






Research Article

Vision-Based Nonlinear Control of Quadrotors Using the Photogrammetric Technique

José Trinidad Guillén-Bonilla ¹, Claudia Carolina Vaca García ²,
Stefano Di Gennaro ^{3,4}, María Eugenia Sánchez Morales ²,
and Cuauhtémoc Acosta Lúa ^{2,4}

¹Centro Universitario de Ciencias Exactas e Ingeniería–UdG, Blvd. Gral. Marcelino García Barragán 1421, Olímpica, 44430, Guadalajara, Jalisco, Mexico

²Departamento de Ciencias Tecnológicas, Universidad de Guadalajara, Centro Universitario de La Ciénege, Av. Universidad 1115, 47820, Ocotlán, Jalisco, Mexico

³Department of Information Engineering, Computer Science and Mathematics, University of L'Aquila, Via Vetoio, L'Aquila, Coppito, 67100, Italy

⁴Center of Excellence DEWS, University of L'Aquila, Via Vetoio, L'Aquila, Coppito, 67100, Italy

Correspondence should be addressed to Cuauhtémoc Acosta Lúa; cuauhtemoc.acosta@cuci.udg.mx

Received 12 June 2020; Revised 11 August 2020; Accepted 29 September 2020; Published 17 November 2020

Academic Editor: Aimé Lay-Ekuakille

Copyright © 2020 José Trinidad Guillén-Bonilla et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a controller designed via the backstepping technique, for the tracking of a reference trajectory obtained via the photogrammetric technique. The dynamic equations used to represent the motion of the quadrotor helicopter are based on the Newton–Euler model. The resulting quadrotor model has been divided into four subsystems for the altitude, longitudinal, lateral, and yaw motions. A control input is designed for each subsystem. Furthermore, the photogrammetric technique has been used to obtain the reference trajectory to be tracked. The performance and effectiveness of the proposed nonlinear controllers have been tested via numerical simulations using the Pixhawk Pilot Support Package developed for Matlab/Simulink.

1. Introduction

The technology of unmanned aerial vehicles (UAVs) is developing rapidly, based on the integration of many technologies in the mechanical structure, energy management, communication, control, etc, [1]. UAVs are widely used in many applications in civilian as well as in military contexts. For instance, UAVs provide great advantages in geographical information acquisition and safety purposes, as well as significant benefits when used for inspection and detection purposes. Furthermore, UAVs are becoming more and more accessible due to technological developments that enable their increasing use in different and broad application areas.

Different mathematical models can be used to design a controller for UAVs. In [2–4], a Newton–Euler model was

presented. Furthermore, the studies in [5, 6] considered quaternions to describe the angular kinematics, whilst the study in [7] applied the Euler–Lagrange equations to obtain the whole quadrotor mathematical model. Regarding the control of quadrotors, many control techniques were proposed. The linear quadratic regulator control and proportional integral derivative control were proposed in [8, 9]. However, the stability of these methods cannot be guaranteed when the quadrotor moves away from its equilibrium configuration. Compared to linear control, the nonlinear control can ensure global stable flight for a quadrotor. Examples of nonlinear control techniques used to design a controller for a quadrotor are the sliding mode and the backstepping techniques [10–13]. The backstepping and adaptive control [14] were applied to the quadrotor flight control. In [15, 16], an observer-based adaptive fuzzy

backstepping controller was designed for trajectory tracking of a quadrotor subject to wind gusts and parametric uncertainties. In [17], a robust adaptive attitude tracking control for a quadrotor was proposed. In [18], a control trajectory tracking was designed via the sliding mode technique and using neural networks to optimize the controller parameters through a network learning process which is based on the control process.

To design vision-based controllers, researchers used different methods for the recognition of objects, trajectories, road lanes, etc. In [19], an INS/vision-based autonomous landing system on stationary platforms was proposed, whereas in [20], mobile platforms were used for the landing of quadrotors. In [21], the position of a quadrotor was estimated by the detection of a set of concentric circles. In [22], a controller, based on computer vision techniques applied to helipad recognition, was proposed, in which the visual recognition of a black and white pattern of the helipad was exploited. In [23], a computer vision system was developed for the automatic estimation of the position and attitude of a helipad fixed on a mobile platform. In [24], a trajectory tracking problem for a vision-based quadrotor control system via super twisting sliding mode controller was proposed. Also, the study in [25] proposed a neuroadaptive integral robust controller for the tracking of ground moving targets in the presence of various uncertainties, via an Image-Based Visual Servo (IBVS) framework. The authors of [26] presented a predictive control algorithm for autonomous approaches of quadrotor helicopters to a window using a reference image captured with a photographic camera. The target is selected by an operator in a reference image which is sent to the vehicle. Besides, the authors of [27] proposed an adaptive sliding mode controller based on the backstepping technique for a tracking problem using a monocular algorithm to obtain the accurate location information of the quadrotor and its reference. In the same context, the work in [28, 29] developed a vision-based attitude dynamic surface controller, constructing an IBVS to ensure that the visual target remains in the camera's field of view all the time.

In this paper, a controller for the tracking of a reference trajectory is designed via the backstepping technique. The quadrotor model is divided into four subsystems for the altitude, longitudinal, lateral, and yaw motions, and a control input is designed for each subsystem. Furthermore, the photogrammetric technique is used to obtain the reference trajectory to be tracked. The performance and effectiveness of the proposed nonlinear controllers are tested via numerical simulations using the simulation software called Pixhawk Pilot Support Package (PSP), which predicts accurately the real dynamic quadrotor helicopter behavior. The main contributions of this article are as follows:

- (i) The backstepping technique is used to design a controller capable of tracking a reference position and yaw of a quadrotor

- (ii) The photogrammetric technique is used to obtain the reference trajectory
- (iii) The performance and effectiveness of the proposed controller have been tested in PSP

The paper is organized as follows: Section 2 introduces the description and the mathematical model of the quadrotor. In Section 3, the control problem is formulated, whilst in Section 4, the nonlinear controller is designed. In Section 5, the reference trajectory is obtained via the photogrammetric technique, and some numerical simulations implemented in PSP are presented. Some comments conclude the paper.

2. Mathematical Model of a Quadrotor

The quadrotor considered in this work consists of a rigid frame equipped with four rotors. The rotors generate the propeller force $F_i = b\omega_{p,i}^2$, proportional to the propeller angular velocity $\omega_{p,i}$, $i = 1, 2, 3, 4$. Propellers 1 and 3 rotate counterclockwise, and propellers 2 and 4 rotate clockwise.

Let us indicate with $RC(O, e_1, e_2, e_3)$ and $R\Gamma(\Omega, \epsilon_1, \epsilon_2, \epsilon_3)$ the frames fixed with the Earth and the quadrotor, respectively, and Ω is coincident with the center of mass of the quadrotor (see Figure 1). The absolute position of quadrotor in RC is described by $p = (x, y, z)^T$, whereas its attitude is described by the Euler angles $\alpha = (\phi, \theta, \psi)^T$, with $\phi, \theta, \psi \in (-\pi/2, \pi/2)$ being the pitch, roll, and yaw angles, respectively. The sequence 3-2-1 is here considered [30]. Moreover, $v = (v_x, v_y, v_z)^T$ and $\omega = (\omega_1, \omega_2, \omega_3)^T$ are the linear and angular velocities of the center of mass of the quadrotor, expressed in RC and in R Γ , respectively.

The translation dynamics, expressed in RC, and rotation dynamics, expressed in R Γ , of the quadrotor are

$$\begin{aligned} \dot{p} &= v, \\ \dot{v} &= \frac{1}{m} \mathcal{R}(\alpha) F_{\text{prop}} + \frac{1}{m} F_{\text{grav}} + \frac{1}{m} F_d, \end{aligned} \quad (1)$$

$$\dot{\alpha} = M(\alpha)\omega,$$

$$\dot{\omega} = J^{-1}(-\tilde{\omega}J\omega + \tau_{\text{prop}} - \tau_{\text{gyro}} + M_d),$$

where m is the mass of the quadrotor, $J = \text{diag}\{J_x, J_y, J_z\}$ (expressed in R Γ) is the inertia matrix of the quadrotor, and

$$\tilde{\omega} = \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix}, \quad (2)$$

is the so-called dyadic representation of ω . Furthermore,

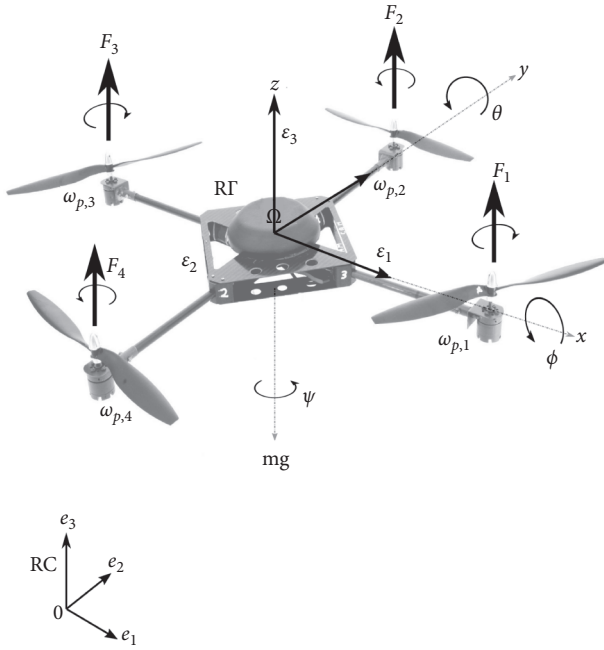


FIGURE 1: Quadrotor orientation using Euler angles.

$$\begin{aligned}
 F_{\text{prop}} &= \begin{pmatrix} 0 \\ 0 \\ \sum_{i=1}^4 F_i \end{pmatrix}, \\
 F_{\text{grav}} &= \begin{pmatrix} 0 \\ 0 \\ -mg \end{pmatrix}, \\
 \tau_{\text{prop}} &= \begin{pmatrix} \tau_1 \\ \tau_3 \\ \tau_2 \end{pmatrix} = \begin{pmatrix} l(F_2 - F_4) \\ l(F_3 - F_1) \\ c(F_1 - F_2 + F_3 - F_4) \end{pmatrix} = \begin{pmatrix} lu_3 \\ lu_2 \\ u_4 \end{pmatrix},
 \end{aligned} \quad (3)$$

are the input forces and moments produced by the propellers, where l is the distance between center of mass CG to the rotor shaft and c is the drag factor. Moreover, F_{grav} in (3) is the force due to the gravity, expressed in RC.

The vectors expressed in RF are transformed into vectors in RC by the rotation matrix

$$\mathcal{R}(\alpha) = \begin{pmatrix} c_\phi c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{pmatrix}, \quad (4)$$

where $c_* = \cos(*)$, $s_* = \sin(*)$, and $* = \phi, \theta, \psi$.

The angular velocity dynamics are expressed using the matrix

$$M(\alpha) = \begin{pmatrix} 1 & s_\phi \text{tg}\theta & c_\phi \text{tg}\theta \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi \text{sc}\theta & c_\phi \text{sc}\theta \end{pmatrix}, \quad (5)$$

with $\text{tg}_* = \tan(*)$, $\text{sc}_\diamond = \sec(*)$, and $* = \phi, \theta, \psi$.

Considering small angles, matrix (5) can be approximated by the identity matrix, i.e., $M(\alpha) = I_{3 \times 3}$. This assumption is justified by the fact that the movements of the quadrotor are slow and soft [4].

The rolling torque τ_1 is produced by the forces F_2 and F_4 . Similarly, the pitching torque τ_3 is produced by the forces F_1 and F_3 . Due to Newton's third law, the propellers produce a yawing torque τ_2 on the body of the quadrotor, in the opposite direction of the propeller rotation. Moreover,

$$\tau_{\text{gyro}} = \sum_{i=1}^4 (-1)^{i+1} J_p \omega_p \tilde{\omega} \epsilon_3, \quad (6)$$

is the gyroscopic torque due to the propeller rotations, with J_p being the propeller moment of inertia with respect to its rotation axis, and $\omega_p = \omega_{p,1} - \omega_{p,2} + \omega_{p,3} - \omega_{p,4}$ is the so-called rotor relative speed. Finally, F_d and M_d are the forces and torques due to the external disturbances, here assumed negligible.

Now, using (1) and (3) and under the small angle assumption, the mathematical model of the quadrotor can be expressed by

$$\begin{aligned}
 \ddot{x} &= \frac{1}{m} (c_\phi s_\theta c_\psi + s_\phi s_\psi) u_1, \\
 \ddot{y} &= \frac{1}{m} (c_\phi s_\theta s_\psi - s_\phi c_\psi) u_1, \\
 \ddot{z} &= -g + \frac{1}{m} c_\phi c_\theta u_1, \\
 \ddot{\phi} &= \frac{J_y - J_z}{J_x} \dot{\theta} \dot{\psi} - \frac{J_p}{J_x} \omega_p \dot{\theta} + \frac{l}{J_x} u_3, \\
 \ddot{\theta} &= \frac{J_z - J_x}{J_y} \dot{\phi} \dot{\psi} + \frac{J_p}{J_y} \omega_p \dot{\phi} + \frac{l}{J_y} u_2, \\
 \ddot{\psi} &= \frac{J_x - J_y}{J_z} \dot{\phi} \dot{\theta} + \frac{1}{J_z} u_4,
 \end{aligned} \quad (7)$$

where the control variables u_j , $j = 1, 2, 3, 4$, are defined as (3). The values of the parameters used in (7) are defined in Table 1.

3. Formulation of the State Feedback Control Problem

The control problem is to ensure the asymptotic converge of the variables $\chi = (x, y, z, \psi)$ to some reference trajectories $\chi_{\text{ref}} = (x_{\text{ref}}, y_{\text{ref}}, z_{\text{ref}}, \psi_{\text{ref}})$. In view of the control of the quadrotor, the following coordinate change is considered: $\phi = \omega_\phi$, $\theta = \omega_\theta$, and $\psi = \omega_\psi$ ($\omega_\phi \approx \omega_1$, $\omega_\theta \approx \omega_2$, $\omega_\psi \approx \omega_3$), and the system of equations (7) became

TABLE 1: Quadrotor's coefficients and variables.

Variable	Value
m	1.1 kg
l	0.223 m
J_x	6.825×10^{-3} kg m ²
J_y	6.825×10^{-3} kg m ²
J_z	12.39×10^{-3} kg m ²
J_p	6×10^{-5} kg m ²
g	9.81 m/s ²
c	1.1×10^{-6} N s ² rad ⁻²
x	m
y	m
z	m
v_x	m/s
v_y	m/s
v_z	m/s
ϕ	deg
θ	deg
ψ	deg
ω_ϕ	deg/s
ω_θ	deg/s
ω_ψ	deg/s

$$\begin{aligned}
\dot{x} &= v_x, \\
\dot{v}_x &= \frac{1}{m} (c_\phi s_\theta c_\psi + s_\phi s_\psi) u_1, \\
\dot{y} &= v_y, \\
\dot{v}_y &= \frac{1}{m} (c_\phi s_\theta s_\psi - s_\phi c_\psi) u_1, \\
\dot{z} &= v_z, \\
\dot{v}_z &= -g + \frac{1}{m} c_\phi c_\theta u_1, \\
\dot{\phi} &= \omega_\phi, \\
\dot{\omega}_\phi &= \frac{J_y - J_z}{J_x} \omega_\theta \omega_\psi - \frac{J_p}{J_x} \omega_p \omega_\theta + \frac{l}{J_x} u_3, \\
\dot{\theta} &= \omega_\theta, \\
\dot{\omega}_\theta &= \frac{J_z - J_x}{J_y} \omega_\phi \omega_\psi + \frac{J_p}{J_y} \omega_p \omega_\phi + \frac{l}{J_y} u_2, \\
\dot{\psi} &= \omega_\psi, \\
\dot{\omega}_\psi &= \frac{J_x - J_y}{J_z} \omega_\phi \omega_\theta + \frac{l}{J_z} u_4.
\end{aligned} \tag{8}$$

Clearly, model (8) is composed of rotational and translational dynamics. Figure 2 shows that the input control u_1 does not influence the rotational dynamics but only the translational ones and can be used to impose $z \rightarrow z_{\text{ref}}$. Furthermore, in the spirit of the backstepping technique, the (ϕ, θ, ψ) can be used to impose $x \rightarrow x_{\text{ref}}$, $y \rightarrow y_{\text{ref}}$. In

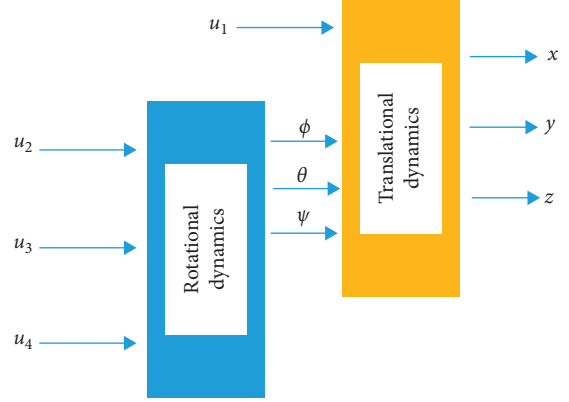


FIGURE 2: Connection between rotational and translational dynamics.

turn, (ϕ, θ, ψ) are influenced by the inputs $u_3, u_2,$ and u_4 , which can be used to impose $\phi \rightarrow \phi_{\text{ref}}$, $\theta \rightarrow \theta_{\text{ref}}$, and $\psi \rightarrow \psi_{\text{ref}}$. Hence, the following four subsystems will be considered: the altitude subsystem S_1 , the latitudinal subsystem S_2 , the longitudinal subsystem S_3 , and the yaw subsystem S_4 :

$$\begin{aligned}
S_1 &= \begin{cases} \dot{z} = v_z \\ \dot{v}_z = -g + \frac{1}{m} c_\phi c_\theta u_1, \end{cases} \\
S_2 &= \begin{cases} \dot{x} = v_x \\ \dot{v}_x = \frac{1}{m} (c_\phi s_\theta c_\psi + s_\phi s_\psi) u_1 \\ \dot{\theta} = \omega_\theta \\ \dot{\omega}_\theta = \frac{J_z - J_x}{J_y} \omega_\phi \omega_\psi + \frac{J_p}{J_y} \omega_p \omega_\phi + \frac{l}{J_y} u_2, \end{cases} \\
S_3 &= \begin{cases} \dot{y} = v_y \\ \dot{v}_y = \frac{1}{m} (c_\phi s_\theta s_\psi - s_\phi c_\psi) u_1 \\ \dot{\phi} = \omega_\phi \\ \dot{\omega}_\phi = \frac{J_y - J_z}{J_x} \omega_\theta \omega_\psi - \frac{J_p}{J_x} \omega_p \omega_\theta + \frac{l}{J_x} u_3, \end{cases} \\
S_4 &= \begin{cases} \dot{\psi} = \omega_\psi \\ \dot{\omega}_\psi = \frac{J_x - J_y}{J_z} \omega_\phi \omega_\theta + \frac{l}{J_z} u_4. \end{cases}
\end{aligned} \tag{9}$$

The controller design for each subsystem will be carried out in the following section.

4. Design of the Nonlinear Controller

The translation dynamics in equation (7) are dependent on the ϕ , θ , and ψ angles. In the next sections, a nonlinear control is used for each subsystem to solve the control problem.

4.1. Altitude Control. In this section are established the altitude control and its stability proof via Lyapunov function. For this purpose, it is considered that the altitude control S_1 subsystem is

$$\begin{aligned} \dot{z} &= v_z, \\ \dot{v}_z &= -g + \frac{1}{m} c_\phi c_\theta u_1. \end{aligned} \quad (10)$$

The control aim is to maintain the quadrotor at a desired constant altitude z_{ref} , with $v_{z,\text{ref}} = \dot{z}_{\text{ref}}$. To this aim, setting the tracking error for the altitude control as

$$\begin{aligned} e_{z,1} &= z - z_{\text{ref}}, \\ e_{z,2} &= v_z - v_{z,\text{ref}} + k_{z,1} e_{z,1}, \end{aligned} \quad (11)$$

the dynamic errors are

$$\begin{aligned} \dot{e}_{z,1} &= -k_{z,1} e_{z,1} + e_{z,2}, \\ \dot{e}_{z,2} &= -g + \frac{1}{m} c_\phi c_\theta u_1 - \dot{v}_{z,\text{ref}} \\ &\quad + k_{z,1} (e_{z,2} - k_{z,1} e_{z,1} + v_{z,\text{ref}} - \dot{z}_{\text{ref}}). \end{aligned} \quad (12)$$

The proposed altitude control u_1 is

$$\begin{aligned} u_1 &= \frac{m}{c_\phi c_\theta} \left(g + \dot{v}_{z,\text{ref}} - k_{z,1} (e_{z,2} - k_{z,1} e_{z,1} + v_{z,\text{ref}} - \dot{z}_{\text{ref}}) \right. \\ &\quad \left. - k_3 e_{z,1} - k_4 e_{z,2} \right), \end{aligned} \quad (13)$$

where $c_\phi \neq 0$ and $c_\theta \neq 0$. Therefore, using altitude control (13) in (12), the altitude error dynamics became

$$\begin{aligned} \begin{pmatrix} \dot{e}_{z,1} \\ \dot{e}_{z,2} \end{pmatrix} &= A_z \begin{pmatrix} e_{z,1} \\ e_{z,2} \end{pmatrix}, \\ A_z &= \begin{pmatrix} -k_{z,1} & 1 \\ -k_3 & -k_4 \end{pmatrix}. \end{aligned} \quad (14)$$

For the stability analysis of system (14), the following Lyapunov function candidate is proposed:

$$\begin{aligned} V_z(e_z) &= \frac{1}{2} e_z^T P_z e_z, \\ e_z &= \begin{pmatrix} e_{z,1} \\ e_{z,2} \end{pmatrix}, \\ P_z &= \begin{pmatrix} P_{z,11} & P_{z,12} \\ P_{z,12} & P_{z,22} \end{pmatrix}, \end{aligned} \quad (15)$$

where $P_z = P_z^T > 0$. Deriving Lyapunov function (15), one obtains

$$\dot{V}_z(e_z) = e_z^T (A_z^T P_z + P_z A_z) e_z, \quad (16)$$

where P_z is taken as solution of the Lyapunov equation

$$\begin{aligned} A_z^T P_z + P_z A_z &= -Q_z, \\ Q_z &= \begin{pmatrix} q_{1,z} & 0 \\ 0 & q_{1,z} \end{pmatrix}, \end{aligned} \quad (17)$$

with $Q_z = Q_z^T > 0$ being fixed. Using (17) into (16), one finally obtains

$$\dot{V}_z(e_z) = -e_z^T Q_z e_z \leq \lambda_{\min}^{Q_z} \|e_z\|^2 \leq -\alpha V_z(e_z), \quad (18)$$

with $\alpha = 2\lambda_{\min}^{Q_z} / \lambda_{\max}^{P_z}$ and P_z as solution of (17). Therefore, by [31], $V_z(e_z) \leq e^{-\alpha t} V(e_z(0))$, so that the tracking errors converge globally exponential to zero.

4.2. Longitudinal Motion Control. In this section, the longitudinal motion is studied making use of the subsystem S_2 and using the backstepping technique. One considers the tracking error $e_{x,1} = x - x_{\text{ref}}$, where $v_{x,\text{ref}} = \dot{x}_{\text{ref}}$, and the error dynamics

$$\dot{e}_{x,1} = v_x - \dot{x}_{\text{ref}}, \quad (19)$$

or, equivalently,

$$\begin{aligned} \dot{e}_{x,1} &= -k_{x,1} e_{x,1} + (v_x - \dot{x}_{\text{ref}} + k_{x,1} e_{x,1}) \\ &= -k_{x,1} e_{x,1} + e_{x,2}, \quad k_{x,1} > 0, \end{aligned} \quad (20)$$

where

$$e_{x,2} = v_x - v_{x,\text{ref}}, \quad (21)$$

is the velocity error and

$$v_{x,\text{ref}} = \dot{x}_{\text{ref}} - k_{x,1} e_{x,1}, \quad (22)$$

is the reference velocity. Successively, one works out the velocity error dynamics as

$$\begin{aligned}
\dot{e}_{x,2} &= \frac{1}{m}(c_\phi s_\theta c_\psi + s_\phi s_\psi)u_1 - \ddot{x}_{\text{ref}} + k_{x,1}\dot{e}_{x,1} \\
&= -k_{x,2}e_{x,2} + \frac{1}{m}c_\phi s_\theta c_\psi u_1 + \frac{1}{m}s_\phi s_\psi u_1 - \ddot{x}_{\text{ref}} - k_{x,1}^2 e_{x,1} \\
&\quad + (k_{x,1} + k_{x,2})e_{x,2},
\end{aligned} \tag{23}$$

with $k_{x,2} > 0$, where u_1 is given by (13). For the stabilization of $e_{x,2}$, the reference value for s_θ can be fixed as follows:

$$s_{\theta,\text{ref}} = \frac{m}{c_\phi c_\psi u_1} (\ddot{x}_{\text{ref}} + k_{x,1}^2 e_{x,1} - (k_{x,1} + k_{x,2})e_{x,2}) - \frac{s_\phi s_\psi}{c_\phi c_\psi}, \tag{24}$$

where $c_\phi \neq 0$ and $c_\psi \neq 0$, so that

$$\begin{aligned}
\dot{e}_{x,2} &= -k_{x,2}e_{x,2} + \frac{1}{m}c_\phi c_\psi u_1 e_{\theta,1}, \\
e_{\theta,1} &= s_\theta - s_{\theta,\text{ref}}, \\
\theta_{\text{ref}} &= \arcsin s_{\theta,\text{ref}}.
\end{aligned} \tag{25}$$

To impose such a reference for θ , one considers the tracking error $e_{\theta,1}$ and its derivative:

$$\dot{e}_{\theta,1} = -k_{\theta,1}e_{\theta,1} + c_\theta \omega_\theta - \dot{\theta}_{\text{ref}} c_{\theta,\text{ref}} + k_{\theta,1}e_{\theta,1} = -k_{\theta,1}e_{\theta,1} + c_\theta e_{\theta,2}, \tag{26}$$

with $k_{\theta,1} > 0$, where

$$\omega_{\theta,\text{ref}} = \frac{1}{c_\theta} (\dot{\theta}_{\text{ref}} c_{\theta,\text{ref}} - k_{\theta,1}e_{\theta,1}), \tag{27}$$

is the reference angular velocity and

$$e_{\theta,2} = \omega_\theta - \omega_{\theta,\text{ref}}, \tag{28}$$

is the angular velocity error. Therefore, one works out the velocity error dynamics as

$$\dot{e}_{\theta,2} = \frac{J_z - J_x}{J_y} \omega_\phi \omega_\psi + \frac{J_p}{J_y} \omega_p \omega_\phi + \frac{l}{J_y} u_2 - \dot{\omega}_{\theta,\text{ref}}. \tag{29}$$

Finally, one determines the input control u_2 as

$$u_2 = \frac{J_z - J_x}{l} \omega_\phi \omega_\psi - \frac{J_p}{l} \omega_p \omega_\phi + \frac{J_y}{l} \dot{\omega}_{\theta,\text{ref}} - k_{\theta,2} \frac{J_y}{l} e_{\theta,2}, \tag{30}$$

so that the velocity error dynamics become

$$\dot{e}_{\theta,2} = -k_{\theta,2}e_{\theta,2}, \tag{31}$$

with $k_{\theta,2} > 0$.

4.3. Lateral Motion Control. In this section is designed the controller for the lateral motion along the y -axis of the quadrotor, using the same procedure followed in Section 4.2. One starts with the tracking error

$$e_{y,1} = y - y_{\text{ref}}, \tag{32}$$

and the dynamics

$$\begin{aligned}
\dot{e}_{y,1} &= v_y - \dot{y}_{\text{ref}} = -k_{y,1}e_{y,1} + (v_y - \dot{y}_{\text{ref}} + k_{y,1}e_{y,1}), \quad k_{y,1} > 0, \\
&= -k_{y,1}e_{y,1} + e_{y,2}, \quad e_{y,2} = v_y - v_{y,\text{ref}},
\end{aligned} \tag{33}$$

with $v_{y,\text{ref}} = \dot{y}_{\text{ref}} - k_{y,1}e_{y,1}$. The dynamics of $e_{y,2}$ are

$$\begin{aligned}
\dot{e}_{y,2} &= \dot{v}_y - \dot{v}_{y,\text{ref}} = -k_{y,2}e_{y,2} - \frac{1}{m}c_\psi u_1 e_{\phi,1}, \\
s_{\phi,\text{ref}} &= \frac{m}{c_\psi u_1} (-\dot{y}_{\text{ref}} - k_{y,1}^2 e_{y,1} + (k_{y,1} + k_{y,2})e_{y,2}) + \frac{s_\psi}{c_\psi} c_\phi s_\theta, \\
e_{\phi,1} &= s_\phi - s_{\phi,\text{ref}}, \\
\phi_{\text{ref}} &= \arcsin s_{\phi,\text{ref}},
\end{aligned} \tag{34}$$

with $k_{y,2} > 0$, $c_\psi \neq 0$, and u_1 given by (13). Furthermore, the angular error dynamics are

$$\begin{aligned}
\dot{e}_{\phi,1} &= -k_{\phi,1}e_{\phi,1} + c_\phi \omega_\phi - \dot{\phi}_{\text{ref}} c_{\phi,\text{ref}} + k_{\phi,1}e_{\phi,1} \\
&= -k_{\phi,1}e_{\phi,1} + c_\phi e_{\phi,2},
\end{aligned} \tag{35}$$

with $k_{\phi,1} > 0$ and

$$\begin{aligned}
e_{\phi,2} &= \omega_\phi - \omega_{\phi,\text{ref}}, \\
\omega_{\phi,\text{ref}} &= \frac{1}{c_\phi} (\dot{\phi}_{\text{ref}} c_{\phi,\text{ref}} - k_{\phi,1}e_{\phi,1}).
\end{aligned} \tag{36}$$

Finally, the angular velocity error dynamics are

$$\dot{e}_{\phi,2} = \frac{J_y - J_z}{J_x} \omega_\theta \omega_\psi - \frac{J_p}{J_x} \omega_p \omega_\theta + \frac{l}{J_x} u_3 - \dot{\omega}_{\phi,\text{ref}}, \tag{37}$$

so that the control input is

$$u_3 = -\frac{J_y - J_z}{l} \omega_\theta \omega_\psi + \frac{J_p}{l} \omega_p \omega_\theta + \frac{J_x}{l} \dot{\omega}_{\phi,\text{ref}} - k_{\phi,2} \frac{J_x}{l} e_{\phi,2}, \tag{38}$$

with $k_{\phi,2} > 0$.

4.4. Yaw Motion Control. Considering the subsystem S_4 , and the errors

$$e_{\psi,1} = \psi - \psi_{\text{ref}}, \tag{39}$$

$$e_{\psi,2} = \omega_\psi - \omega_{\psi,\text{ref}},$$

one works out the error dynamics as

$$\begin{aligned}
\dot{e}_{\psi,1} &= \omega_\psi - \dot{\psi}_{\text{ref}} = -k_{\psi,1}e_{\psi,1} + e_{\psi,2}, \\
\omega_{\psi,\text{ref}} &= \dot{\psi}_{\text{ref}} - k_{\psi,1}e_{\psi,1}, \\
\dot{e}_{\psi,2} &= \frac{J_x - J_y}{J_z} \omega_\phi \omega_\theta + \frac{1}{J_z} u_4 - \dot{\omega}_{\psi,\text{ref}}, \\
\dot{\omega}_{\psi,\text{ref}} &= \ddot{\psi}_{\text{ref}} + k_{\psi,1}^2 e_{\psi,1} - k_{\psi,1} e_{\psi,2},
\end{aligned} \tag{40}$$

with $k_{\psi,1} > 0$. Therefore, with the control

$$u_4 = -(J_x - J_y)\omega_\phi\omega_\theta + J_z\dot{\omega}_{\psi,\text{ref}} - k_{\psi,2}J_z e_{\psi,2}, \quad k_{\psi,2} > 0, \quad (41)$$

one finally obtains the yaw error dynamics as

$$\begin{pmatrix} \dot{e}_{\psi,1} \\ \dot{e}_{\psi,2} \end{pmatrix} = \begin{pmatrix} -k_{\psi,1} & 1 \\ 0 & -k_{\psi,2} \end{pmatrix} \begin{pmatrix} e_{\psi,1} \\ e_{\psi,2} \end{pmatrix}. \quad (42)$$

The global exponential stability can be inferred analogously to what is done in Section 4.1.

5. Simulation Results

The photogrammetric technique has been used for measuring the important physical quantities in both ground and flight testing, including attitude, position, and shape of objects. To generate the trajectory reference

$$p_{\text{ref}}(t) = (x_{\text{ref}}(t), y_{\text{ref}}(t), z_{\text{ref}}(t)), \quad (43)$$

a photogrammetric technique was applied to analyze a natural scenery, with

$$\begin{aligned} x_{\text{ref}}(t) &= f_1(x(t), y(t), z(t)), \\ y_{\text{ref}}(t) &= f_2(x(t), y(t), z(t)), \\ z_{\text{ref}}(t) &= f_3(x(t), y(t), z(t)), \end{aligned} \quad (44)$$

and with f_i , $i = 1, 2, 3$, functions of $(x(t), y(t), z(t))$. The selected scenario is a serial array of trees, where all trees have approximately the same separation. Figures 3 and 4 show the trees' frontal and lateral views, respectively. Using the photogrammetry, the following estimations were determined: $W_f = 0.85$ m for the trees' width, $T_t = 2.72$ m for the distance between the two trees, and $W_{s,1} = 0.75$ m and $W_{s,2} = 0.522$ m for the widths of the two trees.

For this application, the reference trajectory has two important characteristics:

- (1) The altitude is constant
- (2) The reference trajectory is periodic over the x - y plane
- (3) The reference trajectory avoids the collision between the quadrotor and the trees

Therefore, the reference position was set equal to

$$\begin{aligned} p_{\text{ref}}(t) &= \begin{pmatrix} x_{\text{ref}}(t) \\ y_{\text{ref}}(t) \\ z_{\text{ref}}(t) \end{pmatrix} \\ &= \begin{pmatrix} v_{x,\text{ref}}t \\ A_{\text{ref}} \sin(2\pi f_{\text{ref}} x_{\text{ref}}(t)) \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} 0.5t \\ 4.25 \sin(0.250\pi x_{\text{ref}}(t)) \\ 1 \end{pmatrix}, \end{aligned} \quad (45)$$

with $v_{x,\text{ref}} = 0.5$ m/s, A_{ref} being the amplitude, and f_{ref} being the frequency of the periodical function, such as



FIGURE 3: Frontal view of the natural scene.



FIGURE 4: Lateral view of the natural scene.

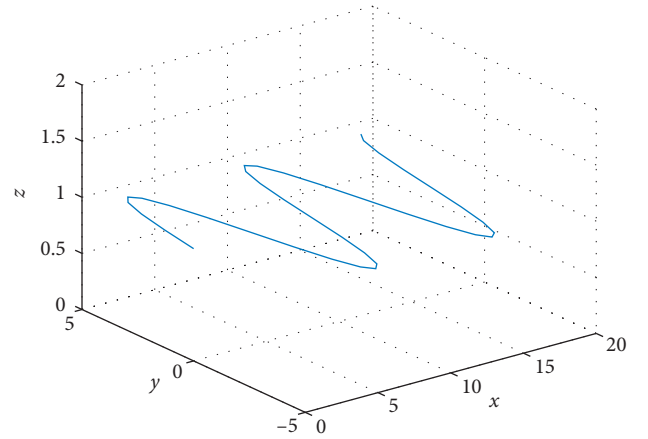


FIGURE 5: Reference trajectory $p_{\text{ref}}(t)$.

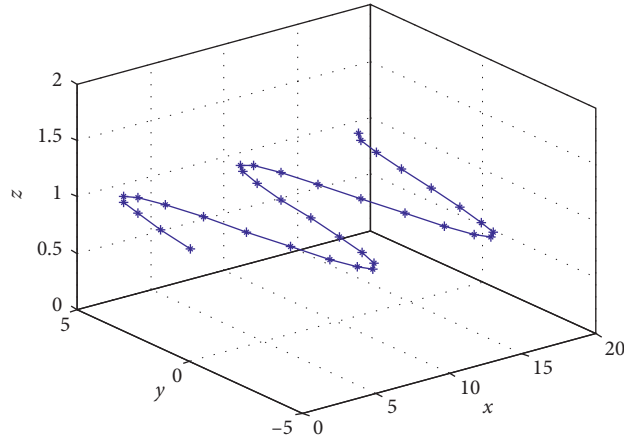
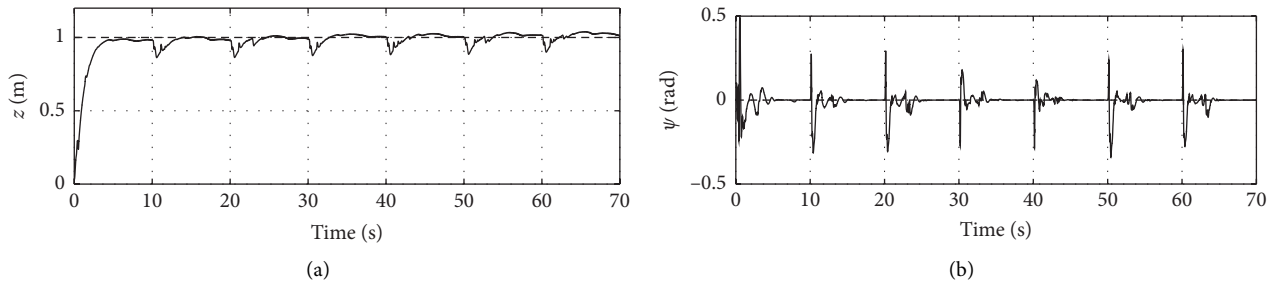
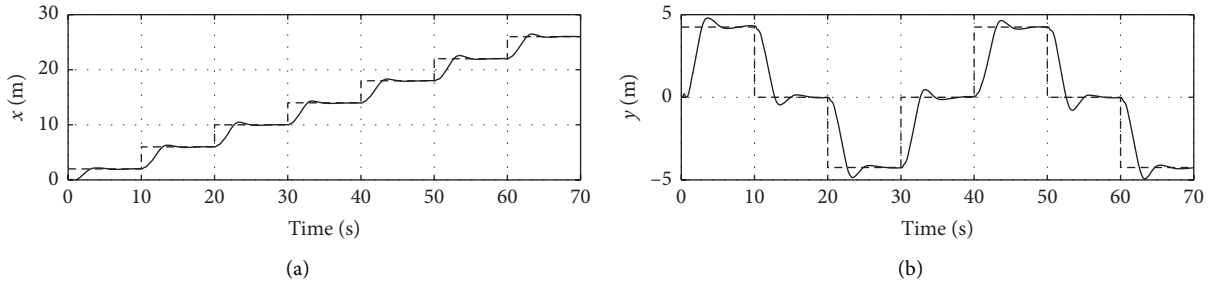
TABLE 2: Controllers' gains values.

$k_{z,1} = 0.5$	$k_3 = 120$	$k_4 = 120$	$q_{1,z} = 1$
$k_{x,1} = 5$	$k_{x,2} = 2$	$k_{\theta,1} = 25$	$k_{\theta,2} = 46$
$k_{y,1} = 5$	$k_{y,2} = 2$	$k_{\phi,1} = 25$	$k_{\phi,2} = 32$
$k_{\psi,1} = 5$	$k_{\psi,2} = 9$		

$$f_{\text{ref}} = \frac{1}{T_{\text{ref}}} = \frac{1}{2(W_{s,1} + W_{s,2} + T_t)} = 0.125 \text{ cycles/m}, \quad (46)$$

where T_{ref} is the period. Figure 5 shows the reference trajectory for $x_{\text{ref}} \in [0, 18]$ m.

The behavior of the controllers designed in Section 4 has been tested with numerical simulations on the quadrotor described by equation (1). The parameters used are given in Table 1, whilst the variables and gains used in the controllers

FIGURE 6: Sampled reference trajectory p_{ref} .FIGURE 7: (a) Quadrotor's altitude $z(t)$ (solid) and reference altitude $z_{\text{ref}}(t)$ (dashed); (b) quadrotor's yaw angle $\psi(t)$ (solid) and yaw angle reference $\psi_{\text{ref}}(t)$ (dashed).FIGURE 8: (a) Longitudinal motion of the UAV x (solid) and the reference $x_{1,\text{ref}}$ (dash); (b) latitudinal motion of the UAV y (solid) and the reference $y_{1,\text{ref}}$ (dash).

are given in Table 2. In order to simulate the controller for each subsystem, the software developed by the PSP for Simulink was used. The reason for choosing the PSP software was due to the performance of this software in predicting the dynamic quadcopter behavior, which is very close to the real behavior. In fact, the PSP software is proven to represent accurately the quadrotor dynamics. In order to appropriately implement the controllers, the reference trajectory was sampled as in Figure 6.

The simulations results of the closed-loop system are shown in Figures 7 and 8, where the quadrotor's initial conditions have been set equal to $p(0) = (0, 0, 0)^T$,

$v(0) = (0, 0, 0)^T$, $\alpha(0) = (0, 0, 0)^T$, and $\omega(0) = (0, 0, 0)^T$. Figure 7 shows the quadrotor altitude and yaw, $z(t)$ and $\psi(t)$, ensured by the controllers (13) and (18). Regarding the latter, the most significant tracking error is 0.5 rad immediately after the initial time.

Figure 8 shows the quadrotor x and y positions ensured by the controllers (30) and (38). The x position achieves its reference value in 3 s, and the y position achieves its reference value in 6 s. The overall controller performance is accurate and fast. Moreover, this performance could be improved using also integral terms in the controllers.

6. Conclusion

In this paper, a controller based on the backstepping technique has been designed for the position and yaw control of a quadrotor helicopter. The quadrotor dynamics have been divided into four subsystems, each with a control input and an output variable. The overall controller leads to satisfactory results. The photogrammetric technique on a scenario with fixed obstacles, due to some trees, has been used to determine the scene geometry and to fix the position reference trajectory. The numerical simulations of the proposed controllers have been implemented using the PSP, ensuring an accurate approximation of the real quadrotor dynamics, and the results show a good performance and effectiveness of the proposed control law. Future work will regard the real implementation of the proposed controller.

Data Availability

The figures, tables, and other data used to support this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was partially supported by the European Project ECSEL–JURIA–2018 “Comp4Drones” and by the Project “Coordination of Autonomous Unmanned Vehicles for Highly Complex Performances,” Executive Program of Scientific and Technological Agreement between Italy (Ministry of Foreign Affairs and International Cooperation, Italy) and Mexico (Mexican International Cooperation Agency for the Development), SAAP3.

References

- [1] W. He, T. Wang, X. He, L. H. Yang, and O. Kaynak, “Dynamical modeling and boundary vibration control of a rigid–flexible wing system,” *IEEE/ASME Transaction on Mechatronics*, vol. 99, 2020 In press.
- [2] S. Bouabdallah, P. Murrieri, and R. Siegwart, “Design and control of an indoor micro quadrotor,” in *Proceedings IEEE International Conference on Robotics and Automation, IEEE International Conference on Robotics and Automation – ICRA*, pp. 1–6, New Orleans, LA, USA, May 2004.
- [3] Y. Zeng and L. Zhao, “Parameter identification for unmanned four–rotor helicopter with nonlinear model,” in *Proceedings of 2014 IEEE Chinese Guidance, Navigation and Control Conference*, pp. 922–926, Yantai, China, August 2014.
- [4] A. Nagaty, S. Saeedi, C. Thibault, M. Seto, and H. Li, “Control and navigation framework for quadrotor helicopters,” *Journal of Intelligent & Robotic Systems*, vol. 70, no. 1–4, pp. 1–12, 2013.
- [5] O. Magnussen, M. Ottestad, and G. Hovland, “Experimental validation of a quaternion–based attitude estimation with direct input to a quadcopter control system,” in *Proceedings of the International Conference on Unmanned Aircraft Systems – ICUAS*, pp. 48–485, Atlanta, GA, USA, May 2013.
- [6] R. G. Valenti, I. Dryanovski, and J. Xiao, “Keeping a good attitude: a quaternion–based orientation filter for imus and margs,” *Sensors*, vol. 15, pp. 19302–19330, 2015.
- [7] M. E. Antonio–Toledo, A. Y. Alanis, and E. N. Sanchez, “Robust neural decentralized control for a quadrotor UAV,” in *Proceedings of the International Joint Conference on Neural Networks – IJCNN*, pp. 714–719, 2016.
- [8] B. Panomruttanarug, K. Higuchi, and F. Mora–Camino, “Attitude control of a quadrotor aircraft using lqr state feedback controller with full order state observer,” in *Proceeding of International Conference on Instrumentation, Control and Information Technology – SICE*, pp. 2041–2046, Beijing, China, October 2013.
- [9] P. E. I. Pounds, D. R. Bersak, and A. M. Dollar, “Stability of small–scale UAV helicopters and quadrotors with added payload mass under PID control,” *Autonomous Robots*, vol. 33, no. 1–2, pp. 129–142, 2012.
- [10] L. Luque–Vega, B. Castillo–Toledo, and A. G. Loukianov, “Robust block second order sliding mode control for a quadrotor,” *Journal of the Franklin Institute*, vol. 349, no. 2, pp. 719–739, 2012.
- [11] C. A. Arellano, L. Luque–Vega, B. Castillo–Toledo, and A. G. Loukianov, “Backstepping control with sliding mode estimation for a hexacopter,” in *Proceedings of the International Conference on Electrical Engineering, Computing Science and Automatic Control*, pp. 1–6, Mexico City, Mexico, September 2013.
- [12] J.-J. Xiong and G.-B. Zhang, “Global fast dynamic terminal sliding mode control for a quadrotor UAV,” *ISA Transactions*, vol. 66, pp. 233–240, 2017.
- [13] J. A. Ramoz–León, C. Acosta–Lua, M. E. Sánchez–Morales, S. Di Gennaro, and A. Navarrete–Guzmán, “Altitude and attitude non linear controller applied to a quadrotor with a slung load,” in *Proceedings of the IEEE International Autumn Meeting on Power, Electronics and Computing – ROPEC*, pp. 1–6, Ixtapa, Mexico, November 2019.
- [14] D. Matouk, O. Gherouat, F. Abdessemed, and A. Hassam, “Quadrotor position and attitude control via backstepping approach,” in *Proceedings of the 8th International Conference on Modelling, Identification and Control – ICMIC–2016*, Algiers, Algeria, November 2016.
- [15] S. Islam, J. Dias, and L. D. Seneviratne, “Adaptive output feedback control for miniature unmanned aerial vehicle,” in *Proceedings of the 2016 IEEE International Conference on Advanced Intelligent Mechatronics – AIM 2016*, pp. 318–322, Alberta, Canada, July 2016.
- [16] A. Moeini, A. F. Lynch, and Q. Zhao, “Disturbance observer–based nonlinear control of a quadrotor UAV,” *Advanced Control for Applications*, vol. 2, no. 24, pp. 1–20, 2019.
- [17] Y.-C. Liu, J. Zhang, T. Zhang, and J.-Y. Song, “Robust adaptive spacecraft attitude tracking control based on similar skew-symmetric structure,” *Computers & Electrical Engineering*, vol. 56, pp. 784–794, 2016.
- [18] S. Raiesdana, “Control of quadrotor trajectory tracking with sliding mode control optimized by neural networks,” *Journal of Systems and Control Engineering*, vol. 234, pp. 1–19, 2020.
- [19] T. Merz, S. Duranti, and G. Conte, “Autonomous landing of an unmanned helicopter based on vision and inertial sensing,” *Experimental Robotics IX, Springer Tracts in Advanced Robotics*, vol. 21, pp. 343–352, 2006.
- [20] D. Falanga, A. Zanchettin, A. Simovic, J. Delmerico, and D. Scaramuzza, “Vision–based autonomous quadrotor landing on a moving platform,” in *Proceedings of the 2017*

- IEEE International Symposium on Safety, Security and Rescue Robotics – SSRR*, pp. 1–6, Shanghai, China, October 2017.
- [21] S. Lange, N. Sunderhauf, and P. Protzel, “A vision based onboard approach for landing and position control of an autonomous multirotor UAV in GPS-denied environments,” in *Proceedings of the IEEE International Conference on Advanced Robotics – ICAR*, pp. 1–6, Munich, Germany, June 2009.
- [22] D. Lee, T. Ryan, and H. J. Kim, “Autonomous landing of a VTOL UAV on a moving platform using image-based visual servoing,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 971–976, Saint Paul, MN, USA, May 2012.
- [23] J. L. Sanchez-Lopez, J. Pestana, S. Saripalli, and P. Campoy, “An approach toward visual autonomous ship board landing of a VTOL UAV,” *Journal of Intelligent & Robotic Systems*, vol. 74, no. 1-2, pp. 113–127, 2014.
- [24] W. Wu, X. Jin, and Y. Tang, “Vision-based trajectory tracking control of quadrotors using super twisting sliding mode control,” *Cyber-Physical Systems*, vol. 2020, pp. 1–24, 2020.
- [25] X. Shao, N. Liu, Z. Wang, W. Zhang, and W. Yang, “Neuroadaptive integral robust control of visual quadrotor for tracking a moving object,” *Mechanical Systems and Signal Processing*, vol. 136, pp. 1–17.
- [26] K. Mueller, M. Fennel, and G. F. Trommer, “Model predictive control for vision-based quadrotor guidance,” in *Proceedings of the IEEE/ION Position, Location and Navigation Symposium – PLANS*, pp. 50–61, Portland, OR, USA, April 2020.
- [27] B. Zhao, Y. Tang, C. Wu, and W. Du, “Vision-based tracking control of quadrotor with backstepping sliding mode control,” *IEEE Access*, vol. 6, pp. 72439–72448, 2018.
- [28] M. Shirzadeh, H. J. Asl, A. Amirkhani, and A. A. Jalali, “Vision-based control of a quadrotor utilizing artificial neural networks for tracking of moving targets,” *Engineering Applications of Artificial Intelligence*, vol. 58, pp. 34–48, 2017.
- [29] N. Liu, X. Shao, and W. Yang, “Integral barrier Lyapunov function based saturated dynamic surface control for vision-based quadrotors via back-stepping,” *IEEE Access*, vol. 6, pp. 63292–63304, 2018.
- [30] P. C. Hughes, *Spacecraft Attitude Dynamics*, Dover Publications, Inc., Mineola, NY, USA, 1986.
- [31] H. K. Khalil, *Nonlinear Systems*, Prentice-Hall, Upper Saddle River, NJ, USA, 2002.