\mathbf{F}}\Omega\right| \le 2C_0 e^{-\alpha t}.$$

Set

$$\mathcal{U}_t := \mathcal{W} + \frac{2C_0}{\alpha}e^{-\alpha t}$$

to be a time-dependent positive function for secondary control-task error analysis. From (59), along with the discussion above, it follows that

$$\dot{\mathcal{U}}_t \le -c_1\sigma_1^T YY^T\sigma_1 - c_2\sigma_2^2.$$

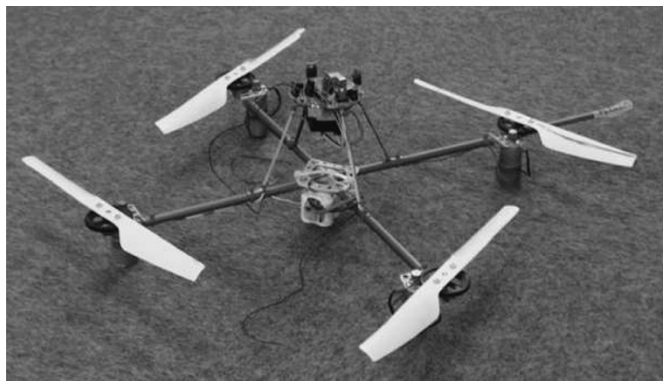

Fig. 4. Prototype (uninstrumented) X4 flyer.

Since $\mathcal{U}_t > 0$ and its derivative is negative semidefinite, $\mathcal{U}_t \to \mathcal{U}^* \ge 0$ converges asymptotically to a minimum value. Applying Barbalat's lemma to the integral of the right-hand side [21, Th. 4.8], it follows that

$$c_1|Y^T\sigma_1|^2 + c_2|\sigma_2|^2 \to 0$$

asymptotically. Combining this with *Lemma 3.3* completes the proof. ■

Lemma 3.4 ensures convergence of q_2^0 into the plane given by span $\{\mathbf{F}_2, \mathbf{F}\}$. Additional constraints on the error cannot be achieved without imposing constraints on the primary control task. By wisely choosing the body-fixed frame direction $\mathbf{F}_2$, the secondary control design proposed can be applied to most desired trajectories. In particular cases, it may be desirable to consider an alternative visual feature to q_2^0 that allows additional freedom in specification of the secondary control task. The two visual features used in the proposed control design, q_1 and q_2^0, have the advantage that they are highly robust to camera calibration error and pixel noise, and are simple and quick to calculate.

IV. Example System and Simulation

In this section, the procedure presented in Section III is applied to an idealized model of the dynamics of an X4 flyer.

An X4 flyer consists of four individual rotors fixed to a rigid cross frame (cf. Fig. 4). It operates as an omnidirectional vehicle capable of quasi-stationary flight. An idealized dynamic model of the X4 flyer [1], [15] is given by the rigid-body equations (1)–(4) along with the external force and torque inputs

$$F = -TE_3^a + mgR^T E_3^a \tag{60}$$
$$\Gamma = \Gamma_1 E_1^a + \Gamma_2 E_2^a + \Gamma_3 E_3^a \tag{61}$$

where (cf. Fig. 5)

$$T = T_{rr} + T_{rl} + T_{fr} + T_{fl} \tag{62}$$
$$\Gamma_1 = d(T_{fr} + T_{fl} - T_{rr} - T_{rl}) \tag{63}$$
$$\Gamma_2 = d(T_{rl} + T_{fl} - T_{rr} - T_{fr}) \tag{64}$$
$$\Gamma_3 = \kappa(T_{fr} + T_{rl} - T_{fl} - T_{rr}). \tag{65}$$

The individual thrust of each motor is denoted $T_{(\cdot)}$, while κ is the constant ratio between thrust and induced couple due to air resistance, and d is the offset of each rotor from the central axis of the flyer.

To simplify the analysis, assume, without loss of generality, that $u = E_x$ is parallel to the x axis of the world frame. The

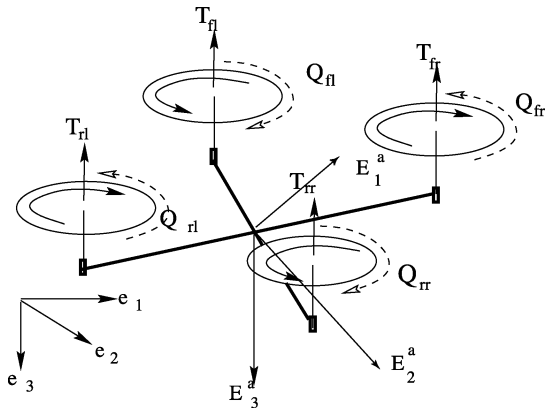

Fig. 5. Force and torque inputs for an X4 flyer.

LHS of (41) that governs how the physical control action enters the system is

$$T\mathrm{sk}(\mathbf{F})\tau + \mathbf{F}f = \begin{pmatrix} T\tau_2 \\ T\tau_1 \\ f \end{pmatrix}. \tag{66}$$

It is clear that as long as $T \neq 0$, the control signals τ_1, τ_2, and $f = \ddot{T}$ may be used to achieve the primary control task. In the case of an X4 flyer, flying with a small forward velocity (i.e., in quasi-stationary flight), the thrust must approximately cancel the gravitational force $T \approx mg$. The yaw control τ_3 is free to stabilize the yaw angle using the second objective (57). The second reference direction $\mathbf{F}_2$ is chosen to be

$$\mathbf{F}_2 = E_1^a. \tag{67}$$

A. Simulation Results

The simulations undertaken consider the case of tracking two parallel lines with a constant velocity. The desired image of the two lines is chosen such that the camera set point is located 2 m above the target. The desired forward velocity is 0.5 m.s^{-1}. The X4 parameters used are $m = 0.6$ kg, $\mathbf{I} = \mathrm{diag}[0.4, 0.4, 0.6]$ kg m^2, $g = 10$ m/s^2, $d = 0.25$ m, and $\kappa = 0.01$.

For the simulation undertaken, limiting bounds were chosen on the eigenvalues of the matrix Q [cf. (17)]

$$\lambda_{\min} = 0.01 \text{ m}^{-1} \text{ and } \lambda_{\max} = 5 \text{ m}^{-1}.$$

The region in task space for which these bounds remain valid is a compact set, denoted Υ, that excludes the target points (cf. Fig. 6). Extensive simulation has shown that the form of the region Υ shown in Fig. 6 is indicative for sets of linear features lying roughly on a planar surface. By construction, the closed-loop analysis undertaken in Section III is valid for any trajectory that lies within the region. In fact, only the bound on $\lambda_{\max}$ is required for *Lemma 3.3*. Geometrically, the bound on $\lambda_{\max}$ corresponds to excluding a small ball, of radius approximately $1/\lambda_{\max}$ around each target point in the (x, z) plane (cf. Fig. 6). This is due to the dependence of Q [cf. (13)] on the inverse distance of the target point. Physically, the bound is associated with the sensitivity of the image feature to motion in Cartesian space; a large bound $\lambda_{\max}$ indicates that a small motion in Cartesian space may cause a large change of q in image space.

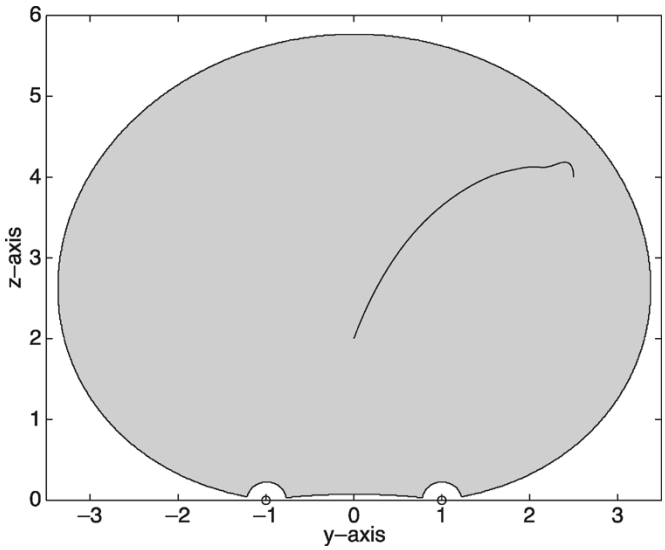

Fig. 6. Cross section in the (y, z) plane of region for which the bounds $\lambda_{\max} < 5$, $\lambda_{\min} > 0.01$ are valid for two parallel line targets (marked by circles), with u parallel to the x axis (pointing out of the page). Trajectory shown is the projection onto the (x, z) plane of the ideal closed-loop response of the system.

The lower bound $\lambda_{\min}$ leads to the outer boundary of Υ. The outer boundary is approximately spherical around a distributed target cluster. In the case that the targets are co-planar, as is the case in this example, then $\lambda_{\min}(Q) = 0$ on the plane containing the target points, and the spherical boundary is pinched together along the target plane (cf. Fig. 6). Physically, the bound is associated with the inverse sensitivity of image feature to motion in Cartesian space; a small value of $\lambda_{\min}$ indicates that a large motion in Cartesian space will cause only a small change of q in image space. A similar situation in a classical IBVS control design would be of concern, since the linearizing control in classical IBVS design involves inverting the image Jacobian containing the matrix Q [cf. (15)]. Implementing a classical IBVS controller leads to a demanded control input of the order $1/\lambda_{\min}$, close to the boundaries of Υ. The nonlinear approach taken does not attempt to invert the matrix Q, and this problem is avoided. Indeed, when distant from the target cluster, the image error is bounded above by

$$|\delta_1| \leq |q^*| + |q_1| \leq 2n$$

and consequently, the desired velocity is bounded above. By scaling the image feature, this velocity upper bound can be arbitrarily assigned and used as a practical saturation on the closed-loop system response. Indeed, for initial conditions distant from the target cluster and goal vectors lying within Υ, the image error is dominated by the image feature q_1 (that points directly toward the target cluster), and the closed-loop system will converge toward Υ with velocity equal to the saturation bound chosen. It is only within Υ that the exponential convergence associated with the linearized error dynamics inherent in a classical backstepping procedure will be observed.

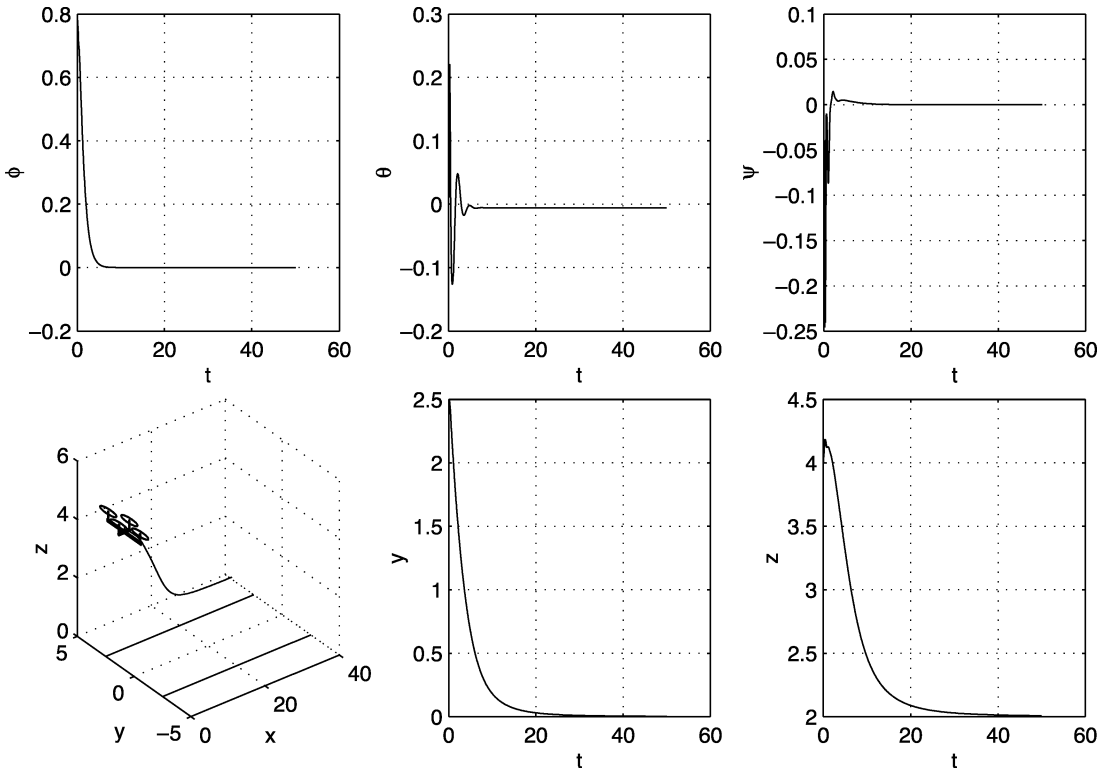

Fig. 7. Evolution of the closed-loop trajectory for the proposed control design in the ideal case.

An estimate of the size of the region Υ can be computed by considering the case of two image points and a test point symmetrically above two target points at height h. The minimum sensitivity of change in Q will occur in the vertical direction and will be equal to the rate of change of $|q|$ in this direction

$$\lambda_{\min}(Q) = \left| \frac{d}{dh} q(h) \right|.$$

Direct geometric construction yields

$$\lambda_{\min} \approx \frac{4r^2}{(4h_{\max}^2 + r^2)^{\frac{3}{2}}} \tag{68}$$

where $h_{\max}$ is the maximum height of the region while r is the distance between the target points. For the case shown in the example, choosing $\lambda_{\min} = 0.01$ leads to a maximum height estimate $h_{\max} \approx 5.75$ m. Additional target points lying on the line between the outer two points will act to increase the size of Υ. The size of the set shown in Fig. 6 depends (in a natural manner) on the size of the target.

Given suitable bounds $\lambda_{\max}$ and $\lambda_{\min}$ it is possible to make the following qualitative observations.

1) Within Υ, the Cartesian response and the image-based response will be comparable, and the backstepping control design ensures locally exponential convergence.
2) For $\lambda_i(Q) < \lambda_{\min}$, the image error will be dominated by the component q_1 and the system will converge toward the target cluster with constant velocity.
3) For $\lambda_i(Q) > \lambda_{\max}$, the convergence proof contained in *Lemma 3.3* does not hold. This is due to the very high sensitivity of the image features to movement in Cartesian space, and the failure of the stability analysis does not necessarily indicate the presence of extreme or undesirable behavior of the system in Cartesian space.

For the simulation undertaken, the initial condition was chosen such that the X4 was initially in stable stationary flight in the specified region (Fig. 6)

$$\xi_0 = (5 \;\; 2.5 \;\; -4)^T, \quad R_0 = I_3, \quad \dot{\xi}_0 = \dot{\Omega} = 0.$$

According to standard aeronautical conventions, height is measured down relative to the aircraft, and hence, the height of the X4 is negative with respect to the world frame. The control gains used were $k_1 = 0.4$ kg.m.s^{-1}, $k_2 = 6$ kg^{-1}, $k_3 = 3.5$ kg.s^{-1}, $k_4 = 3.5$ kg.s^{-1}, and $l_2 = l_3 = l_4 = 1$ s^{-1}. These gains satisfy the conditions of *Lemmas 3.3* and *3.4*.

In Fig. 7, the performance of the algorithm in the ideal case is shown. In this experiment, the camera is modeled as a continuous system with no pixel quantization or frame-rate delay. No disturbances to the system dynamics are considered. Fig. 7 clearly shows the exponential convergence of the error signals. The X4 achieves perfect tracking after a short transient. It is interesting to note that the roll and pitch of the X4 converge to zero, despite the fact that the X4 is flying forward with constant velocity. This is natural, since the simulation does not consider aerodynamic drag effects.

The idealized model [(1)–(4) with external force and torque (60) and (61)] is at best an approximation of the full dynamics of the X4 flyer. There are four main causes of error in the dynamic model used in the control design.

1) The expression of the force F in (60) is affected by dynamic perturbations

$$F = -TE_3^a + R^T mgE_3^a + \Delta. \tag{69}$$

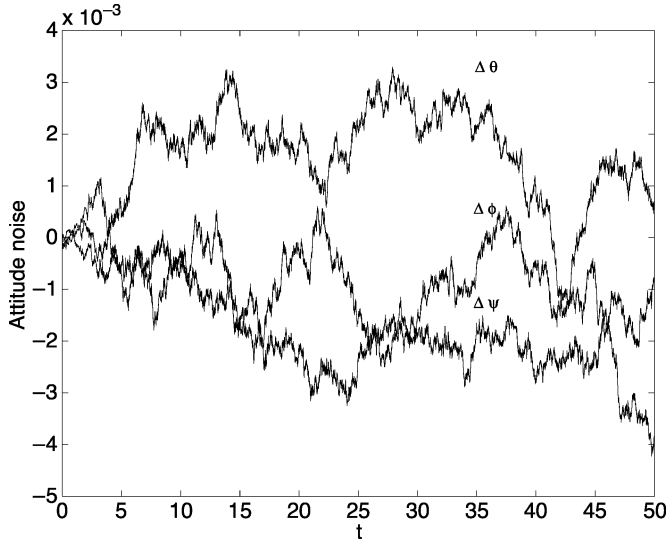

Fig. 8. High-frequency noise added to the attitude estimates.

The perturbation Δ contains unmodeled lift and drag effects, wind gust effects, and intrinsic parasitic forces linked to coupling between torque control and translation forces (cf. the discussion of zero dynamics in recent work on flying robots [14], [22], [24], [42]).

2) Several rotors operating in close proximity to each other can interfere with each other's airflow characteristics and lead to disturbances in the torque control inputs.
3) The state measurements of velocity, attitude, and angular rate are subject to sensor noise that is only partly reduced by filtering.
4) The discrete image sampling, both in time (frame rate) and space (pixels), will generate both dynamic and static error in the image feature δ_1 and σ_1.

In practice, the best performance that can be expected is a practical stabilization of the system to a neighborhood of the desired trajectory. A full analysis of the practical stability margins of the system is beyond the scope of this paper (a similar analysis is undertaken in [24]). The simulations presented demonstrate the effects of the important perturbations and show viability of the control design.

In the second simulation, the more important disturbances to the X4 system were modeled as best as possible. The camera is modeled by a discrete image sampled at a frame rate of 30 Hz. High-frequency noise is added directly to the estimates of the attitude. The magnitude of the noise signal used is shown in Fig. 8. In addition to the noise terms, a deterministic (low frequency) noise-like disturbance is added to the torque control

$$\tau = \tau_{\text{control}} + \tau_{\text{noise}}. \tag{70}$$

This simulates the "surge"-type effects that occur due to interaction of inflow and wakes of the four rotors

$$\tau_{\text{noise}} = 0.05 \times \left[0 \quad \left(0.1\cos\left(\frac{t}{10}\right)\sin\left(\frac{t}{5}\right)\right) \quad \left(\cos\left(\frac{t}{5}\right)\sin\left(\frac{t}{10}\right)\right)\right]. \tag{71}$$

Finally, an aerodynamic drag term of the form

$$D = -\text{abs}(V)^T \begin{pmatrix} 0.3 & 0 & 0 \\ 0 & 0.3 & 0 \\ 0 & 0 & 0.5 \end{pmatrix} V,\ \text{abs}(V) = \begin{pmatrix} |V_1| \\ |V_2| \\ |V_3| \end{pmatrix} \tag{72}$$

was added to the translation dynamics. Due to the form of the X4 flyer, the drag in the z axis is greater than the drag in the x, y directions.

The effects of the disturbances in the closed-loop response is clear from Fig. 9. The attitude response of the system is perturbed by the measurement noise and surge disturbances in the torque control. Due to the backstepping design, there is an inherent low-pass filter between the attitude dynamics and the position control, and the high-frequency noise in the attitude dynamics is highly attenuated in the translation dynamics. The low-frequency surge effects do affect the translation dynamics. This can be seen by comparing the offset present in the roll ϕ due to surge disturbance and the slight offset in the y axis tracking (an error of around 18 cm).

The proposed backstepping control may be considered as a nonlinear analog of a proportional-derivative-type controller. Consequently, it is expected that the drag term added to the perturbation of the translation dynamics will lead to a slight offset in the tracking error η_2. To provide a clear picture of this effect, a close-up view of the pitch and velocity response are plotted in Fig. 10, along with the (ideal) desired set points. Based on the model of the drag term, the choice of gain $l_2 = 1$, and the practical stability provided by a quadratic Lyapunov function, one may approximate the relationship between the actual velocity achieved by the closed-loop system and the desired set point velocity by

$$v_{\text{desired}} - v_{\text{actual}} \approx 0.2 v_{\text{actual}}^2. \tag{73}$$

Solving the quadratic for v_{actual} for the given set point $v_{\text{desired}} = 0.5$, one obtains an estimate $v_{\text{actual}} \approx 0.442$ m/s. Regarding Fig. 10, this corresponds to the observed velocity v_x. If exact tracking is required and the drag coefficient is known, it is possible to use the above analysis to anticipate the tracking offset for a given set point, and compensate with a slight increase in the set point velocity. This approach is effectively a combined feedforward/feedback control architecture. Alternatively, if the drag coefficient is unknown or slowly time varying, then adding an integrator or developing a full adaptive control design is possible. In practice, the tracking error is so small that the proposed control leads to acceptable closed-loop performance.

V. Conclusions

In this paper, an IBVS control design was proposed to track parallel linear features for underactuated rigid-body dynamics. The approach taken uses the binormalized Euclidean Plücker coordinates to represent linear features. The tasks of line tracking and velocity regulation are decoupled, and the passivity-like properties of rigid-body dynamics are preserved separately in each control task. A secondary control design

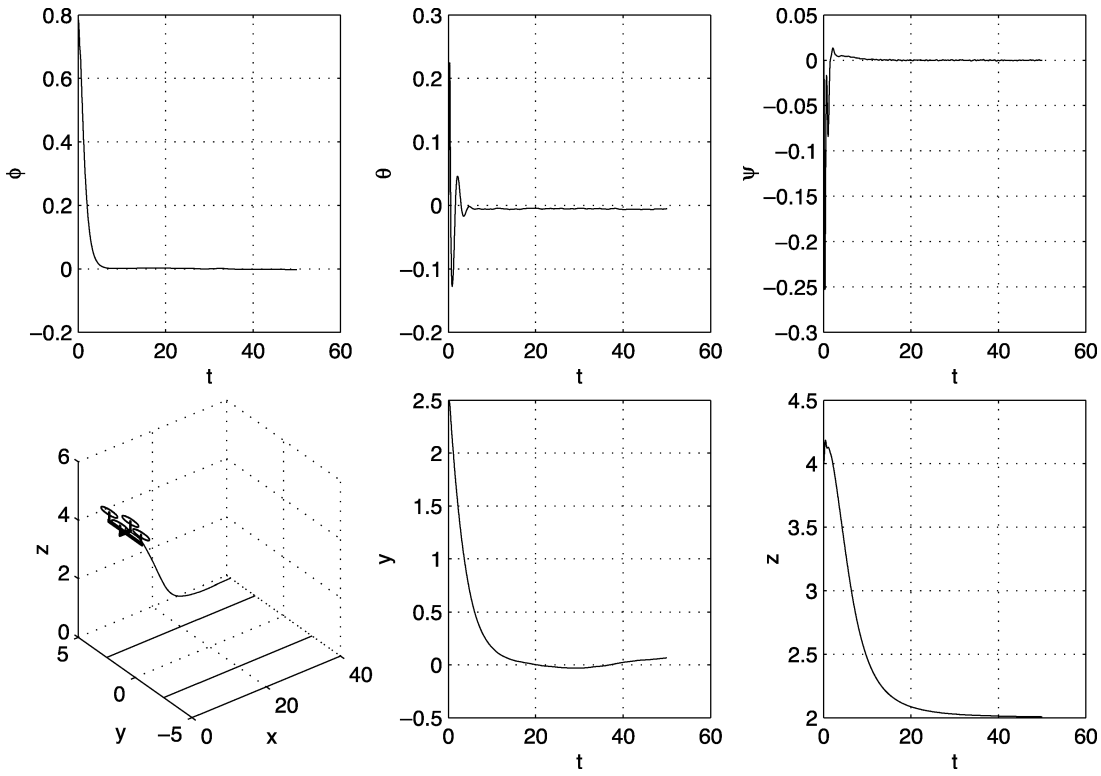

Fig. 9. Closed-loop response for the proposed control design in the presence of dynamic perturbations.

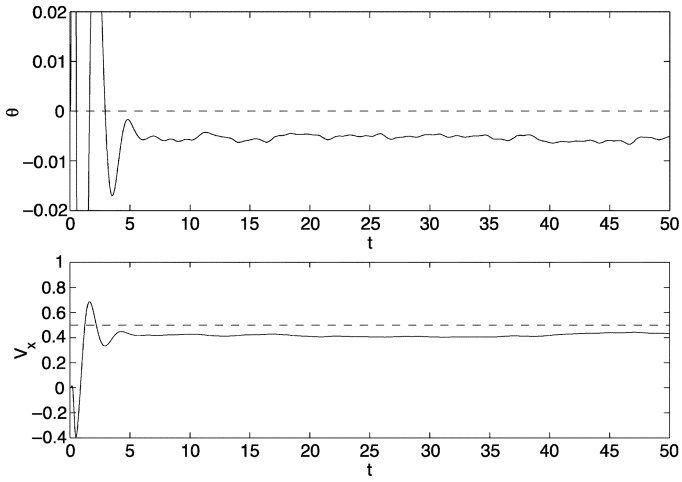

Fig. 10. Steady-state response of the pitch θ and the velocity v_x for the perturbed stimulation. The presence of unmodeled drag disturbance leads to a tracking error in the velocity and an associated constant pitch offset in the direction of motion.

is undertaken to regulate the remaining degree of liberty in the attitude dynamics typical of dynamic models of UAVs capable of quasi-stationary flight. This paper forms a part of an ongoing worldwide effort to control small UAVs in real-world conditions, using low-cost light sensor systems. In future work, we will consider dynamic manoeuvres, such as landing an aeroplane, where the path tracked is not a constant line parallel to the data observed. We are also working on implementing the results on prototype flying robots and actively investigating issues associated with the sensitivity of the control design and image features used.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers of this paper for providing excellent comments and perspective on the material.

REFERENCES

[1] E. Altug, J. Ostrowski, and R. Mahony, "Control of a quadrotor helicopter using visual feedback," in *Proc. IEEE Int. Conf. Robot. Autom.*, Washington, DC, May 2002, pp. 72–77.

[2] O. Amidi, T. Kanade, and R. Miller, "Vision-based autonomous helicopter research at Carnegie Mellon Robotics Institute (1991–1998)," in *Robust Vision for Vision-Based Control of Motion*, M. Vincze and G. D. Hager, Eds. New York: IEEE Press/SPIE, 1999, ch. 15, pp. 221–232.

[3] N. Andreff and B. Espiau, "Revisiting Plüker coordinates in vision-based control," in *Proc. Symp. Adv. Robot Kinematics*, Caldes de Maravella, Spain, 2002, pp. 265–274.

[4] J. R. Azinheira, P. Rives, and J. R. H. Carvalho, "Visual servo control for the hovering of an outdoor robotic airship," in *Proc. IEEE Int. Conf. Robot. Autom.*, Washington, DC, May 2002, pp. 2787–2792.

[5] J. Chahl and M. Srinivasan, "Panoramic vision system for imaging, ranging and navigation in three dimensions," in *Proc. Int. Conf. Field, Service Robot*, Pittsburgh, PA, Aug. 1999, Session 4B.

[6] F. Chaumette, "La relation vision-commande: Théorie et application à des tâches robotiques," Ph.D. dissertation (in French), Thèse de l'Univ. de Rennes 1, IRISA, Univ. de Rennes, Rennes, France, 1990.

[7] F. Conticelli, B. Allotta, and P. K. Khosla, "Image-based visual servoing of nonholonomic mobile robots," in *Proc. Conf. Decision, Control*, Phoenix, AZ, 1999, pp. 3496–3501.

[8] C. Eck, J. Chapuis, and H. P. Geering, "Software-supported design and evaluation of low-cost navigation units," in *Proc. 8th St. Petersburg Int. Conf. Integ. Navigat. Syst.*, St. Petersburg, Russia, 2001, pp. 163–172.

[9] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Trans. Robot. Autom.*, vol. 8, pp. 313–326, Jun. 1992.

[10] B. Espiau, "Effect of camera calibration errors on visual servoing in robotics," in *3rd International Symposium on Experimental Robotics*. ser. Lecture Notes in Control and Information Sciences, T. Yoshikawa and F. Miyasaki, Eds. New York: Springer-Verlag, 1993, vol. 200.

[11] M. Dahlen, E. Frazzoli, and E. Feron, "Trajectory tracking control design for autonomous helicopters using a backstepping algorithm," in *Proc. Amer. Control Conf.*, Chicago, IL, 2000, pp. 4102–4107.
[12] O. Faugeras, *Three-Dimentional Computer Vision—A Geometric Viewpoint*. Cambridge, MA: MIT Press, 1993.
[13] T. Fitzgibbons and E. Nebot, "Application of vision in simultaneous localization and mapping," in *Intelligent Autonomous Systems*. Amsterdam, The Netherlands: IOS Press, 2002, pp. 92–99.
[14] E. Frazzoli, M. A. Dahleh, and E. Feron, "Real-time motion planning for agile autonomous vehicles," *AIAA J. Guid., Control, Dynam.*, vol. 5, no. 1, pp. 116–129, 2002.
[15] T. Hamel, R. Mahony, R. Lozano, and J. Ostrowski, "Dynamic modeling and configuration stabilization for an X4 flyer," in *Proc. Int. Federat. Autom. Control Symp.*, Barcelona, Spain, 2002, [CD-ROM].
[16] T. Hamel and R. Mahony, "Visual servoing of an under-actuated dynamic rigid-body system: An image-based approach," *IEEE Trans. Robot. Autom.*, vol. 18, pp. 187–198, Apr. 2002.
[17] K. Hosada and M. Asada, "Versatile visual servoing without knowledge of true Jacobian," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots, Syst.*, Munich, Germany, 1994, pp. 186–193.
[18] S. Hutchinson, G. Hager, and P. Cork, "A tutorial on visual servo control," *IEEE Trans. Robot. Autom.*, vol. 12, pp. 651–670, Oct. 1996.
[19] A. Iggidr, B. Kalitine, and R. Outbib, "Semi-definite Lyapunov function, stability and stabilization," *Math. Control, Signals, Syst.*, vol. 9, pp. 95–106, 1996.
[20] A. Isidori, L. Marconi, and A. Serrani, "Robust nonlinear motion control of a helicopter," in *Proc. 40th Conf. Decision, Control*, Orlando, FL, 2001, pp. 4586–4591.
[21] H. K. Khalil, *Nonlinear Systems*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1996.
[22] T. J. Koo and S. Sastry, "Output tracking control design of a helicopter model based on approximate linearization," in *Proc. IEEE Conf. Decision, Control*, 1998, pp. 3635–3640.
[23] N. E. Leonard, "Stability of a bottom-heavy underwater vehicle," *Automatica*, vol. 33, pp. 331–346, 1997.
[24] R. Mahony, T. Hamel, and A. Dzul, "Hover control via approximate Lyapunov control for a model helicopter," in *Proc. Conf. Decision, Control*, Phoenix, AZ, 1999, pp. 533–534.
[25] R. Mahony and T. Hamel, "Visual servoing using linear features for under-actuated rigid body dynamics," in *Proc. Int. Conf. Intell. Robots, Syst.*, Maui, HI, 2001, pp. 1153–1158.
[26] E. Malis, F. Chaumette, and S. Boudet, "2-1/2-D visual servoing," *IEEE Trans. Robot. Autom.*, vol. 15, pp. 238–250, Apr. 1999.
[27] A. Maruyama and M. Fujita, "Visual feedback control of rigid body motion based on dissipation theoretical approach," in *Proc. 38th Conf. Decision, Control*, Phoenix, AZ, 1999, pp. 4161–4166.
[28] Y. Ma, J. Kosecka, and S. Sastry, "Vision-guided navigation for a nonholonomic mobile robot," *IEEE Trans. Robot. Autom.*, vol. 15, pp. 521–536, Jun. 1999.
[29] ——, "Linear differential algorithms for motion recovery: A geometric approach," *Int. J. Computer Vis.*, vol. 36, no. 1, pp. 71–89, 2000.
[30] B. Mettler, *Identification Modeling and Characteristics of Miniature Rotorcraft*. Dordrecht, The Netherlands: Kluwer, 2003.
[31] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. Boca Raton, FL: CRC Press, 1994.
[32] N. Papanikolopoulos, P. K. Khosla, and T. Kanade, "Adaptive robot visual tracking," in *Proc. Amer. Control Conf.*, 1991, pp. 962–967.
[33] J. A. Piepmeier, "A dynamic quasi-Newton method for model independent visual servoing," Ph.D. dissertation, Georgia Inst. Technol., Atlanta, GA, Jul. 1999.
[34] R. Pissard-Gibollet and P. Rives, "Applying visual servoing techniques to control of a mobile hand-eye system," in *Proc. IEEE Int. Conf. Robot. Autom.*, Nagasaki, Japan, 1995, pp. 166–171.
[35] J. Plücker, "On a new geometry of space," *Philos. Trans. Roy. Soc. London*, vol. 155, 1865.
[36] H. Pottman, M. Peternell, and B. Ravani, "Approximation in line space: Application in robot kinematics and surface reconstruction," in *Advances in Robot Kinematics: Analysis and Control*, J. Lenarçiç and M. L. Husty, Eds. Norwell, MA: Kluwer, 1998, pp. 403–412.
[37] C. Samson, M. Le Borgne, and B. Espiau, *Robot Control: The Task Function Approach*. Oxford, U.K.: Oxford Univ. Press, 1991.
[38] S. Saripalli, J. F. Montgomery, and G. S. Sukhatme, "Vision-based autonomous landing of an unmanned aerial vehicle," in *Proc. Int. Conf. Robot. Autom.*, Washington, DC, 2002, pp. 2799–2804.
[39] R. Sepulchre, M. Janković, and P. Kokotović, *Constructive Nonlinear Control*. London, U.K.: Springer-Verlag, 1997.
[40] O. Shakernia, Y. Ma, T. J. Koo, and S. Sastry, "Landing an unmanned air vehicle: Vision-based motion estimation and nonlinear control," *Asian J. Control*, vol. 1, no. 3, pp. 128–146, 1999.
[41] L. G. Shapiro and G. C. Stockman, *Computer Vision*. Upper Saddle River, NJ: Prentice-Hall, 2001.
[42] H. Sira-Ramirez, R. Castro-Linares, and E. Liceaga-Castro, "A Liouvillian systems approach for the trajectory planning-based control of helicopter models," *Int. J. Robust, Nonlinear Control*, vol. 10, pp. 301–320, 2000.
[43] S. Soatto and P. Perona, "Structure-independent visual motion control on the essential manifold," in *Proc. IFAC Symp. Robot Control*, Capri, Italy, 1994, pp. 869–876.
[44] M. W. Spong and M. Vidyasagar, Eds., *Robot Dynamics and Control*. New York: Wiley, 1989.
[45] M. Srinivasan, J. Chahl, K. Weber, S. Venkatesh, M. Nagle, and S. Zhang, "Robot navigation inspired by principles of insect vision," *Robot. Auton. Syst.*, vol. 26, pp. 203–216, 1999.
[46] A. R. Teel, "A nonlinear small gain theorem for the analysis of control systems with saturation," *IEEE Trans. Autom. Control*, vol. 41, pp. 1256–1270, Sep. 1996.
[47] E. Zergeroglu, D. Dawson, M. de Queiroz, and S. Nagarkatti, "Robust visual-servo control of robot manipulators in the presence of uncertainty," in *Proc. 38th Conf. Decision, Control*, Phoenix, AZ, 1999, pp. 3628–3634.
[48] H. Zhang and J. P. Ostrowski, "Visual servoing with dynamics: Control of an unmanned blimp," *IEEE Trans. Robot. Autom.*, vol. 18, pp. 199–208, Apr. 2002.

Robert Mahony (S'92–M'95) obtained a science degree in applied mathematics and geology in 1989, and the Ph.D. degree in systems engineering in 1994, both from the Australian National University (ANU), Canberra, Australia.

Between 1994 and 1997, he was a Research Fellow with the Cooperative Research Centre for Robust and Adaptive Systems, Research School of Information Sciences and Engineering, ANU. From 1997 to 1999, he was a Postdoctoral Fellow with the CNRS Laboratory for Heuristics Diagnostics and Complex Systems (Heudiasyc), Compiègne University of Technology, Compiègne, France. Between 1999 and 2001, he was a Logan Fellow with the Department of Engineering and Computer Science, Monash University, Melbourne, Australia. Since July 2001, he has been a Senior Lecturer in Mechatronics with the Department of Engineering, ANU. His research interests are in nonlinear control theory with applications in mechanical systems and motion systems, mathematical systems theory, and geometric optimization techniques with applications in linear algebra and digital signal processing.

Tarek Hamel received the Bachelor of Engineering degree from the Institut d'Electronique et d'Automatique d'Annaba, Annaba, Algeria, in 1991, and the Ph.D. degree in robotics from the University of Technology of Compiègne, Compiègne, France, in 1995.

After two years as a Research Assistant with the University of Technology of Compiègne, he joined the Centre d'Etudes de Mècanique d'Iles de France, Courcouronnes, France, in 1997 as an Assistant Professor. In 2001–2002, he spent one year as a CNRS Researcher at the CNRS Laboratory for Heuristics Diagnostics and Complex Systems (Heudiasyc), Compiègne University of Technology. Since 2003, he has been a Professor with the I3S UNSA-CNRS Laboratory, University of Nice-Sophia Antipolis, Nice, France. His research interests include control theory and robotics with particular focus on nonlinear control and vision-based control. He is involved in application of these techniques to the control of unmanned aerial vehicles and mobile robots.